

Excel 2002 Kompendium

Said Baloui

Excel 2002

Professionell analysieren und planen

Bitte beachten Sie: Der originalen Printversion liegt eine CD-ROM bei.
In der vorliegenden elektronischen Version ist die Lieferung einer CD-ROM nicht enthalten.
Alle Hinweise und alle Verweise auf die CD-ROM sind ungültig.

KOMPENDIUM

Arbeitsbuch
Nachschlagewerk
Praxisführer

Markt+Technik Verlag

Die Deutsche Bibliothek – CIP-Einheitsaufnahme

Ein Titeldatensatz für diese Publikation ist bei
Der Deutschen Bibliothek erhältlich.

Die Informationen in diesem Buch werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht. Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt. Bei der Zusammenstellung von Texten und Abbildungen wurde mit größter Sorgfalt vorgegangen. Trotzdem können Fehler nicht vollständig ausgeschlossen werden. Verlag, Herausgeber und Autoren können für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen. Für Verbesserungsvorschläge und Hinweise auf Fehler sind Verlag und Herausgeber dankbar.

Alle Rechte vorbehalten, auch die der fotomechanischen Wiedergabe und der Speicherung in elektronischen Medien. Die gewerbliche Nutzung der in diesem Produkt gezeigten Modelle und Arbeiten ist nicht zulässig.

Fast alle Hardware- und Softwarebezeichnungen, die in diesem Buch erwähnt werden, sind gleichzeitig auch eingetragene Warenzeichen oder sollten als solche betrachtet werden.

Umwelthinweis:

Dieses Buch wurde auf chlorfrei gebleichtem Papier gedruckt.

Die Einschrumpffolie – zum Schutz vor Verschmutzung – ist aus umweltverträglichem und recyclingfähigem PE-Material.

10 9 8 7 6 5 4 3 2 1

05 04 03 02 01

ISBN 3-8272-6047-7

© 2001 by Markt+Technik Verlag,
ein Imprint der Pearson Education Deutschland GmbH,
Martin-Kollar-Straße 10–12, D-81829 München/Germany
Alle Rechte vorbehalten
Einbandgestaltung: Grafikdesign Heinz H. Rauner, Gmund
Lektorat: Rainer Fuchs, rfuchs@pearson.de
Herstellung: Elisabeth Egger, eegger@pearson.de
Satz: reemers publishing services gmbh, Krefeld (www.reemers.de)
Druck und Verarbeitung: Bercker, Kvelaer
Printed in Germany

Für Maria,
meine »Co-Autorin«

Excel 2002 Kompendium

Inhaltsübersicht

	Vorwort	27
	Hinweise zu den verwendeten Symbolen	29
	Was ist neu – und wo finden Sie es?	31
Teil 1	Bedienung	57
Kapitel 1	Benutzeroberfläche	59
Teil 2	Tabellen	105
Kapitel 2	Grundlagen	107
Kapitel 3	Dateien	139
Kapitel 4	Selektieren	157
Kapitel 5	Editieren	163
Kapitel 6	Formatieren	213
Kapitel 7	Drucken	271
Kapitel 8	Arbeitsmappen und Vorlagen	297
Kapitel 9	Namen	331
Kapitel 10	Gliederungen	347
Kapitel 11	Funktionen	357
Teil 3	Datenverwaltung und Datenanalyse	393
Kapitel 12	Was-wäre-wenn-Analysen	395
Kapitel 13	Listen und Datenbanken	417
Kapitel 14	Microsoft Query – Zugriff auf externe Datenbanken	445

Kapitel 15	Konsolidieren	473
Kapitel 16	Pivot-Tabellen	485
Kapitel 17	Mehrfachoperationen	519
Kapitel 18	Matrixformeln	529
Kapitel 19	Datenexport und Datenimport	539
Teil 4	Grafikobjekte und Diagramme	575
Kapitel 20	Grafikobjekte	577
Kapitel 21	Bildobjekte aller Art	589
Kapitel 22	Diagramme erzeugen	597
Kapitel 23	Diagrammarten	629
Kapitel 24	Diagramme formatieren	655
Teil 5	Excel und das Internet	695
Kapitel 25	Hyperlinks	699
Kapitel 26	Excel-Tabellen im Internet/Intranet	711
Kapitel 27	Webseiten	721
Kapitel 28	Webabfragen	735
Kapitel 29	Webserver einrichten	745
Kapitel 30	Sonstiges	759
Teil 6	Makros und VBA-Programme	763
Kapitel 31	Makros	765
Kapitel 32	Visual Basic	773
Kapitel 33	Sprachelemente	817
Kapitel 34	Kontrollstrukturen	873
Kapitel 35	Objekte	903
Kapitel 36	Tabellen und Diagramme	919

Kapitel 37	Formulare und Steuerelemente	949
Kapitel 38	Praxisführer	985
Teil 7	Anhang	1063
A	Anhang	1065
	Stichwortverzeichnis	1091

Inhaltsverzeichnis

	Vorwort	27
	Hinweise zu den verwendeten Symbolen	29
	Was ist neu – und wo finden Sie es?	31
	Tabellarische Übersicht	31
	Bedienung	34
	Tabellen	37
	Datenverwaltung und Datenanalyse	49
	Grafikobjekte, Landkarten und Diagramme	51
	Excel und das Internet	52
	Sonstiges	54
TEIL 1	BEDIENUNG	57
Kapitel 1	Benutzeroberfläche	59
1.1	Vor dem Start	59
	Installation	60
	Aufruf	64
	Aktivierung	65
	Komponenten des Excel-Fensters	67
1.2	Menüs, Symbolleisten und Dialogfelder	68
	Menüs	68
	Symbolleisten	71
	Dialogfelder	76
	Aufgabenbereich	82
	Smart Tags	84
1.3	Fenster	85
	Bildlauf	85
	Fenstermodi	88

Inhaltsverzeichnis

Größe und Position. 90

Mit mehreren Fenstern arbeiten 91

1.4 Arbeitsmappen 94

1.5 Hilfestellungen. 96

Kontextsensitive Hilfe 96

Der Office-Assistent 97

Das Hilfefenster 100

TEIL 2 TABELLEN. 105

Kapitel 2 Grundlagen 107

2.1 Dateneingabe 107

2.2 Editieren 109

2.3 Texte, Zahlen und Symbole 110

2.4 AutoEingabe 113

2.5 Formeln 114

Hilfestellungen. 116

Zellbezüge. 118

Bezugsarten 120

Formeln kopieren. 124

Funktionen 127

Bereichsangaben. 131

Summen, Mittelwerte etc. 133

Formeln testen 134

AutoBerechnung 135

2.6 Kommentare 136

2.7 Symbole. 137

Kapitel 3 Dateien 139

3.1 Speichern 139

Speicheroptionen. 144

Dateieigenschaften 147

Speichern unter... 150

Arbeitsbereich speichern 151

3.2 Öffnen 152

Optionen beim Öffnen 153

Dateien wiederherstellen 155

Kapitel 4	Selektieren	157
4.1	Bereiche	157
4.2	Mehrfachselektion	159
4.3	Spalten, Zeilen und das Arbeitsblatt	160
Kapitel 5	Editieren	163
5.1	Rückgängig machen und Wiederholen	164
5.2	Kontextmenüs	164
5.3	Demo »Kostenübersicht«	166
5.4	Dateneingabe in selektierten Bereichen	167
5.5	Verschieben	169
5.6	Einfügen, Entfernen und Löschen	171
5.7	Spalten und Zeilen behandeln	177
5.8	Kopieren	179
5.9	Inhalte einfügen	185
	Inhalte auswählen	186
	Leerzellen überspringen	187
	Transponieren	187
	Rechenoperationen	188
	Verknüpfen	189
5.10	Füllen	190
	AutoAusfüllen	190
	Reihen berechnen	196
5.11	Suchen und Ersetzen	201
5.12	Gehe zu	204
	Bestimmte Datenarten suchen	206
	Zeilen- und Spaltenunterschiede suchen	208
	Vorgänger- und Nachfolgerzellen suchen	211
Kapitel 6	Formatieren	213
6.1	Die Format-Symboleiste	214
6.2	Formate zuweisen und kopieren	216
6.3	Demo »Buchführung«	218

Inhaltsverzeichnis

6.4	Zahlen-, Datums- und Zeitformate	220
	Zahlenformate	222
	Datums- und Zeitformate	228
	Selbstdefinierte Formate	232
6.5	Schriftformate	236
	Schriftarten und -farben	236
	Druckerschriftarten und TrueType-Schriften	241
6.6	Spaltenbreite und Zeilenhöhe	243
	Mit der Maus	243
	Menübefehle	245
6.7	Text- und Zahlenausrichtung	247
6.8	Rahmen	252
6.9	Muster und Blatthintergrund	255
6.10	Formatvorlagen	257
	Definition und Zuweisung	257
	Formatkomponenten	258
	Ändern, Löschen und Zusammenführen	259
	Formatierungs-Voreinstellungen ändern	261
6.11	AutoFormatieren	262
6.12	Bedingte Formatierung	266
Kapitel 7	Drucken	271
7.1	Druckerinstallation	271
	Drucker einrichten	274
7.2	Ausdrucken	277
	Seitenansicht	278
	Interaktives Layouten	280
7.3	Seitenlayout	281
	Papierformat	281
	Seitenränder	283
	Kopfzeile/Fußzeile	284
	Tabelle	287
	Manueller Seitenumbruch	291

Kapitel 8	Arbeitsmappen und Vorlagen	297
8.1	Teamarbeit – Arbeitsmappen gemeinsam benutzen	298
	Mappen versenden und weiterleiten	298
	Arbeitsmappen freigeben	300
	Änderungen verfolgen und rückgängig machen	303
	Arbeitsmappen zusammenführen	306
8.2	Gruppen bilden und editieren	307
8.3	Umbenennen, Aus-/Einblenden	309
8.4	Löschen, Einfügen, Verschieben und Kopieren	311
8.5	Fenster-Handling	314
	Mehrere Fenster für eine Mappe	314
	Aus- und Einblenden	315
	Unterfenster	316
	Fenster fixieren	317
	Zoom-Funktion	318
8.6	3D-Verknüpfungen	319
8.7	Externe Bezüge	322
8.8	Arbeitsmappen als Vorlagen verwenden und automatisch öffnen	326
	Startordner	326
	Mustervorlagen	327
	Automatische Mustervorlagen	329
	Mitgelieferte Vorlagen	329
Kapitel 9	Namen	331
9.1	Definieren und Editieren	331
9.2	Namen erstellen	338
9.3	Namen übernehmen	339
9.4	Schnittpunkt-Operator und impliziter Schnittpunkt	341
Kapitel 10	Gliederungen	347
10.1	Automatische Gliederungserstellung	347
10.2	Manuelle Gliederungen	351
10.3	Selektives Aus- und Einblenden	355

- Kapitel 11 Funktionen 357**
 - 11.1 Funktionsvorschrift und Argument 357**
 - 11.2 Verschachtelte Funktionsaufrufe. 358**
 - 11.3 Datentypen 359**
 - 11.4 Funktionsgruppen 362**
 - Mathematische Funktionen. 362
 - Textfunktionen. 364
 - Datum- und Uhrzeitfunktionen 365
 - Statistikfunktionen 369
 - Suchfunktionen 374
 - Logische Funktionen 380
 - Informationsfunktionen. 387
 - Finanzmathematische Funktionen 387
 - Trigonometrische Funktionen 388
 - Matrixfunktionen 389
 - Datenbankfunktionen 389
- TEIL 3 DATENVERWALTUNG UND DATENANALYSE. 393**
- Kapitel 12 Was-wäre-wenn-Analysen 395**
 - 12.1 Zielwertsuche 396**
 - 12.2 Solver 397**
 - 12.3 Szenario-Manager 408**
 - 12.4 Ansichten-Manager 415**
- Kapitel 13 Listen und Datenbanken 417**
 - 13.1 Sortieren 418**
 - Sortierkriterien 418
 - Sortieroptionen 423
 - 13.2 Filtern 426**
 - AutoFilter 426
 - Top 10-Auswahl 428
 - Filterkriterien 429
 - Spezialfilter 432
 - 13.3 Datenmasken 438**
 - 13.4 Gruppierte Teilergebnisse 441**

Kapitel 14	Microsoft Query – Zugriff auf externe Datenbanken	445
14.1	Die Funktionsweise von Query	445
14.2	Externe Daten in Excel importieren	447
14.3	Datenbanken erstellen/verändern	455
14.4	Abfragen	459
	Sortieren und Filtern	463
	Verknüpfte Tabellen	465
Kapitel 15	Konsolidieren	473
15.1	Statische Verknüpfungen	474
15.2	Orientierung an Beschriftungen	478
15.3	Dynamische Verknüpfungen	480
Kapitel 16	Pivot-Tabellen	485
16.1	Der Pivot-Tabellen-Assistent	487
16.2	Layout- und Tabellen-Optionen	489
16.3	Layouteinstellungen	492
16.4	Feldeinstellungen	495
	Datenfelder	495
	Beschriftungs- und Seitenfelder	498
16.5	Mehr als zwei Variablen	499
16.6	Sortieren und Gliedern	501
16.7	Anzeige-Einschränkung und Seitenanzeige	505
16.8	Zugriff auf externe Datenquellen	507
	MS Query	507
	OLAP-Cubes	511
16.9	Pivot-Diagramme und Berichte	517
Kapitel 17	Mehrfachoperationen	519
17.1	Mit einem Eingabefeld	519
17.2	Testen mehrerer Formeln	523
17.3	Mehrfachoperationsbereiche editieren	524
17.4	Mit zwei Eingabefeldern	525

Kapitel 18	Matrixformeln	529
18.1	Eingabe und Anwendung	529
18.2	Editieren	534
18.3	Matrixkonstanten und Matrixexpansion	535
Kapitel 19	Datenexport und Datenimport	539
19.1	Datenbankprogramme und ODBC	540
19.2	Dateiformate	541
19.3	Im Textformat exportieren	543
19.4	Textdatei-Import – der Textkonvertierungs-Assistent	545
	Trennzeichen	546
	Feste Spaltenbreiten	551
	Variable Leerzeichenfolgen	552
	Ein nützliches Hilfsprogramm	555
19.5	Spezifische Dateiformate	558
	Multiplan	558
	Lotus und Symphony	559
	dBase	560
19.6	Windows-Programme	561
	Die Office-Zwischenablage	562
	Verknüpfen	565
	Einfügen und Einbetten	569
	OLE 2.0	571
TEIL 4	GRAFIKOBJEKTE UND DIAGRAMME	575
Kapitel 20	Grafikobjekte	577
20.1	Erzeugen	577
20.2	Selektieren und manipulieren	579
20.3	Formatieren	582
Kapitel 21	Bildobjekte aller Art	589
21.1	Grafikdateien	589
21.2	WordArt	591
21.3	AutoFormen	591
21.4	ClipArt	593

Kapitel 22	Diagramme erzeugen	597
22.1	Diagrammelemente	598
22.2	Der Diagramm-Assistent	599
	Diagramm erzeugen	600
	Diagramm verändern	603
22.3	Datenreihen und Rubriken	606
22.4	Diagrammblätter	611
22.5	Zusätzliche Diagrammelemente einfügen	613
	Diagramm- und Achsentitel	614
	Achsen	614
	Gitternetzlinien	616
	Legende	617
	Datenbeschriftung	618
	Datentabelle	619
	Ungebundene Texte	619
	Trendlinien	619
	Zusätzliche Daten	624
Kapitel 23	Diagrammarten	629
23.1	Diagrammtypen und -varianten	630
23.2	3D-Diagramme	647
23.3	Verbunddiagramme	650
23.4	Benutzerdefinierte Diagrammtypen und der Standard-Diagrammtyp	652
Kapitel 24	Diagramme formatieren	655
24.1	Selektieren	656
24.2	Elemente löschen	659
24.3	Formatier-Dialogfelder	659
24.4	Zeichnungsfläche	662
24.5	Diagrammfläche	663
24.6	Legende	665
24.7	Datenpunkte	666
24.8	Datenreihen	668
	Achsen	668

Inhaltsverzeichnis

	Fehlerindikatoren	670
	Datenreihenanzordnung	674
	Optionen	674
24.9	Achsen	681
	Ausrichtung	681
	Achsenwerte	682
	Achsenmuster	683
	Skalierung der Größenachse	684
	Skalierung der Rubrikenachse	691
TEIL 5	EXCEL UND DAS INTERNET	695
Kapitel 25	Hyperlinks	699
25.1	Hyperlinkfelder	699
25.2	Hyperlinks bearbeiten	702
25.3	Die Web-Symboleiste	702
25.4	Links zu beliebigen Dokumenten	704
	Dateien	704
	Webseiten	707
	Neue Dokumente	708
	E-Mail-Adressen	708
Kapitel 26	Excel-Tabellen im Internet/Intranet	711
26.1	FTP-Server	711
26.2	Webordner	714
26.3	Verknüpfungen	718
Kapitel 27	Webseiten	721
27.1	Importieren	721
27.2	Als Webseite speichern	723
27.3	Interaktive Webseiten	728
Kapitel 28	Webabfragen	735
28.1	Ausführen	735
28.2	Erstellen und Editieren	738
28.3	Daten aktualisieren	741

28.4	Abfrage-Eigenschaften und Parameter	741
28.5	Aktienkurse abfragen per Smart Tag	743
Kapitel 29	Webserver einrichten	745
29.1	Netzwerk konfigurieren.	746
29.2	Webserver konfigurieren.	751
29.3	Kritische Punkte	758
Kapitel 30	Sonstiges	759
30.1	Weboptionen	759
30.2	Bilder komprimieren	760
TEIL 6	MAKROS UND VBA-PROGRAMME	763
Kapitel 31	Makros	765
31.1	Der Makro-Recorder	765
	Aufzeichnen und ausführen	766
	Bearbeiten	768
31.2	Auto-Makros	769
31.3	Makros mit Objekten verknüpfen	770
Kapitel 32	Visual Basic	773
32.1	VBA-Projekte – ein Überblick	776
	Funktionsprozeduren	785
	Sub-Prozeduren	790
	Ereignisprozeduren	792
32.2	Programme eingeben und testen	794
	Bearbeitungsmöglichkeiten	794
	Objektkatalog	798
	Das Direktfenster	801
	Fehlerverfolgung	803
32.3	Add-Ins erstellen	815
Kapitel 33	Sprachelemente	817
33.1	Ein-/Ausgabeelemente	818
	Debug.Print	818

Inhaltsverzeichnis

	MsgBox	819
	InputBox	822
33.2	Codezeilen, Anweisungen und Aktionen	824
33.3	Variablen	824
	Datentypen	826
	Deklarationen	829
33.4	Konstanten	832
	Selbstdefinierte Konstanten	832
	Excel- und VBA-Konstanten	833
33.5	VBA-Prozeduren	834
33.6	Selbstdefinierte Prozeduren	838
	Lokale Variablen	842
	Übergabe von Argumenten	844
	Benannte und optionale Argumente	846
	Als Referenz	847
	Als Wert	849
33.7	Geltungsbereiche	851
	Prozeduren	851
	Variablen und Konstanten	852
	Bibliotheken und Prozedurbezüge	854
33.8	Arrays (Datenfelder)	857
33.9	Selbstdefinierte Datentypen (Verbunde)	866
Kapitel 34	Kontrollstrukturen	873
34.1	Verzweigungen	874
	Entscheidungen und Bedingungen	876
	Mehrfachtests	882
34.2	Schleifen	884
	For...Next-Schleifen	884
	For Each...Next-Schleifen	887
	While- und Do-Schleifen	889
	Abbruch einer Schleife oder Prozedur	892
34.3	Laufzeitfehler behandeln	893

Kapitel 35	Objekte	903
35.1	Objektypen	903
35.2	Eigenschaften und Methoden	908
35.3	Bezüge	912
35.4	Objektvariablen	914
35.5	Property-Prozeduren	916
Kapitel 36	Tabellen und Diagramme	919
36.1	Referenzieren und Manipulieren	920
36.2	Tabellenfunktionen in VBA	932
36.3	Demo »Aktienkursaktualisierung«	934
36.4	Demo »Diagrammaktualisierung«	943
Kapitel 37	Formulare und Steuerelemente	949
37.1	Steuerelemente in Tabellen	950
37.2	Der Entwurfsmodus und das Eigenschaftenfenster	951
37.3	Ereignisprozeduren	954
37.4	Formulare	955
37.5	Formulare in Excel-Anwendungen integrieren	961
37.6	Programmgesteuerte Veränderung von Eigenschaften	965
37.7	Umgang mit Steuerelementen	966
	Mehrseitige Dialogfelder	968
	Bezeichnungs-, Text- und RefEdit-Felder	969
	Drehfelder und Bildlaufleisten	970
	Kontrollkästchen, Umschaltfelder und Optionsgruppen	973
	Anzeigefelder	975
	Listen-, Dropdown- und Kombinationsfelder	976
	Befehls-Schaltflächen	977
37.8	Demo »Formulargesteuerte Kursaktualisierung«	979
Kapitel 38	Praxisführer	985
38.1	Menü- und Symbolleisten anpassen	985
38.2	Excel-Datenbanklisten mit Access verwalten	994

Inhaltsverzeichnis

38.3	Fehler in Tabellen aufdecken	996
	Fehlerüberprüfung	997
	Fehler-Smart Tag.	997
	Überwachungsfenster	998
	Formelüberwachungsmodus.	999
	Formelauswertung	1000
	Formelüberwachung	1001
38.4	Gültigkeitsregeln benutzen	1004
38.5	Dokumente schützen	1006
	Blätter schützen.	1006
	Zellen entsperren.	1007
	Arbeitsmappen schützen	1008
38.6	Zusätzliche Add-Ins verwenden	1009
	Statistische Analysen	1010
	Excel-4.0-Kompatibilität.	1013
	Teilsummen	1013
	Verweise	1017
	Euro-Währungen umrechnen	1020
38.7	Buchführung leicht gemacht	1022
	Anpassen der Anwendung	1024
	Einnahmen und Ausgaben erfassen	1026
	Monatliche Umsatzsteuer-Voranmeldungen.	1029
	Quartals-Umsatzsteuer-Voranmeldungen.	1030
	Überschussrechnung	1031
38.8	Mit VBA auf Dateien beliebiger Art zugreifen	1034
	Dateiformen	1035
	Dateien öffnen und schließen	1038
	Sequentielle Dateien	1041
	Direktzugriffsdateien	1050
	Binärdateien	1059
TEIL 7	ANHANG	1063
A	Anhang	1065
A.1	Die Demos auf der CD	1065
A.2	Dateisuche	1066

A.3	Der Druck-Manager	1068
A.4	Rechtschreibprüfung	1070
A.5	AutoKorrektur	1073
A.6	Add-Ins	1074
A.7	Excel individuell konfigurieren	1076
	Ansicht	1077
	Berechnen	1078
	Bearbeiten	1080
	Umsteigen	1081
	Allgemein	1082
	Benutzerdefinierte Listen	1083
	Diagramm	1084
	Farbe	1085
	International	1087
	Speichern	1088
	Fehlerüberprüfung	1088
	Rechtschreibung	1088
	Sicherheit	1090
	Stichwortverzeichnis	1091

Vorwort



Falls Sie als erfahrenen Excel-Anwender zunächst interessiert, was Excel an Neuem bietet: »Was ist neu – und wo finden Sie es?« beschreibt alle wesentlichen Neuerungen und Änderungen gegenüber der vorhergehenden Version in Kurzform und verweist zusätzlich auf das Kapitel, in dem das betreffende Thema ausführlich erläutert wird!

In Teil 1 erhalten Sie einen Schnellkursus zu den Eigenheiten der Excel-Oberfläche, in dem es um Excel-typische Elemente wie den Office-Assistenten und konfigurierbare Menü- und Symbolleisten geht.

In Teil 2 erläutere ich die eigentliche Tabellenkalkulation mit Excel, angefangen beim Erstellen, Formatieren und Drucken von Tabellen bis hin zur Teamarbeit und der gleichzeitigen Nutzung von Arbeitsmappen durch mehrere Anwender, komplexen 3D- und Internetverknüpfungen und dem Umgang mit Vorlagen und Gliederungen. Danach erläutere ich den Umgang mit den verschiedenen Funktionsgruppen von Excel.

In Teil 3 geht es anschließend um die Datenverwaltung und Datenanalyse mit den Was-wäre-wenn-Analysen, für die Excel unterschiedliche Werkzeuge wie den Solver oder den Szenario-Manager bereitstellt, aber auch Techniken zur Datenkonsolidierung und Excels Pivot-Tabellen.

Darüber hinaus erläutere ich den Einsatz von Excel als Mini-Datenbank, mit der Listen verwaltet, sortiert, gefiltert und gruppiert werden können – und Microsoft Query, mit dem Excel wunderbar zusammenarbeitet und das bereits eine ausgewachsene Datenbank darstellt.

Zusätzlich zeige ich, welche vielfältigen Möglichkeiten Excel zum Datenaustausch mit anderen Anwendungen bietet: mithilfe von unterschiedlichen Dateiformaten und mit Hilfsmitteln wie ODBC und Query und mit Techniken wie DDE und OLE.

Im Teil 4 geht es um das integrierte Zeichen- und das Grafikpaket von Excel. Das Zeichenpaket ermöglicht Ihnen, nicht nur einfache Grafikobjekte wie Pfeile oder Kreise in Ihre Tabellen einzufügen, sondern auch Grafiken, die sich in Dateien befinden, beispielsweise eingescannte Fotos oder WordArt-Objekte.

Mit dem Grafikpaket wiederum können Sie aus Ihren Zahlenwüsten zwei- oder dreidimensionale Diagramme erzeugen und diese in vielfältigster Weise manipulieren, um beispielsweise nachträglich hinzugekommene Daten, Fehlerindikatoren und Trendlinien, externe Grafiken oder Zeichnungen hinzuzufügen.

Der Teil 5 dreht sich ausschließlich um Excel und das Internet: um Hyperlinks, um das Speichern von Tabellen im Internet und als Webseiten, um Weborder und die vielfältigen Möglichkeiten von Webabfragen und nicht zuletzt um die Voraussetzungen zum Publizieren von Webseiten wie die Einrichtung eines Webserver und spezieller Webserver-Software.

In Teil 6 geht es um die Aufzeichnung von Makros, vor allem um das Erstellen von Programmen mit VBA. Letzteres ermöglicht Ihnen nicht nur das Schreiben selbstdefinierter Funktionen, sondern auch die Entwicklung integrierter Anwendungen, die Excel-Tabellen und Diagramme manipulieren.

Um Lesern, die bereits über Basic-Kenntnisse verfügen, unnötige Sucherei zu ersparen: Die in Sachen Programmierung für Sie wichtigsten Abschnitte enthalten die Kapitel 35, »Objekte«, und Kapitel 36, »Tabellen und Diagramme«, da in diesen – und in den folgenden – Kapiteln erläutert wird, wie Sie Ihre bereits vorhandenen Programmierkenntnisse in Excel anwenden können!

Viel Spaß mit dem vorliegenden Buch!



Hinweise zu den verwendeten Symbolen

Mit Hilfe dieses Symbols gebe ich Ihnen Informationen, die Ihnen helfen können, Excel leichter zu bedienen.



Dieses Hinweissymbol soll Sie auf besonders wichtige Textstellen aufmerksam machen, die helfen, typische Fehler zu vermeiden.



Neben diesem Symbol finden Sie wissenswerte Informationen.



Dieses Symbol befindet sich neben Absätzen, die auf Textstellen mit weiterführenden Informationen hinweisen.



Solche Texte beziehen sich auf Inhalte der Begleit-CD.



Hier finden Sie Arbeitsschrittabläufe, die Sie in dieser Reihenfolge nachvollziehen können.



Sonstige Symbole wie jenes neben diesem Absatz befinden sich in den Excel-Symboleisten oder im *Anpassen*-Dialogfeld (siehe Kapitel 1.2.2, »Symboleisten«). Befindet sich neben einem Absatz ein derartiges Symbol, vereinfacht es die darin beschriebene Aktion.



Stichwörter neben einem Absatz weisen Sie auf das Thema hin, das in diesem – und eventuell den folgenden – Absätzen behandelt wird.

*Einfügen und
Löschen*

Was ist neu – und wo finden Sie es?

Für Umsteiger von Excel 2000 ist wohl das größte Problem, im vorliegenden Wälzer möglichst schnell die Textstellen zu finden, die Neuerungen gegenüber jener Version beschreiben.

Tabellarische Übersicht

Die folgende Tabelle listet die interessantesten Neuerungen/Erweiterungen auf (Kleinkram erwähne ich nicht). Der Tabelle folgt eine Kurzerläuterung jedes Themas (teilweise ist diese Kurzerläuterung bereits erschöpfend, dann finden Sie im genannten Kapitel auch nicht mehr darüber).

Bedienung

Stichwort	Kapitel
Aufgabenbereich	1.2.4, »Aufgabenbereich«
Smart Tag	1.2.5, »Smart Tags«
Standard-/Format-Symbolleisten: Ein- bzw. zweizeilige Darstellung	Praxisführer A, »Menü- und Symbolleisten anpassen«
Hilfe-Eingabefeld	1.5.3, »Das Hilfefenster«
Excel-Fenster an Hilfefenster anpassen	1.5.3, »Das Hilfefenster«
Hilfe im Internet	1.5.3, »Das Hilfefenster«

Tabellen

Stichwort	Kapitel
Funktionssuche	2.5.5, »Funktionen«
Bereichsangaben korrigieren	2.5.6, »Bereichsangaben«
Hilfe zu Funktionsargumenten	2.5.6, »Bereichsangaben«

Was ist neu – und wo finden Sie es?

Stichwort	Kapitel
Summen, Mittelwerte etc.	2.5.7, »Summen, Mittelwerte etc.«
Symbole einfügen	2.7, »Symbole«
Dateiauswahl-Dialogfeld	3.1, »Speichern«
Dateiauswahl-Dialogfeld vergrößern/verkleinern	3.1, »Speichern«
Dateien verschlüsselt speichern	3.1.1, »Speicheroptionen«
Dateien reparieren/Daten extrahieren	3.2.1, »Optionen beim Öffnen«
Dateien wiederherstellen	3.2.2, »Dateien wiederherstellen«
Einfügen-Smart Tag	5.8, »Kopieren«
Einfügen-Symbolliste	5.9, »Inhalte einfügen«
Inhalte einfügen: Zusätzliche Optionen	5.9, »Inhalte einfügen«
AutoAusfüllen-Smart Tag	5.10.1, »AutoAusfüllen«
Formatierungen suchen/ersetzen	5.11, »Suchen und Ersetzen«
Landesspezifische Formate	6.4.1, »Zahlenformate«
Von Rechts nach Links orientierte Sprachen	6.7, »Text- und Zahlenausrichtung«
Zell-/Bereichsrahmen zeichnen	6.8, »Rahmen«
Kopf-/Fußzeilen: zusätzliche Elemente	7.3.3, »Kopf-/Fußzeile«
Ausdruck von Fehlerwerten festlegen	7.3.4, »Tabelle«
Farbe der Blattregisterkarten bestimmen	8.3, »Umbenennen, Aus-/Einblenden«
Umgang mit Vorlagen	8.8.2, »Mustervorlagen«, und 8.8.4, »Mitgelieferte Vorlagen«
Zur Überarbeitung versenden	8.1.1, »Mappen versenden und weiterleiten«
Fehlerüberprüfung	Kapitel 38.3, »Fehler in Tabellen aufdecken«, Abschnitt »Fehlerüberprüfung«

Stichwort	Kapitel
Fehler-Smart Tag	Kapitel 38.3, »Fehler in Tabellen aufdecken«, Abschnitt »Fehler-Smart Tag«
Überwachungsfenster	Kapitel 38.3, »Fehler in Tabellen aufdecken«, Abschnitt »Überwachungsfenster«
Formelüberwachungsmodus	Kapitel 38.3, »Fehler in Tabellen aufdecken«, Abschnitt »Formelüberwachungsmodus«
Formelauswertung	Kapitel 38.3, »Fehler in Tabellen aufdecken«, Abschnitt »Formelauswertung«
Passwortgeschützte Bereichsfreigabe in geschützten Blättern	Kapitel 38.5, »Dokumente schützen«, Abschnitt »Zellen entsperren«
Euro-Währungen umrechnen	Kapitel 38.6, »Zusätzliche Add-Ins verwenden«, Abschnitt »Euro-Währungen umrechnen«

Datenverwaltung und Datenanalyse

Stichwort	Kapitel
Datenübergabe von Query an Excel	14.2, »Externe Daten in Excel importieren«
Datenquellen öffnen und definieren	14.2, »Externe Daten in Excel importieren«
Pivot-Tabellen: Felder einfügen	16.2, »Layout- und Tabellenoptionen«
Datenaustausch per Drag&Drop	19.6.1, »Die Office-Zwischenablage«
Office-Zwischenablage	19.6.1, »Die Office-Zwischenablage«
Verknüpfte Objekte	19.6.2, »Verknüpfen«

Grafikobjekte, Landkarten und Diagramme

Stichwort	Kapitel
Objekte rotieren	20.2, »Selektieren und Manipulieren«
Bilder manipulieren	20.4.1, »Grafikdateien«
ClipArt	20.4.4, »ClipArt«
Diagrammelemente selektieren	24.1., »Selektieren«
Datenreihen/-punkte beschriften	22.5.5., »Datenbeschriftungen«

Excel und das Internet

Stichwort	Kapitel
Hyperlinks einfügen	25.1, »Hyperlinkfelder« und 25.4, »Links zu beliebigen Dokumenten«
Webordner: Erstellen und Verknüpfungen	26.2, »Webordner«
Als Webarchiv speichern	27.2, »Als Webseite speichern«
XML-Format	27.2, »Als Webseite speichern«
Interaktive Webseiten erstellen	27.3, »Interaktive Webseiten«
Webabfragen erstellen und editieren	28.2, »Erstellen und Editieren«
Aktienkurs-Smart Tag	28.5, »Aktienkurse abfragen per Smart Tag«
Weboptionen	30.1, »Weboptionen«
Bilder komprimieren	30.2, »Bilder komprimieren«

Sonstiges

Stichwort	Kapitel
Aktivierung von Excel bzw. Office	1.1.3, »Aktivierung«
Dateisuche	Anhang A.2, »Dateisuche«
Rechtschreibprüfung	Anhang A.4, »Rechtschreibprüfung«
AutoKorrektur	Anhang A.5, »AutoKorrektur«
Add-In-Manager: COM-Add-Ins einbinden	Anhang A.6, »Add-Ins«
Zusätzliche Optionen	Anhang A.7, »Excel individuell konfigurieren«

Bedienung

Aufgabenbereich

Mehrere Excel-Befehle öffnen statt eines herkömmlichen Dialogfelds den einen beträchtlichen Teil des Excel-Fensters einnehmenden Aufgabenbereich, beispielsweise der Befehl DATEI|NEU... .



Abbildung 1:
Aufgabenbereich

Dieser Bereich ist eine Mischung aus verschiedenen Komponenten:

- ➔ **Menüs:** Die aufgelisteten Kommandos werden wie »Hyperlinks« dargestellt: Schwebt der Cursor über einem Kommando wie *Leere Arbeitsmappe*, wird es unterstrichen. Klicken Sie darauf, wird das Kommando ausgeführt, in diesem Fall also eine neue Arbeitsmappe erstellt.
- ➔ **Dialogfelder:** Wie ein Dialogfeld enthält der Aufgabenbereich verschiedene Register. Das Umschalten erfolgt über das Menü, das durch Klicken auf den abwärts weisenden Pfeil am oberen Fensterrand geöffnet wird und in dem alle verfügbaren Register aufgelistet werden.
- ➔ **Webbrowser:** Mit den beiden nach rechts bzw. links gerichteten Pfeilen schlagen Sie das zuvor geöffnete Register wieder auf bzw. anschließend wieder das danach geöffnete Register.

Da er sehr viel Platz einnimmt, empfehle ich Ihnen, den Aufgabenbereich zu schließen, wenn er nicht benötigt wird. Wie gesagt, bei Anwahl eines Befehls, zu dessen Durchführung er benötigt wird, erscheint er automatisch.

Davon unabhängig können Sie ihn jederzeit mit **ANSICHT|AUFGABENBE-REICH** einblenden.

*Der Aufgabenbereich erscheint beim Start von Excel automatisch. Wollen Sie das abstellen, klappen Sie im Aufgabenbereich das Register **Neue Arbeitsmappe** auf und deaktivieren das Kontrollkästchen **Beim Start anzeigen** (oder deaktivieren das Kontrollkästchen **Startaufgabenbereich** im Register **Ansicht** des Befehls **EXTRAS|OPTIONEN...**).*



Was ist neu – und wo finden Sie es?

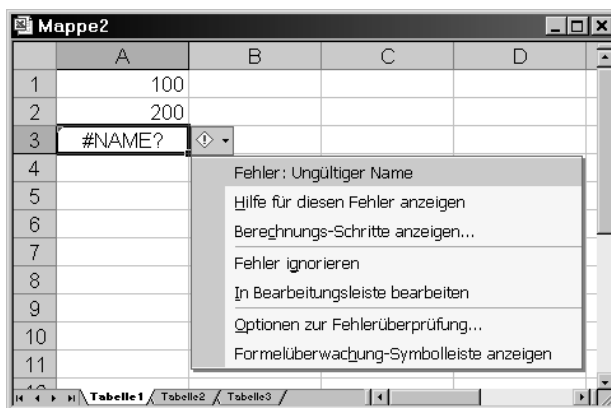
Smart Tags

Ein Smart Tag ist eine Art »Achtung-Schild« an bzw. neben einer Zelle. Zunächst ist es recht dezent, in Form eines kleinen Dreiecks in der linken oberen Ecke der Zelle, dem Smarttag-Indikator.

Tun Sie ihm den Gefallen, auf die Zelle zu klicken, erscheint neben dieser das abgebildete Fehler-Smart Tag, ein deutlich hervorgehobenes Ausrufezeichensymbol.

Klicken Sie darauf, öffnet sich ein Kontextmenü, das zur Fehlerbehebung nützliche Befehle enthält.

Abbildung 2:
Fehler-Smart Tag



Außer diesem Fehler-Smart Tag gibt es noch weitere Smart Tags, die in anderen Situationen erscheinen, beispielsweise beim Kopieren und Einfügen.

Standard-/Format-Symbolleisten: Ein- bzw. zweizeilige Darstellung

Diese Umschaltung können Sie nun blitzschnell vornehmen, indem Sie den abwärts weisenden Pfeil am rechten Ende einer Symbolleiste anklicken und im daraufhin geöffneten Menü den entsprechenden Befehl zur Anzeige in einer bzw. zwei Zeilen wählen.

Hilfe-Eingabefeld

Das listenartige Eingabefeld ganz rechts in der Menüleiste besitzt eine ähnliche Aufgabe wie der Antwort-Assistent. Tippen Sie darin eine Frage ein wie »Wie formatiere ich Zellen?«, erscheint eine Liste mit zum Problem passenden Themen. Wählen Sie eines davon aus, wird das Hilfefenster mit diesem Thema aufgeschlagen.

In der Antwortliste mit »Web: ...« aufgeführte Antworten befinden sich übrigens im Web, das heißt, Ihr Webbrowser wird gestartet und holt sich die Antwort von einer Microsoft-Webseite.

Öffnen Sie die Liste des Eingabefelds, erscheinen darin all Ihre vorhergehenden Fragen, sodass Sie jederzeit die zugehörigen Antwortthemen griffbereit haben.

Excel-Fenster an Hilfefenster anpassen

Das Anklicken dieses Symbols verkleinert und verschiebt das Excel-Fenster so, dass es neben dem Hilfefenster vollständig sichtbar ist (erneutes Anklicken stellt die ursprüngliche Größe des Excel-Fensters wieder her, sodass es womöglich teilweise vom Hilfefenster verdeckt wird.). Ziehen Sie in diesem Zustand an der Trennungslinie, um eines der beiden Fenster zu vergrößern bzw. zu verkleinern, so wird die Größe des anderen Fensters automatisch entsprechend angepasst.



Hilfe im Internet

Klicken Sie im Register *Antwort-Assistent* des Hilfefensters auf die Schaltfläche *Im Web suchen*, so wird Ihnen im rechten Fensterteil ermöglicht, eine Frage einzugeben und auf *Senden und ins Web gehen* zu klicken, woraufhin Ihr Webbrowser gestartet und auf einer speziellen Microsoft-Website nach einer dazu passenden Antwort gesucht wird.

Tabellen

Funktionssuche

Der mit dem abgebildeten Symbol aufgerufene Funktions-Assistent ermöglicht nun die gezielte Suche nach einer Funktion, die zur aktuellen Aufgabenstellung passt.

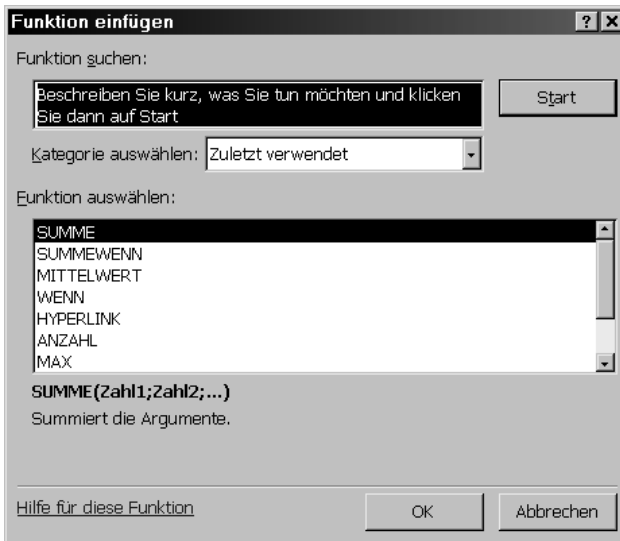


Geben Sie im oberen Eingabefeld eine Beschreibung der gewünschten Tätigkeit ein (»Standardabweichung berechnen«, »Zahl nachschlagen«, »Wert umrechnen« etc.) und klicken Sie auf *Start*, so zeigt Excel daraufhin im unteren Feld die von ihm vorgeschlagenen dazu passenden Funktionen an.

Bereichsangaben korrigieren

Bereichsangaben können Sie nun bereits während der Eingabe per Zeigen korrigieren: Geben Sie »=A1+A2« ein, so wird jede Zelle bzw. jeder Bereich sofort durch einen Rahmen hervorgehoben und Sie können die Bezugsangabe per Ziehen daran korrigieren. Durch Ziehen an den Eckpunkten des Rahmens können Sie den angegebenen Bereich sogar noch während der Eingabe vergrößern oder verkleinern.

Abbildung 3:
Funktionssuche



Hilfe zu Funktionsargumenten

Bei der Eingabe von Funktionsargumenten erscheint ein kleines Fenster. Klicken Sie auf den darin enthaltenen Funktionsnamen, so wird die Hilfe zu dieser Funktion aktiviert. Darüber hinaus zeigt es an, welche Argumente Sie bisher eingegeben haben und welche weiteren Argumente erwartet werden. Zusätzlich können Sie auf eines der bisher eingegebenen Argumente klicken, um es zu ändern, beispielsweise um einen anderen Zellbezug für das betreffende Argument einzugeben.

Summen, Mittelwerte etc.

Das Summe-Symbol ist ein Listenfeld. Öffnen Sie die Liste, können Sie statt *SUMME* eine andere Funktion anwenden, beispielsweise mit *MITTELWERT* den Durchschnitt der Werte im betreffenden Bereich ermitteln oder mit *MAX* den größten darin enthaltenen Wert.

Symbole einfügen

Außer Texten und Zahlen können Sie auch Spezialexsymbole in eine Tabelle eingeben, zum Beispiel Währungssymbole, das mathematische Symbole und so weiter. Dazu wählen Sie *EINFÜGEN|SYMBOL...*, und im darauf erscheinenden Dialogfeld das gewünschte Symbol aus und klicken auf *Einfügen*.

Dateiauswahl-Dialogfeld



Der Pfeil zum Zurückblättern im Dateiauswahl-Dialogfeld bietet nun eine geschlossene Liste. Klicken Sie auf den Listenpfeil, öffnet sich eine Liste mit allen Ordnern, in denen Sie sich zuletzt befanden.

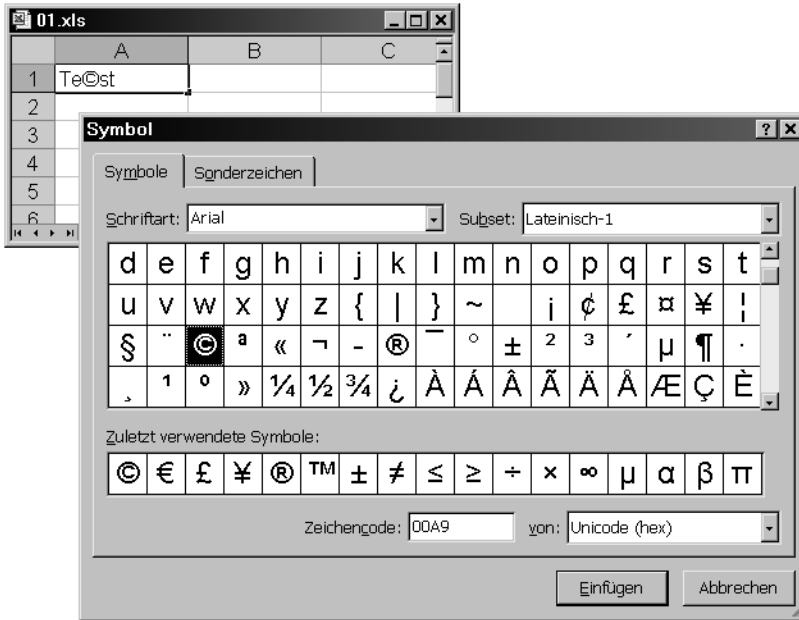


Abbildung 4:
Symbol einfügen

Dateiauswahl-Dialogfeld vergrößern/verkleinern

Das Dateiauswahl-Dialogfeld kann nun wie ein Fenster durch Ziehen an einem der Ränder oder einer Ecke vergrößert bzw. – in Grenzen – verkleinert werden, um gegebenenfalls mehr Dateinamen auf einmal zu sehen.

Dateien verschlüsselt speichern

Sie können Dateien nun verschlüsselt speichern. Dazu wählen Sie zunächst wie gewohnt DATEI|SPEICHERN UNTER... . In der Symbolleiste des zugehörigen Dialogfelds klicken Sie auf das Symbol *Extras* und wählen in der zugehörigen Liste den Befehl ALLGEMEINE OPTIONEN... .

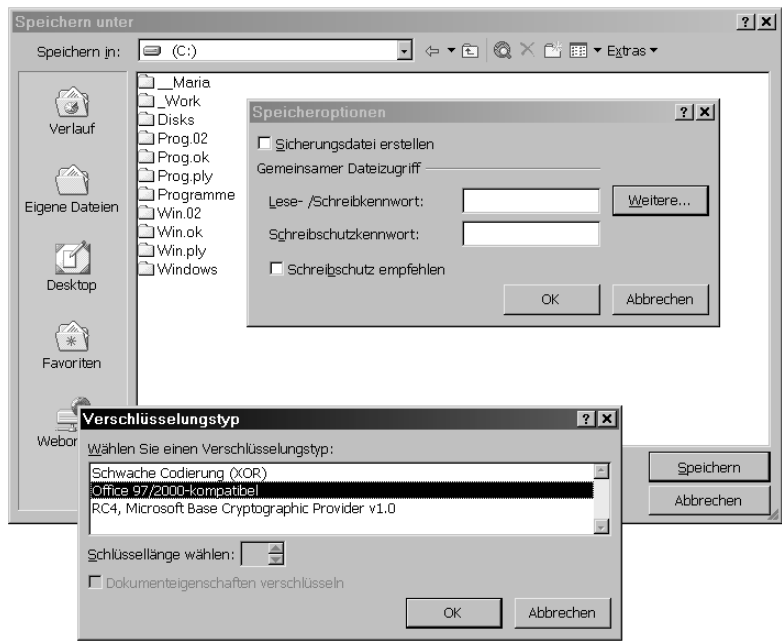
Klicken Sie im daraufhin erscheinenden Dialogfeld auf die Schaltfläche *Weitere...* , so erscheint ein Dialogfeld, in dem Sie unter mehreren Verschlüsselungsverfahren wählen können.

Dateien reparieren/Daten extrahieren

Die Schaltfläche *Öffnen* des Dateiauswahl-Dialogfelds ist bekanntlich ein Listenfeld. Die neue Option ÖFFNEN UND REPARIEREN öffnet ein Dialogfeld, in dem Sie wählen können, ob Excel versuchen soll, die betreffende Datei zu reparieren oder – das ist erst sinnvoll, wenn das Extrahieren scheiterte – soweit möglich die darin enthaltenen Daten zu extrahieren.

Was ist neu – und wo finden Sie es?

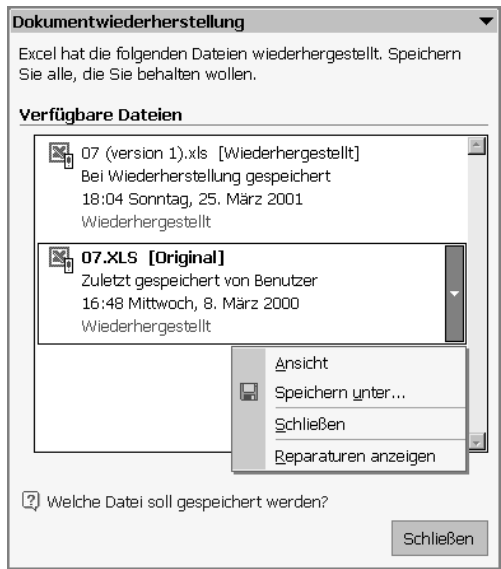
Abbildung 5:
Verschlüsselung



Dateien wiederherstellen

Waren Sie nach vorgenommenen Änderungen nicht mehr in der Lage, eine Datei zu speichern (Stromausfall, Rechnerabsturz etc.) oder trat gar während des Speicherns ein Problem auf, so bietet Ihnen Excel an, beim nächsten Start alle davon betroffenen Dokumente wiederherzustellen – soweit möglich.

Abbildung 6:
Dokumente wiederherstellen



Die betroffenen Dokumente werden aufgelistet und Sie klicken auf das gewünschte, um es zu öffnen oder auf den Listenpfeil eines Dokuments, um die zusätzlichen Optionen zu erhalten, das betreffende Dokument zu speichern oder die vorgenommenen Reparaturen einzusehen (in beiden Fällen wird es zusätzlich auch geöffnet). Anschließend verschwindet das wiederhergestellte Dokument aus der Liste.

Einfügen-Smart Tag

Nach dem Einfügen mit BEARBEITEN|EINFÜGEN erscheint das Einfügen-Smart Tag.

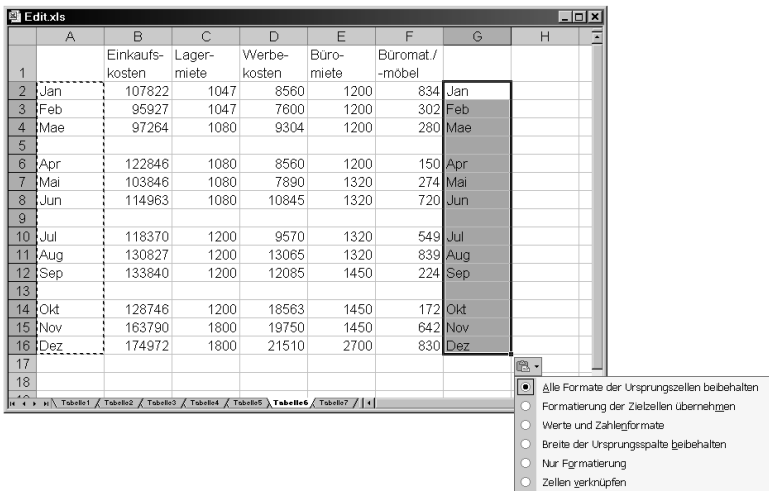
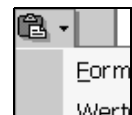


Abbildung 7:
Einfügen-Smart Tag

Das zugehörige Kontextmenü enthält einige beim Einfügen häufig verwendete Optionen, die großteils den verschiedenen Optionen des Befehls BEARBEITEN|INHALTE EINFÜGEN... entsprechen.

Einfügen-Symbolliste

Am rechten Rand des Einfügen-Symbols befindet sich nun ein Listenpfeil. Klicken Sie nicht wie gewohnt auf das Einfügen-Symbol selbst, sondern auf diesen Listenpfeil, erscheint eine Liste mit häufig verwendeten Einfügeoptionen – und zwar mit anderen Optionen, als sie das Einfügen-Smart Tag zur Verfügung stellt.



Inhalte einfügen: Zusätzliche Optionen

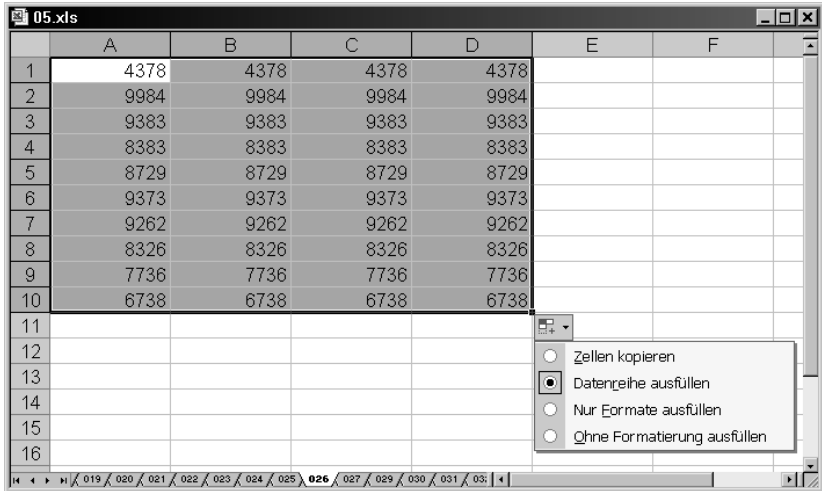
BEARBEITEN|INHALTE EINFÜGEN... wurde um zwei Optionen erweitert:

- ➔ **Formeln und Zahlenformate** fügt nur Formeln (und Werte und Texte) und das Zahlenformat des Quellbereichs ein
- ➔ **Werte und Zahlenformate** fügt nur Zahlen und das Zahlenformat ein

AutoAusfüllen-Smart Tag

Beim Ausfüllen eines Bereichs per Drag&Drop erscheint nach dem Loslassen der Maustaste und dem Ausfüllen des betreffenden Bereichs das AutoAusfüllen-Smart Tag.

Abbildung 8:
AutoAusfüllen-
Smart Tag



Das zugehörige Kontextmenü zeigt durch die aktivierte Option aus, welche Aktion soeben ausgeführt wurde, im Beispiel eine Füllaktion, und ermöglicht Ihnen durch Wählen einer der anderen Optionen, diese Aktion rückgängig zu machen und statt dessen eine der Varianten zum Berechnen von Reihen auszuführen.

Setzen Sie das AutoAusfüllen Smart Tag per Drag&Drop nicht zum Kopieren, sondern zum Berechnen einer Reihe ein, so enthält das Kontextmenü statt dessen die wichtigsten Optionen des Befehls BEARBEITEN|AUSFÜLLEN|REIHE... (das »normale« Kontextmenü – Drag&Drop mit der rechten Maustaste – enthält jedoch mehr).

Formatierungen suchen/ersetzen

Das vollständige Dialogfeld von BEARBEITEN|SUCHEN... sehen Sie nur, wenn Sie auf *Optionen* >> klicken.

Die *Ersetzen*-Funktionen sind nun jederzeit erreichbar, indem darin einfach das gleichnamige Register aufgeklappt wird.

Beide Register enthalten mehrere zusätzliche Optionen, vor allem zum Einbeziehen von Formatierungen in Such- und Ersetzkriterien, beispielsweise um nur den rot formatierten Text »Brutto« durch den blau und fett formatierten Text »Bruttobetrag« zu ersetzen.

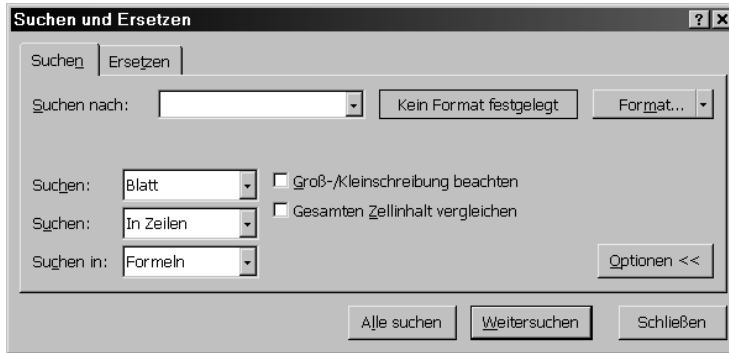


Abbildung 9:
Suchen und Ersetzen

Statt Texten und Zahlen können Sie auch reine Formatierungen in Ihren Tabellen ersetzen. Dazu lassen Sie beide Eingabefelder leer und geben unter *Suchen nach* nur die zu suchende Formatierung an, beispielsweise Kursiv, und unter *Ersetzen durch* die gewünschte neue Formatierung, beispielsweise Rot und Fett.

Landesspezifische Formate

Im Register *Zahlen* des Befehls **FORMAT|ZELLEN...** sind nun für die Kategorien »Datum«, »Uhrzeit« und »Sonderformate« die jeweiligen landesspezifischen Formate verfügbar, indem Sie im Listenfeld *Gebietsschema* das gewünschte Land auswählen.

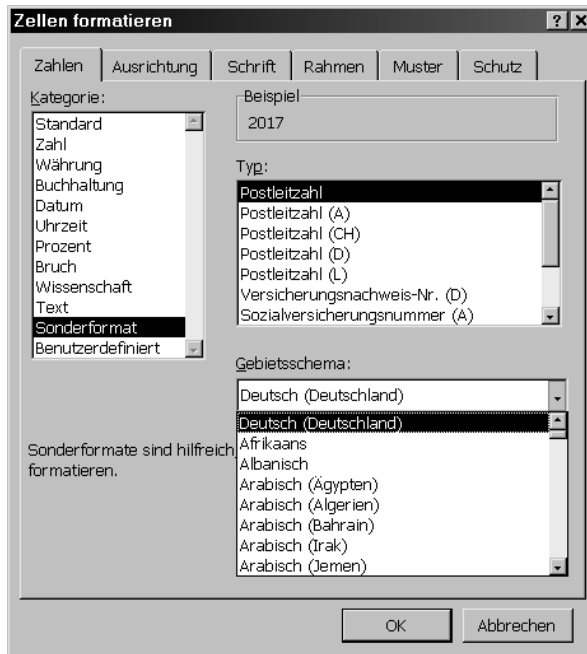


Abbildung 10:
Landesspezifische Formate

Von Rechts nach Links orientierte Sprachen

Das Register *International* des Befehls EXTRAS|OPTIONEN... ermöglicht es Ihnen, ein Excel-Blatt an Sprachen anzupassen, deren Schreibweise von rechts nach links orientiert ist.

Im Register *Ausrichtung* des Befehls FORMAT|ZELLEN... ermöglicht das Listenfeld *Textrichtung* wiederum, das Verhalten einzelner Zellen zu beeinflussen.

Zell-/Bereichsrahmen zeichnen



Am Ende der durch das abgebildete Symbol geöffneten Rahmensymbolleiste befindet sich die Schaltfläche *Rahmenlinien zeichnen...*, die die Rahmenlinien-Symbolleiste öffnet: Deren Stift-Symbol zeichnet beim Anklicken eines Zellrahmens eine Linie (Bereiche: Maustaste gedrückt halten und am entsprechenden Bereichsrand entlangfahren).

Im Listenfeld der Leiste können Sie Dicke bzw. Muster der Linie auswählen, mit dem Farb-Symbol die Linienfarbe bestimmen und mit dem Radiergummi eine Rahmenlinie durch Anklicken wieder entfernen (Maustaste gedrückt halten und über Rahmenlinien fahren = radiert alle Linien wieder weg).

Kopf-/Fußzeilen: zusätzliche Elemente

Für die Gestaltung benutzerdefinierter Kopf-/Fußzeilen stehen Ihnen zusätzliche Elemente zum Einfügen von Pfad plus Dateiname und zum Einfügen (und anschließendem Formatieren) von Grafikdateien zur Verfügung.

Ausdruck von Fehlerwerten festlegen

Im Register *Tabelle* des Dialogfelds *Seite einrichten* können Sie nun festlegen, wie Zellen gedruckt werden, die Fehlerwerte enthalten; ob der Fehlerwert originalgetreu gedruckt werden soll, ob statt dessen eine leere Zelle gedruckt werden soll (also nichts) etc.

Farbe der Blattregisterkarten bestimmen

Klicken Sie mit der rechten Maustaste auf eine Blattregisterkarte, so enthält das Kontextmenü nun zusätzlich den Befehl REGISTERFARBE, der ein Dialogfeld öffnet, in dem Sie eine Farbe auswählen können, mit der die Registerkarten unterstrichen werden.

Umgang mit Vorlagen

Klicken Sie im Aufgabenbereich unter *Neu aus vorhandener Arbeitsmappe* auf *Arbeitsmappe wählen...*, so können Sie im folgenden *Dateiauswahl*-Dialogfeld eine beliebige Excel-Datei als Vorlage für die neue Arbeitsmappe auswählen.

Die zuletzt verwendeten Mustervorlagen erscheinen im Aufgabenbereich unter *Mit Vorlage beginnen* als eigene Einträge.

Die beiden letzten Vorlagenbefehle im Aufgabenbereich ermöglichen Ihnen, Vorlagen zu verwenden, die auf Websites gespeichert sind, die Sie zuvor den Webordnern hinzufügten, oder aber auf der Microsoft-Website, die weitere vorbereitete Vorlagen bereitstellt.

Zur Überarbeitung versenden

Mit dem Befehl DATEI|SENDEN AN|E-MAILEMPFÄNGER (ZUR ÜBERARBEITUNG)... können Sie das Dokument gezielt zur Begutachtung, Kommentierung und Änderung versenden. Die Mappe wird zu diesem Zweck freigegeben und mit dem Kommentar »Bitte überarbeiten...« an eine E-Mail angehängt. Schickt der Empfänger die editierte Mappe an Sie zurück, können Sie anschließend die von ihm vorgenommenen Änderungen verfolgen und individuell annehmen oder ablehnen.

Fehlerüberprüfung

Der Befehl EXTRAS|FEHLERÜBERPRÜFUNG... veranlasst Excel, die aktuelle Tabelle auf Fehler zu überprüfen und in einem Dialogfeld anzuzeigen.

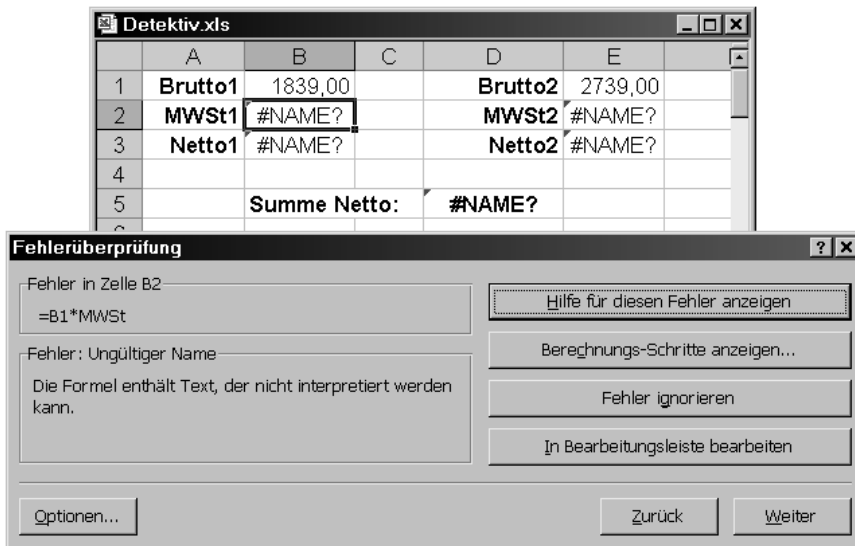


Abbildung 11:
Fehlerüberprüfung

Dabei muss es sich keineswegs nur um »harte« Fehler wie in der gezeigten Abbildung handeln, bei denen ein Fehlerwert wie #NAME in einer Formelzelle erscheint. Sie können Excel dazu bringen, beispielsweise auch nicht geschützte Zellen, die Formeln enthalten, oder Formeln, die auf leere Zellen verweisen, als Fehler zu betrachten.

Fehler-Smart Tag

Erkennt Excel einen Fehler, blendet es das Fehler-Smart Tag ein, ein kleines Dreieck in den oberen linken Ecken der fehlerhaften Zellen. Klicken Sie auf eine davon, erscheint ein Smart Tag, das nach dem Öffnen im Kontextmenü verschiedene Hilfestellungen zum Umgang mit dem Fehler zur Verfügung stellt.

Überwachungsfenster



Der Befehl EXTRAS|FORMELÜBERWACHUNG|ÜBERWACHUNGSFENSTER ANZEIGEN aktiviert die einfachste Hilfestellung von Excel zum Aufdecken von Fehlern, das Überwachungsfenster, mit dem Sie den Inhalt ausgewählter Zellen überwachen können.

Abbildung 12:
Das Überwachungs-
fenster

Arbeitsmappe	Tabelle	Name	Ze...	Wert	Formel
DETEKTIV.XLS	Tabelle1	E1		2739,00	
DETEKTIV.XLS	Tabelle1	B3		#NAME?	=B1-B2
DETEKTIV.XLS	Tabelle2	B13		2050,00	=SUMME(B1:B12)
DETEKTIV.XLS	Tabelle2	C11		1074,00	=MITTELWERT(B13;B10)

Um eine zu überwachende Zelle hinzuzufügen, klicken Sie mit der rechten Maustaste auf die Zelle und wählen ÜBERWACHUNG HINZUFÜGEN... .



Sie können das Überwachungsfenster auch bei fehlerfreien Tabellen sinnvoll einsetzen: Scrollen Sie häufig in einer Tabelle, wollen aber ständig bestimmte Werte im Blick haben, vielleicht sogar Werte, die sich in einer anderen Tabelle befinden, so fügen Sie diese einfach in das Überwachungsfenster ein.

Formelüberwachungsmodus

Der mit EXTRAS|FORMELÜBERWACHUNG|FORMELÜBERWACHUNGSMODUS aktivierte Formelüberwachungsmodus verschafft Ihnen einen schnellen Überblick darüber, wo welche Formeln in Ihrer Tabelle enthalten sind.

Ist er aktiviert (erneute Anwahl deaktiviert ihn), so werden in der Tabelle die Formeln selbst angezeigt statt der aktuellen Werte dieser Formeln.

Formelauswertung

Das mit EXTRAS|FORMELÜBERWACHUNG|FORMELAUSWERTUNG eingeblendete Formelauswertungsfenster hilft dabei, den aktuellen Wert einer Formel zu ermitteln und festzustellen, wie er zu Stande kommt.

	A	B	C
1			Quartal
2	Jan	173	
3	Feb	94	
4	Mrz	184	=SUMME(B2:B4)
5			
6	Apr	193	
7	Mai	152	
8	Jun	152	=SUMME(B6:B8)
9			
10	Jul	194	
11	Aug	281	
12	Sep	182	=SUMME(B10:B12)
13			
14	Okt	98	
15	Nov	163	
16	Dez	184	=SUMME(B14:B16)
17			
18	Gesamt		= C4 + C8 + C12 + C16

Abbildung 13:
Der Formelüber-
wachungsmodus

Setzen Sie den Cursor in eine Zelle und klicken Sie das abgebildete Symbol an, so wird das Formelauswertungsfenster geöffnet und zeigt die darin enthaltene Formel an, die Sie nun Schritt für Schritt auswerten können.



	A	B	C
1			Quartal
2	Jan	173	
3	Feb	94	
4	Mrz	184	451
5			
6	Apr	193	
7	Mai	152	
8	Jun	152	497
9			
10	Jul	194	
11	Aug	281	
12	Sep	182	657
13			
14	Okt	98	
15	Nov	163	
16	Dez	184	445
17			
18	Gesamt		2050

Formel auswerten

Bezug: Tabelle0!\$C\$18

Auswertung: = C4+C8+C12+C16

Um das Ergebnis des unterstrichenen Ausdruckes anzuzeigen, klicken Sie auf Auswerten. Das aktuellste Ergebnis wird unterstrichen dargestellt.

Abbildung 14:
Formelauswertung

Passwortgeschützte Bereichsfreigabe in geschützten Blättern

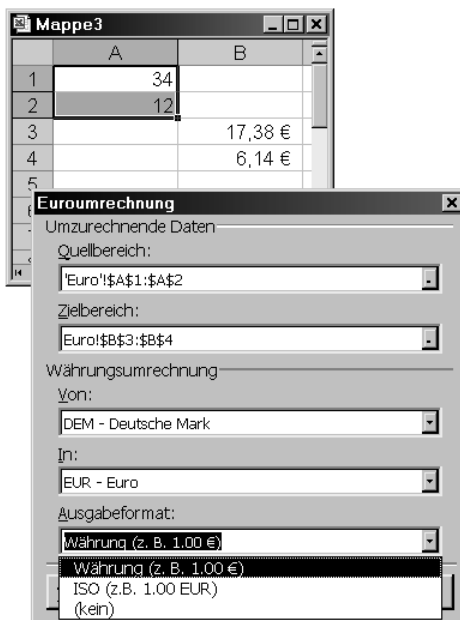
Mit EXTRAS|SCHUTZ|BENUTZER DÜRFEN BEREICHE BEARBEITEN... können Sie Bereiche einer Tabelle zum Editieren freigeben. Nur Benutzer, die das dem betreffenden Bereich zugehörige Passwort kennen, können diesen nach dem Schützen des Blattes editieren.

Euro-Währungen umrechnen

Das Euro-Währungen-Add-In vereinfacht Umrechnungen einer Euro-Währung in eine andere Euro-Währung (anhand der feststehenden Wechselkurse dieser Währungen zueinander).

Nach der Einbindung dieses Add-Ins erscheint eine Symbolleiste mit einer Liste aller Euro-Währungen, die in EUR konvertiert werden können (»xxx -> EUR«) und aller Währungen, in die umgekehrt gemäß der feststehenden Wechselkurse der Euro konvertiert werden kann (»EUR -> xxx«). Um eine einzelne Zelle oder einen selektierten Bereich umzurechnen, klicken Sie einfach auf das entsprechende Symbol.

Abbildung 15:
Euro-Währungen



Klicken Sie beispielsweise auf »DEM -> EUR«, wird davon ausgegangen, dass die Zellen im selektierten Bereich als DM zu interpretieren sind und die Summe dieser DM-Werte in den entsprechenden Euro-Betrag umgerechnet und in der Symbolleiste angezeigt werden soll. Zusätzlich fügt dieses Add-In den Befehl EXTRAS|EUROUMRECHNUNG... ein, der ein Dialogfeld öffnet, das die einfache Umrechnung aller Werte in einem anzugebenden Bereich ermöglicht. Dazu sind folgende Angaben notwendig:

**Weitere
Möglichkeiten**

- ➔ **Quellbereich:** Der Bereich mit den zu konvertierenden Werten, beispielsweise A1:B3
- ➔ **Zielbereich:** Der Bereich (oder dessen obere linke Ecke), in dem die umgerechneten Werte gespeichert werden sollen (Überschreiben des Quellbereichs ist nicht zulässig)

- ➔ *Von:* Die im Quellbereich verwendete Währung, beispielsweise »DEM – Deutsche Mark«
- ➔ *In:* Die Währung, in die jene Werte umgerechnet werden sollen, beispielsweise »EUR – Euro« oder »FRP – Französische Franc«
- ➔ *Ausgabeformat:* Das im Zielbereich zu verwendende Währungsformat

OK speichert die konvertierten Werte im Zielbereich.

Die Schaltfläche Weitere öffnet ein Dialogfeld, in dem beispielsweise die Umrechnungsgenauigkeit bestimmt oder festgelegt werden kann, dass die umgerechneten Werte im Zielbereich mit den Werten im Quellbereich verknüpft und somit bei Änderungen der Originalwerte automatisch aktualisiert werden.



Datenverwaltung und Datenanalyse

Datenübergabe von Query an Excel

Bei der Übergabe externer Daten aus Query an Excel können Sie nun im letzten Moment die Übergabe noch mit *Abfrage bearbeiten...* abbrechen, um die Daten in Query noch einmal zu editieren.

Vor allem aber können Sie mit *PivotTable Bericht erstellen...* den Pivot-Tabellen-Assistenten aufrufen, um aus den Daten eine Pivot-Tabelle zu erzeugen.



Abbildung 16:
Dialogfeld zum
Datenimport

Datenquellen öffnen und definieren

DATEN|EXTERNE DATEN IMPORTIEREN|DATEN IMPORTIEREN... ermöglicht das Öffnen bzw. Definieren unterschiedlichster Datenquellen:

- ➔ Gespeicherte Query-Abfragen (Endung DQY)
- ➔ Gespeicherte Webabfragen (Endung IQY)

Was ist neu – und wo finden Sie es?

- ➔ Bereits definierte Datenquellen (Endung DSN)
- ➔ Neue Datenquellen (Endung ODC) – mithilfe des zugehörigen Assistenten

Pivot-Tabellen: Felder einfügen

Die Pivot-Tabellen-Symbolleiste enthält nun keine Felder mehr. Die sind statt dessen nun in einer eigenen Feldliste enthalten, die zusätzliche Möglichkeiten zum Einfügen von Feldern in die verschiedenen Bereiche einer Pivot-Tabelle bietet.

Datenaustausch per Drag&Drop

Nach dem Einfügen von Excel-Daten in ein anderes Office-Programm wie Word erscheint ein Einfügen-Smart Tag, mit dem Sie das Format festlegen können, in dem die Daten eingefügt werden: egal ob nur reiner Text eingefügt wird oder das Originalformat der Daten erhalten bleibt oder die Formatierung einer Word-Tabelle angepasst wird, in die Sie die Daten eventuell einfügen, oder ob Sie das Format nun individuell festlegen wollen.

Zusätzlich gibt es eine Option, um die eingefügten Daten mit der Excel-Tabelle zu verknüpfen, damit sie bei Änderungen dieser Tabelle automatisch entsprechend aktualisiert werden.

Office-Zwischenablage

Abbildung 17:
Office-Zwischen-
ablage



Die Office-Zwischenablage fasst nun 24 Dokumente und wird durch das Register *Zwischenablage* des Aufgabenbereichs dargestellt, das Sie durch zweifaches Drücken von **[Strg]+[C]** oder mit dem Befehl **BEARBEITEN|OFFICE-ZWISCHENABLAGE...** einblenden.

Verknüpfte Objekte

Der Befehl **BEARBEITEN|VERKNÜPFUNGEN...** enthält einige zusätzliche Optionen zum Umgang mit Verknüpfungen, die unter anderem das Aktualisierungsverhalten beim Öffnen einer Mappe steuern, die externe Verknüpfungen enthält.

Grafikobjekte, Landkarten und Diagramme

Objekte rotieren

Durch Ziehen am grünen »Henkel« oberhalb eines aktivierten Objekts können Sie es drehen.

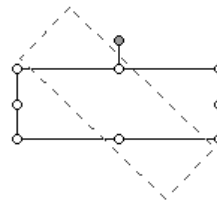


Abbildung 18:
Objekt drehen

Ist das eingefügte Bild selektiert, können Sie es am Henkel drehen, es verkleinern, vergrößern etc. Darüber hinaus ermöglicht Ihnen jedoch die automatisch eingeblendete *Grafik-Symbolleiste* noch weitere Bildmanipulationen.

Bilder manipulieren

Ist ein eingefügtes Bild selektiert, so können Sie mit der automatisch eingeblendeten *Grafik-Symbolleiste* die Farbanzahl verändern, es beispielsweise auf Graustufen oder Schwarzweiß-Darstellung reduzieren, den Kontrast und die Helligkeit verändern, es zuschneiden, komprimieren und eine Farbe auswählen, die transparent sein soll.

ClipArt

Der Umgang mit ClipArt hat sich deutlich verändert, da es mehr und andere Möglichkeiten bietet als zuvor und nun über den ebenfalls neuen Aufgabenbereich gesteuert wird.

Diagrammelemente selektieren

Die Diagramm-Symboleiste enthält ein Listefeld mit allen Elementen eines Diagramms. Wählen Sie eines davon aus, so wird es selektiert, als hätten Sie es mit der Maus angeklickt.

Datenreihen/-punkte beschriften

Datenpunkte können nun wahlweise mit den Rubriken- und/oder den Datenreihennamen beschriftet werden. Bei Verwendung mehrerer Beschriftungen kann das Trennzeichen bestimmt werden. Es kann außerdem festgelegt werden, dass die Beschriftungen untereinander in separaten Reihen erscheinen.

Excel und das Internet

Hyperlinks einfügen

Das Dialogfeld zum Einfügen und Editieren von Hyperlinks hat sich deutlich geändert. Vor allem das Einfügen von Hyperlinks auf Webseiten ist erheblich vereinfacht worden.

Webordner: Erstellen und Verknüpfungen

Der Assistent zum Erstellen neuer Webordner bzw. von Verknüpfungen zu bereits existierenden Webordnern hat sich ein wenig verändert.

Als Webarchiv speichern

Das Dateiformat »Webarchiv« ermöglicht die Speicherung einer Mappe als Webseite ohne separate Speicherung gegebenenfalls unzähliger Hilfsdateien, setzt zur Betrachtung jedoch den Internet Explorer ab Version 4.0 voraus.

XML-Format

Excel kann Webseiten in XML-Dateien öffnen bzw. selbst Mappen im XML-Format speichern.

Um Excel-Mappen als Webseiten zu speichern, sollten Sie daher entweder den einfachsten Standard verwenden, der die geringsten Anforderungen auf Betrachterseite stellt, also HTML – oder gleich den jüngeren Standard XML, der jedoch geeignete moderne Webbrowser beim Betrachter voraussetzt.

Interaktive Webseiten erstellen

Die zusätzliche Option *Automatisch Wiederveröffentlichen...* publiziert jedesmal eine Mappe, wenn Sie sie speichern, erneut mit den aktuellen Einstellungen als Webseite, sodass jene immer aktuell ist, auch nach – gespeicherten – Änderungen der Excel-Mappe.

Webabfragen erstellen und editieren

Das Erstellen und Editieren von Webabfragen wurde deutlich vereinfacht, da Sie nun direkt in der betreffenden Webseite die zu importierenden Abschnitt auswählen können.

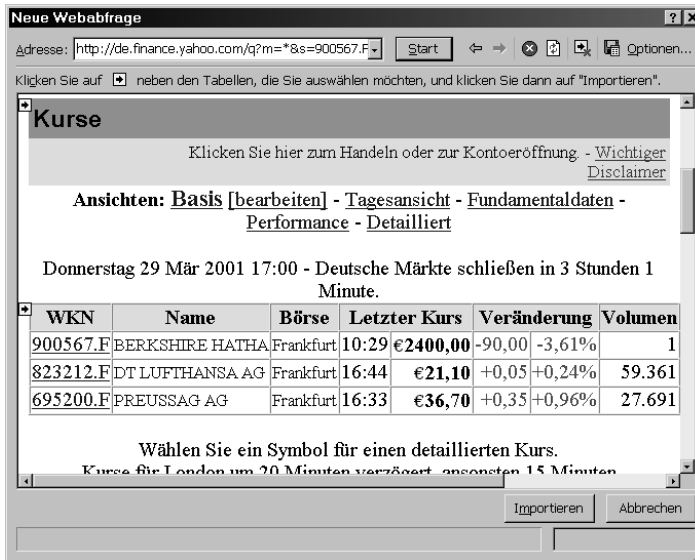


Abbildung 19:
Webabfrage
erstellen/editieren

Aktienkurs-Smart Tag

Angenommen, in Ihrem Webbrowser wird eine Webseite mit Kursen amerikanischer Aktien angezeigt. Sie markieren die interessierenden Kurse, befördern Sie mit BEARBEITEN|KOPIEREN in die Zwischenablage und fügen sie anschließend mit BEARBEITEN|EINFÜGEN in die Excel-Tabelle ein.

Excel erkennt die im eingefügten Bereich enthaltenen amerikanischen Wertpapiersymbole wie »MSFT« oder »KO« und blendet das Aktienkurs-Smart Tag ein.

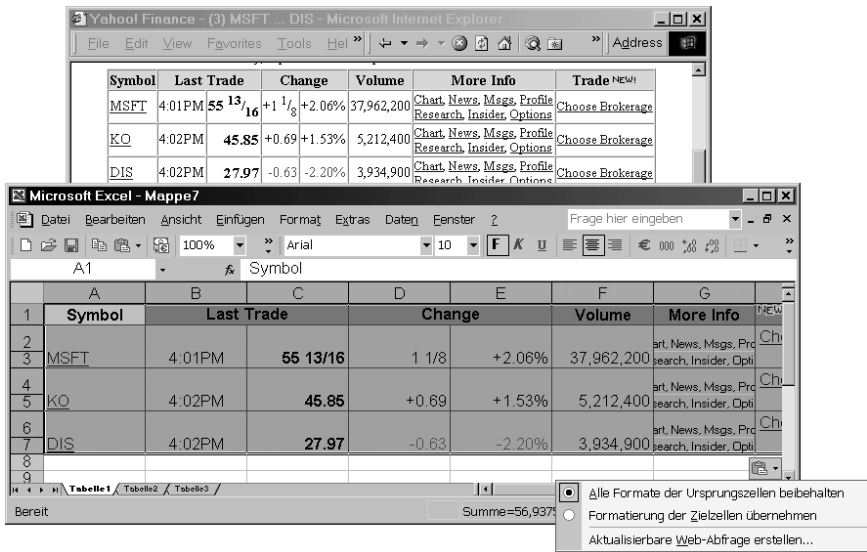
AKTUALISIERBARE WEB-ABFRAGE ERSTELLEN... öffnet das Dialogfeld zum Importieren von Daten aus Webseiten, in dem Sie die zu importierenden Abschnitte auswählen. Excel erstellt anschließend eine Webabfrage, die die ausgewählte(n) Tabelle(n) importiert und von Ihnen jederzeit aktualisiert werden kann.

Weboptionen

Excel verfügt über weitaus vielfältigere Weboptionen als zuvor (Register Allgemein des Befehls EXTRAS|OPTIONEN... bzw. DATEI|SPEICHERN wählen, im Dateiauswahl-Dialogfeld das Extras-Listenfeld öffnen und WEB-OPTIONEN... wählen).

Was ist neu – und wo finden Sie es?

Abbildung 20:
Aktienkurs-
Smart Tag



Bilder komprimieren

Wenn Sie DATEI|SPEICHERN wählen, im *Dateiauswahl*-Dialogfeld das *Extras*-Listenfeld öffnen und BILD KOMPRIMIEREN... wählen, können Sie detailliert festlegen, ob und wie Bilder komprimiert werden sollen; das ist eine sinnvolle Option beim Speichern von Tabellen als Webseiten im Internet, um zu große Webseiten mit entsprechend langwierigem Download zu vermeiden.

Das gleiche Dialogfeld erscheint, wenn ein Bild selektiert ist und Sie in der *Grafik*-Symbolleiste auf das Symbol zum Komprimieren von Grafiken klicken.

Sonstiges

Aktivierung von Excel bzw. Office

Nach dem fünfzigsten Aufruf (dazu zählt auch jeder Aufruf eines anderen Office-Programms) läuft Excel nur noch mit eingeschränkter Funktionalität und Sie können beispielsweise keine Dokumente speichern. Sie verfügen somit nur noch über eine Art Demoversion, die zur echten Arbeit untauglich ist.

Um das zu ändern, müssen Sie Excel mit dem nach dem Start automatisch erscheinenden Assistenten (den Sie auch später jederzeit mit ?|PRODUKT AKTIVIEREN... aufrufen können) per Telefon oder übers Internet aktivieren.

Dateisuche

Die Dateisuche hat sich geändert. Sie ermöglicht nun eine gezielte Auswahl des Suchorts und ist einfacher zu bedienen, da nach dem Öffnen des zugehörigen Dialogfelds zunächst ein Register aktiviert ist, das nur die häufig benötigte Suche nach bestimmten in Dateien enthaltenen Texten ermöglicht, die verwirrende Anzahl an Suchoptionen des vollständigen Registers jedoch unterschlägt.



Rechtschreibprüfung

Die Rechtschreibprüfung hat sich ebenfalls geändert und ermöglicht nun unter anderem die Festlegung landesspezifischer Optionen und die direkte Anwahl des AutoKorrektur-Dialogfelds.

AutoKorrektur

Das AutoKorrektur-Dialogfeld besitzt zusätzliche Register, die Ihnen unter anderem ermöglichen, bestimmten Daten wie E-mail-Adressen automatisch Smart Tag anheften zu lassen, die den schnellen Zugriff auf ein entsprechendes Kontextmenü ermöglichen.

Add-In-Manager: COM-Add-Ins einbinden

Die Schaltfläche *Automation* des Add-In-Managers öffnet ein Dialogfeld, in dem alle COM-Add-Ins aufgelistet werden, die eingebunden werden können, und das selbst wiederum eine *Durchsuchen*-Schaltfläche besitzt, mit der nach weiteren COM-Add-Ins gefahndet werden kann.

Zusätzliche Optionen

Aufgrund der neuen Möglichkeiten wurde auch das Dialogfeld des Befehls EXTRAS|OPTIONEN... entsprechend erweitert. Die zuvor vorhandenen Register änderten sich nur minimal, dafür sind jedoch einige neue Register hinzugekommen (die teilweise jedoch nur Kopien der von anderen Befehlen geöffneten Dialogfelder darstellen).

Teil 1 Bedienung

Der erste Teil des Buches beschäftigt sich mit der Bedienung von Excel.

Nach einer Kurzerläuterung der Installation geht es um Oberflächenelemente wie Menüs, Symbolleisten und Assistenten. Ich erläutere beispielsweise, wie Sie die Excel-Symbolleisten individuell anpassen können.

Danach erkläre ich, wie in Excel mit Fenstern, vor allem mit mehreren Fenstern gleichzeitig, gearbeitet wird, wie Arbeitsmappen benutzt werden und welche spezifischen Features die Excel-Hilfestellung besitzt.

1 Benutzeroberfläche

Im folgenden Kapitel lernen Sie den Umgang mit der Oberfläche von Excel. Der Schwerpunkt liegt dabei nicht auf Standardelementen wie Menüs und Dialogfeldern, die Sie wahrscheinlich bereits aus anderen Windows-Programmen kennen, sondern auf Excel-spezifischen Bedienelementen wie den individuell konfigurierbaren Menü- und Symbolleisten, den Arbeitsmappen und dem Office-Assistenten.

Zunächst zeige ich Ihnen, wie Sie bei der Installation auswählen, welche Komponenten von Excel installiert werden, und aus welchen Elementen sich der Excel-Bildschirm zusammensetzt, mit dem Sie es von nun an zu tun haben.

Danach erläutere ich kurz die üblichen Windows-Elemente wie Menü, Symbolleiste, Dialogfelder und Fenster. Anschließend erfahren Sie, wie Sie die Menüs und Symbolleisten von Excel maßschneidern und an Ihre individuellen Bedürfnisse anpassen können.

Nach der Erläuterung der Arbeitsmappen, in denen all Ihre Kalkulationen, Grafiken und sonstigen Excel-Erzeugnisse gespeichert werden, geht es um vielfältige Hilfestellungen, die Ihnen Excel bietet. Ich erläutere

- ➔ die Funktionsweise der kontextsensitiven Hilfe, die Ihnen gezielt Hilfestellung zu aktuellen Problemen anbietet,
- ➔ den Umgang mit dem niedlichen Office-Assistenten, einem freundlichen Helfer, der immer dann auftaucht, wenn er es für angebracht hält (wenn es beispielsweise einen einfacheren Weg gibt als jenen, den Sie zuvor wählten)
- ➔ und das Hilfefenster, das sehr vielseitig ist und teilweise ebenfalls individuell angepasst werden kann, beispielsweise, indem Sie zu den vorgegebenen Hilfstexten eigene Kommentare hinzufügen.

1.1 Vor dem Start

Setzen wir voraus, dass Sie Windows installiert haben und nun entweder nur Excel installieren wollen oder gar Microsoft Office besitzen und installieren wollen, also das Microsoft-Gesamtpaket, das außer Excel auch Word, Powerpoint und gegebenenfalls (in der Professional-Version) Access enthält.

1.1.1 Installation

In jedem Fall müssen Sie nun das Installationsprogramm ausführen. Nach Einlegen der CD-ROM erscheint bei geeigneter Konfiguration Ihres PCs automatisch ein Menü, das Ihnen die Programminstallation anbietet.

Ohne diese Automatik müssen Sie das Programm SETUP.EXE suchen und ausführen, indem Sie in Windows START|AUSFÜHREN... wählen, auf *Durchsuchen...* klicken, die Datei SETUP.EXE suchen und dieses Programm per Klicken auf OK starten.

Zunächst werden eventuell eine ganze Menge neuer Systemdateien auf Ihrem Rechner installiert, abhängig vom Alter Ihrer Windows-Installation und erst danach Office selbst.

Die Installation selbst läuft weitestgehend automatisch ab und erfordert nur wenige Angaben wie Ihren Namen und den CD-Key, praktisch die Seriennummer Ihrer CD.

Anschließend dürfen Sie sich mit den Lizenzbedingungen einverstanden erklären (oder auch nicht, dann wird jedoch nichts aus der Installation) und danach eine der Installationsoptionen wählen (Abbildung 1.1).

Abbildung 1.1:
Installations-
optionen



Die vorgegebene Standardoption *Jetzt aktualisieren* installiert Office mit den gängigsten Programmen und Features (ist noch keine ältere Version von Office auf Ihrem Rechner installiert, heißt sie stattdessen *Jetzt installieren* und entfernt zuvor diese ältere Office-Version, die ja meist nicht mehr benötigt wird).

Standard installiert die meistverwendeten Programmteile, *Vollständig* installiert das komplette Office, und *Benutzerdefiniert* ermöglicht Ihnen, die zu installierenden Komponenten detailliert auszuwählen. Ist bereits eine Office-Version installiert, können Sie übrigens bei jeder dieser drei Optionen im übernächsten Schritt wählen, ob jene ältere Office-Programme vollständig oder teilweise entfernt oder aber unverändert behalten werden soll.

Vor dem nächsten Schritt können Sie bei Bedarf zuvor noch im unteren Teil des Fensters den vorgegebenen Installationspfad C:\PROGRAMME\MICROSOFT OFFICE ändern: entweder per Hand durch Eintippen oder aber per Anklicken von *Durchsuchen...* und Auswahl des gewünschten Verzeichnisses im folgenden Dialogfeld.

Abhängig von der gewählten Installationsoption können Sie gegebenenfalls im folgenden Schritt wählen, welche Office-Programme installiert werden sollen (Abbildung 1.2).

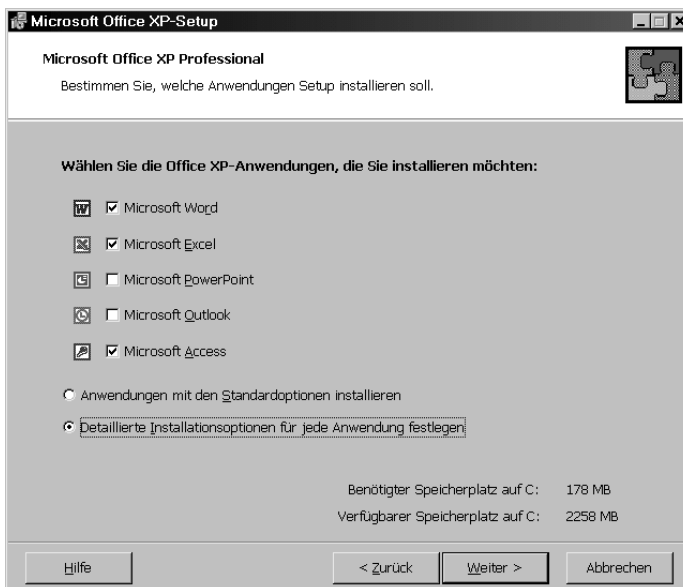


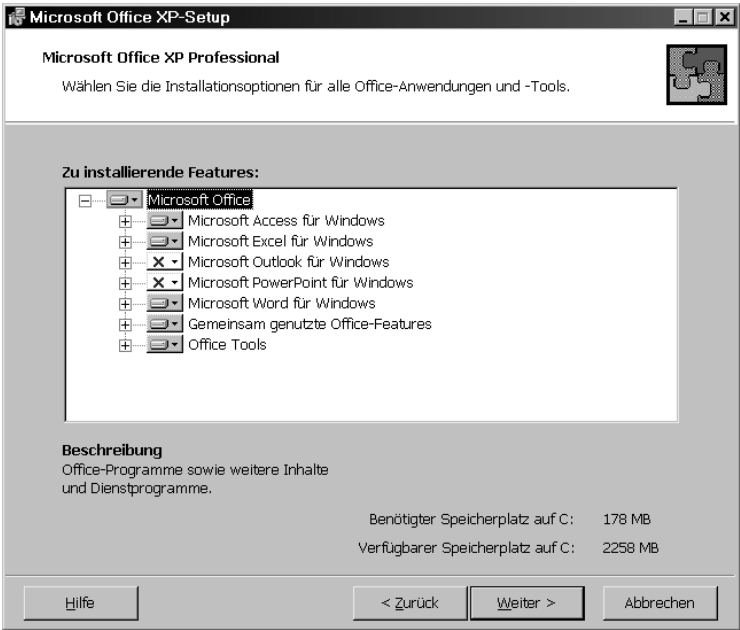
Abbildung 1.2:
Anwendungs-
auswahl

Die benutzerdefinierte Installation ist am vielseitigsten. Dabei wählen Sie zunächst aus, welche Office-Anwendungen installiert werden sollen, und danach – vorausgesetzt, Sie aktivieren die Option *Detaillierte Installationsoptionen für jede Anwendung festlegen* – bestimmen Sie, welche der einzelnen Komponenten dieser Programme installiert werden soll (Abbildung 1.3).

**Benutzer-
definierte
Installation**

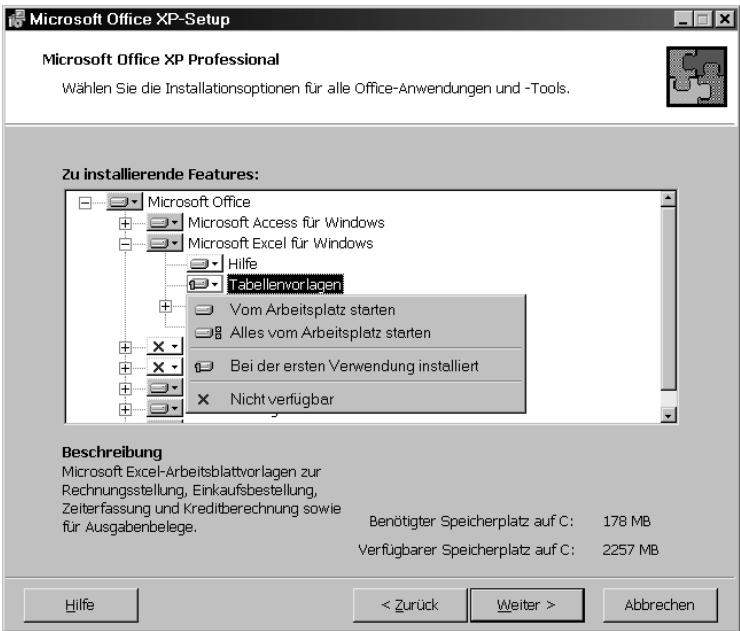
»+« öffnet den betreffenden »Komponentenzweig«, wobei gegebenenfalls mehrere Unterzweige sichtbar werden, die eigene »+«-Zeichen besitzen, mit denen Sie sie ebenfalls aufklappen können. »-« schließt den jeweiligen Zweig wieder.

Abbildung 1.3:
Komponenten-
auswahl



Klicken Sie bei einer Komponente auf den abwärts gerichteten Pfeil, so erscheint ein Auswahlménü (Abbildung 1.4).

Abbildung 1.4:
Optionen



- ➔ *Vom Arbeitsplatz starten:* Die betreffende Komponente wird auf Ihrer Festplatte installiert und bei jeder Benutzung direkt von dort gestartet (schnell, benötigt aber viel Speicherplatz auf der Festplatte).
- ➔ *Alles vom Arbeitsplatz starten:* Installiert die Komponente und alle darin enthaltenen Unterkomponenten auf Ihrer Festplatte.
- ➔ *Bei der ersten Verwendung installieren:* Die Komponente wird noch gar nicht installiert. Das erfolgt erst, wenn Sie sie zum ersten Mal benutzen.
- ➔ *Nicht verfügbar:* Die betreffende Komponente wird nicht installiert.

Installieren Sie Excel/Office über das Netzwerk, ist die CD also im CD-ROM-Laufwerk eines anderen Rechners eingelegt, der mit Ihrem verbunden ist, gibt es zusätzlich noch die Option, die Komponente so zu installieren, dass sie über das Netzwerk geladen wird.

Mit den Vorgaben werden nur die wichtigsten Excel-Komponenten installiert und nicht auch die weniger gebräuchlichen! Geben Sie sich damit zufrieden, werden Sie die Besprechung verschiedener Excel-Bestandteile in diesem Buch nicht nachvollziehen können, da die betreffenden Bestandteile nicht installiert wurden! Ich gehe davon aus, dass Excel vollständig installiert ist.

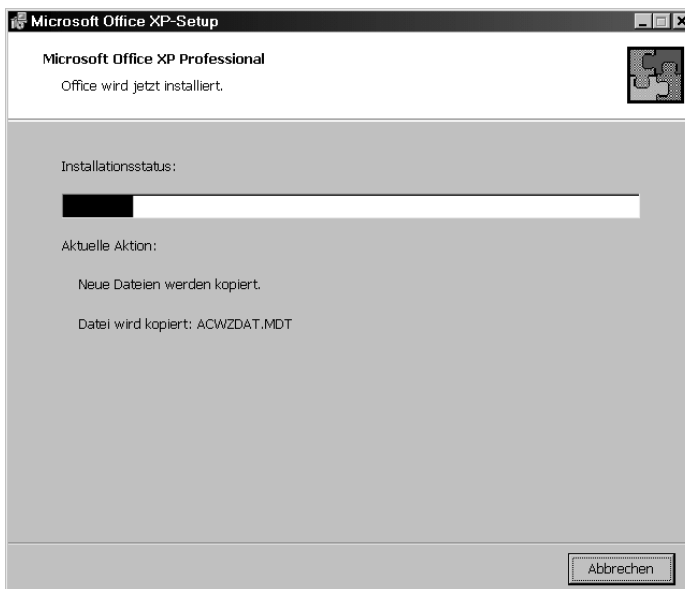


Abbildung 1.5:
Installationsprozess

Die Installation selbst dauert geraume Zeit, erst recht, wenn Sie nicht nur Excel selbst, sondern das gesamte Office-Paket installieren (Abbildung 1.5).



Sie können Ihre Office-Installation jederzeit nachträglich ändern, indem Sie im Windows-Ordner SYSTEMSTEUERUNG das Objekt Software öffnen und im daraufhin erscheinenden Dialogfeld auf den Microsoft Office-Eintrag doppelklicken.

Daraufhin wird das Installationsprogramm erneut gestartet und Sie können wählen, ob Sie Office reparieren, entfernen oder Komponenten hinzufügen/entfernen wollen. Letzteres öffnet den zuvor gezeigten Komponentenbaum, in dem Sie jede zu installierende Komponente einzeln auswählen können.

1.1.2 Aufruf

Nach erfolgter Installation gibt es verschiedene Möglichkeiten, Excel aufzurufen, je nachdem, ob Sie nur Excel installierten oder aber das gesamte Office-Paket. Fast immer läuft es jedoch auf einen einfachen oder einen doppelten Klick auf das typische Excel-Symbol hinaus (Abbildung 1.6).

Abbildung 1.6:
Das Excel-Symbol



- ➔ Die umständliche Methode: Sie suchen dieses Symbol mit dem Windows-Explorer auf Ihrer Festplatte. Das Symbol und die zugehörige Datei EXCEL.EXE (oder eine Verknüpfung zu dieser Datei) finden Sie in jenem Verzeichnis, in dem Excel installiert wurde. Das könnte das Verzeichnis EXCEL sein, wenn Sie nur Excel installierten, oder aber das Verzeichnis PROGRAMME\MICROSOFT OFFICE, wenn Sie diese automatische Vorgabe bei der Installation des Office-Pakets übernahmen); ein Doppelklick darauf ruft Excel auf.
- ➔ Die einfachere Technik: Sie finden dieses Symbol nach dem Klicken auf die Windows-Start-Schaltfläche unter der Rubrik *Programme*. Per einfachem Klick darauf wird Excel aufgerufen.

Für Excel-Neulinge weniger geeignet: Installierten Sie Office, dann können Sie Excel auch durch Anklicken des Eintrags *Neues Office-Dokument* starten, der durch Klicken auf die *Start*-Schaltfläche sichtbar wird. Allerdings müssen Sie anschließend in einem Dialogfeld erst noch den Eintrag *Leere Arbeitsmappe* selektieren und danach auf OK klicken.

Analog dazu können Sie auch auf *Office-Dokument öffnen* klicken und im folgenden Dialogfeld eine bereits existierende Excel-Datei auswählen, um Excel aufzurufen und gleichzeitig diese Datei zu öffnen.



Sollte Excel irgendwann einmal nicht mehr so wollen wie Sie, sind möglicherweise irgendwelche Bestandteile (Dateien, Einträge in die Windows-Registry etc.) beschädigt oder gelöscht worden. In diesem Fall ermöglicht ?|ERKENNEN UND REPARIEREN... Excel selbst nach derartigen Fehlern suchen und sie gegebenenfalls mithilfe der Installations-CD automatisch reparieren zu lassen, ohne alles neu installieren zu müssen.

1.1.3 Aktivierung

Nach dem Aufruf von Excel erscheint mit kurzer Verzögerung der Aktivierungs-Assistent (Abbildung 1.7).

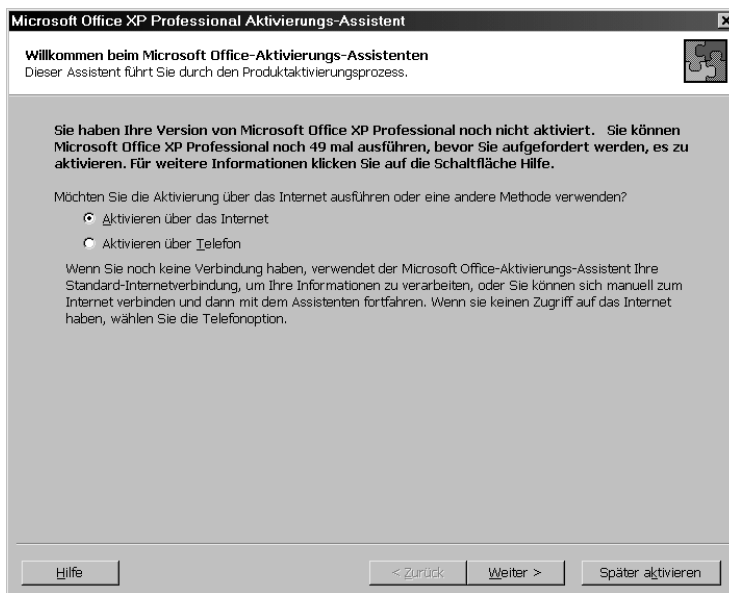


Abbildung 1.7:
Aktivierungs-
Assistent

Klicken Sie auf *Später aktivieren*, so wird Ihnen mitgeteilt, wie oft Sie Excel noch starten können. Klicken Sie danach auf *Beenden*, erscheint Excel und Sie können loslegen.

Was passiert, wenn Sie das jedes Mal so machen, Excel also nie aktivieren? Dann läuft es nach dem fünfzigsten Aufruf (dazu zählt auch jeder Aufruf eines anderen Office-Programms) nur noch mit eingeschränkter Funktionalität und Sie können beispielsweise keine Dokumente speichern. Sie verfügen somit nur noch über eine Art Demoversion, die zur echten Arbeit untauglich ist.

Irgendwann bleibt Ihnen daher keine andere Wahl, als Excel mit Hilfe des Aktivierungs-Assistenten zu aktivieren. Er erscheint wie gesagt automatisch nach dem Aufruf, Sie können ihn aber auch jederzeit während der Arbeit mit dem Befehl ?|PRODUKT AKTIVIEREN... starten.

Per Internet Zunächst wählen Sie (auf das betreffende runde Optionskästchen klicken), ob Sie per Telefon oder per Internet aktivieren wollen. Wählen Sie die Aktivierung über das Internet, wird im folgenden Schritt Ihr Webbrowser gestartet und die Aktivierung erfolgt nahezu vollautomatisch. »Nahezu«, da Sie aus einer Liste das Land auswählen müssen, in dem Sie wohnen (weitere persönliche Informationen können Sie angeben, Sie müssen es jedoch nicht).

Per Telefon Wählen Sie die Aktivierung per Telefon, ist der Aufwand deutlich größer: Klicken Sie zweimal nacheinander auf *Weiter >*, erscheint ein Dialogfeld mit einem Listenfeld (siehe Kapitel 1.2.3, »Dialogfelder«), in dem Sie Ihr Land auswählen müssen. Nach Auswahl von »Deutschland« wird eine Telefonnummer angezeigt (Abbildung 1.8).

Abbildung 1.8:
Aktivierungs-
informationen

Sie werden mit einem Angestellten von Microsoft verbunden und nach im Dialogfeld angezeigten Informationen gefragt. Nach Nennung dieser Informationen gibt er Ihnen einen Freischaltcode, den Sie im unteren Feld eingeben müssen, um den Aktivierungsprozess abzuschließen.



Der Freischaltcode ist an die Hardware des Rechners gebunden. Das heißt, dass nach größeren Änderungen der Rechner-Hardware (neue Hauptplatine etc.) eine erneute Aktivierung mit einem neuen Freischaltcode notwendig ist. Geplant ist von Microsoft, dass dies zwei weitere Male möglich ist, ab dem dritten Mal jedoch nur in Ausnahmefällen ein weiterer Freischaltcode mitgeteilt wird.

1.1.4 Komponenten des Excel-Fensters

Nach dem Aufruf befindet sich am oberen Bildschirmrand die Titelleiste, die den Namen der Windows-Applikation enthält, mit der Sie gerade arbeiten, in unserem Fall »Microsoft Excel« (Abbildung 1.9).

Titelleiste

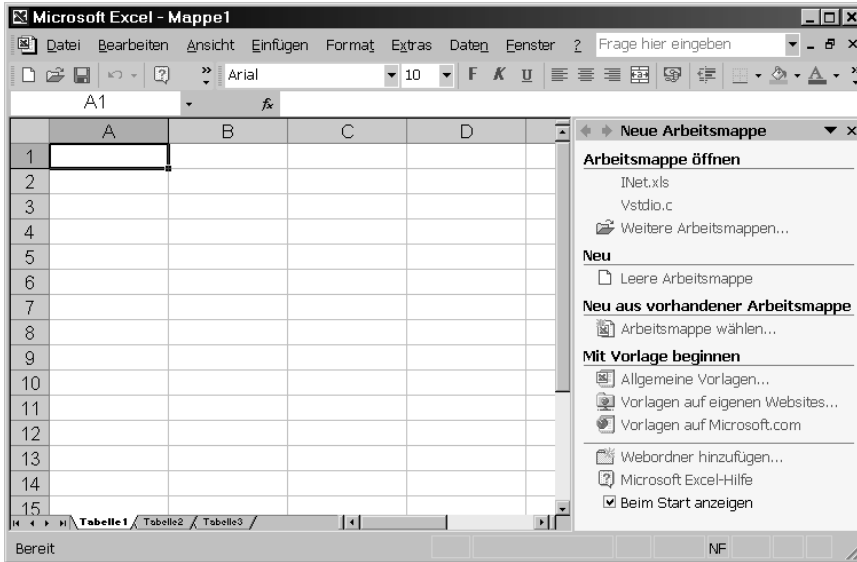


Abbildung 1.9:
Nach dem Aufruf

Darunter befindet sich die Menüleiste mit den Namen der verschiedenen Auswahlmenüs, die Ihnen Excel zur Verfügung stellt, und mit dem Feld zum Anfordern von Hilfe durch das Eingeben einer entsprechenden Frage.

*Menü-und
Symbolleiste*

Darunter befindet sich eine Zeile mit Symbolleisten. Die darin enthaltenen Symbole oder Knöpfe vereinfachen die Auswahl vieler Menübefehle; sie werden per Anklicken mit der Maus aktiviert und führen dann den zugehörigen Excel-Befehl direkt aus, ohne ihn erst in den Menüs finden zu müssen. Beispielsweise genügt ein Mausklick auf das jeweils zuständige Symbol, um eine Grafik zu erzeugen, Zahlen zu addieren oder Texte fett hervorzuheben.

Den rechten Teil des Bildschirm nimmt ebenfalls eine Art Symbolleiste ein, der Aufgabenbereich. Er erscheint automatisch, wenn Sie bestimmte Befehle wählen und sein Inhalt unterscheidet sich dann je nach aktueller Aufgabenstellung. Ich empfehle Ihnen, den sehr viel Bildschirmfläche verschlingenden Aufgabenbereich zu schließen, indem Sie auf das Kreuz in ihrer rechten oberen Ecke klicken, bis wir zur Besprechung jener Befehle kommen.

Aufgabenbereich

Unter den Symbolleisten befindet sich die Bearbeitungsleiste mit Informationen zu dem Teil des Arbeitsblatts, den Sie gerade bearbeiten.

*Eingabe-/Status-
leiste*

Die Statusleiste am unteren Bildschirmrand enthält je nach Situation unterschiedliche Mitteilungen. Im rechten Teil der Statusleiste zeigt Excel an, welche speziellen Tastenfunktionen gerade aktiviert sind. Beispielsweise weist *NF* darauf hin, dass der numerische Ziffernblock eingeschaltet ist, den Sie mit der Taste  ein- oder ausschalten können.

Tabellenfenster

Das wichtigste Element ist das leere Tabellenblatt oder Tabellenfenster, das fast den ganzen Bildschirm einnimmt. Es ist durch Gitternetzlinien in Spalten und Zeilen unterteilt. Die einzelnen Spalten werden durch Buchstaben (A, B, C usw.) und die Zeilen durch Zahlen (1, 2, 3 usw.) gekennzeichnet.

Zellen

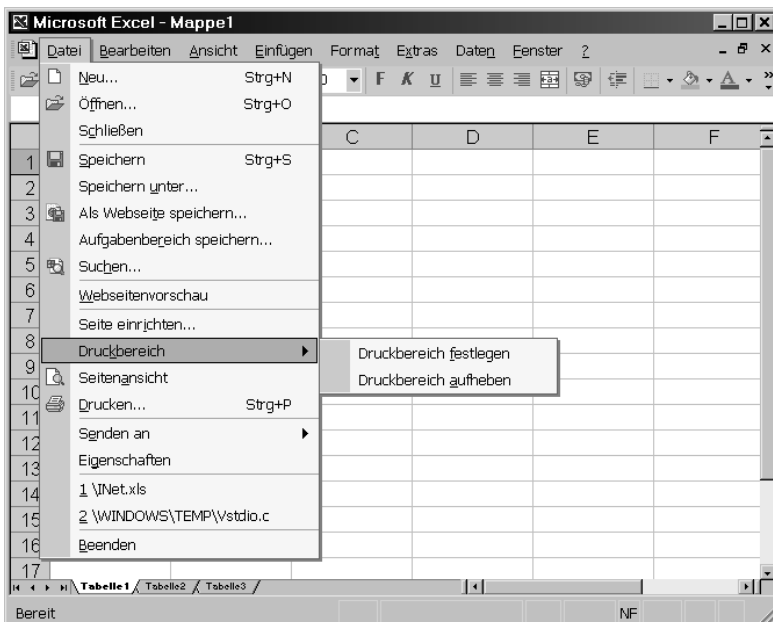
Die kleinen Blöcke, aus denen die Tabelle besteht, sind die Zellen oder Felder. Jede Zelle kann eine bestimmte Information aufnehmen, eine Zahl (123,45), eine Zeichenfolge (»Hallo«), ein Datum (15.11.59) oder eine Formel ($=2*3$).

1.2 Menüs, Symbolleisten und Dialogfelder

1.2.1 Menüs

Excel besitzt mehrere Menüleisten und blendet die je nach Arbeitssituation geeignete automatisch ein. Um ein Menü einer solchen Menüleiste zu öffnen, klicken Sie auf den zugehörigen Menünamen, beispielsweise *DATEI* (Abbildung 1.10).

Abbildung 1.10:
Das Menü »Datei«



Anschließend genügt es, den Cursor nach unten/oben zu bewegen, um darin einen Befehl zu selektieren (er wird hervorgehoben). Bewegen Sie ihn stattdessen nach rechts/links, wird das entsprechend benachbarte Menü geöffnet, beispielsweise BEARBEITEN.

Um einen Befehl auszuführen, klicken Sie einfach darauf (oder selektieren Sie ihn mit den Cursortasten und drücken ).

Nicht zu jedem Zeitpunkt sind Befehle eines Menüs auch wirklich verfügbar, das heißt anwählbar. Auf momentan nicht wählbare Menübefehle weist Excel Sie durch eine abgeblendete Darstellung des Befehls hin.

Ein nach rechts weisendes Dreieck weist darauf hin, dass sich beim Verweilen des Cursors auf dem betreffenden Befehl ein Untermenü öffnet, aus dem Sie nun den letztlich interessierenden Befehl per Anklicken auswählen können.

Zur Benennung von Befehlen verwende ich folgende Konvention:

MENÜNAME|BEFEHLSNAME|UNTERBEFEHLSNAME

DATEI|NEU... kennzeichnet somit den Befehl NEU... im Menü DATEI und DATEI|DRUCKBEREICH|DRUCKBEREICH AUFHEBEN den Befehl DRUCKBEREICH AUFHEBEN, einen Unterbefehl des Befehls DRUCKBEREICH, der sich im Menü DATEI befindet.

Taste	Funktion
 oder 	Menüleiste aktivieren
 +Buchstabe	Menü direkt selektieren
Buchstabe	Befehl direkt selektieren
 / 	Rechtes/linkes Menü selektieren
 / 	Unteren/oberen Befehl selektieren
	Selektierten Befehl ausführen
	Menüleiste deaktivieren (zweimal drücken, falls ein Menü aufgeklappt ist)
 +Buchstabe	Direktausführung des zugehörigen Befehls bei deaktivierter Menüleiste

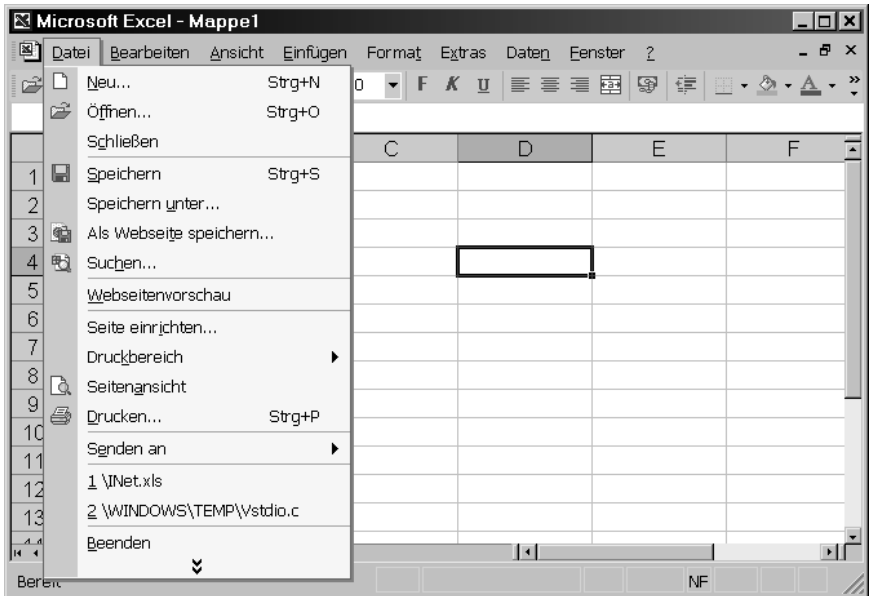
Tabelle 1.1:
Tastenfunktionen
zur Menübedienung



Excel zwingt Sie niemals, einen Befehl auszuführen, wenn Sie es nicht wollen. Sie können die Befehlswahl jederzeit mit [ESC] abbrechen, um ein geöffnetes Menü zu schließen und die Menüleiste wieder zu deaktivieren! Mit der Maus genügt es, auf irgendeine Zelle der Tabelle zu klicken.

Excel zeigt nach dem Öffnen eines Menüs zunächst nur die nach der Vorstellung von Microsoft wichtigsten Befehle an. Öffnen Sie das Menü DATEI, werden alle anderen Befehle zunächst nicht angezeigt (Abbildung 1.11).

Abbildung 1.11:
Anzeige nur der wichtigsten Befehle



Erst nach kurzer Verzögerungszeit oder wenn Sie auf den Pfeil am unteren Ende klicken, werden auch die restlichen Befehle angezeigt.



In Kapitel 38.1, »Menü- und Symbolleisten anpassen«, erläutere ich, wie Sie das Menüverhalten steuern können.

Kontextsensitive Menüs

Außer den Menüs der Menüleiste gibt es zusätzliche kontextsensitive Menüs. Um sie zu öffnen, klicken Sie ein Excel-Objekt mit der *rechten* statt wie bisher mit der linken Maustaste an, zum Beispiel irgendeine Zelle des Arbeitsblatts (Abbildung 1.12).

Die betreffende Zelle wird markiert. Daneben erscheint ein Kontextmenü, das die häufigsten zur Manipulation von Zellen verwendeten Befehle enthält (die übrigens fast alle auch im BEARBEITEN-Menü enthalten sind).

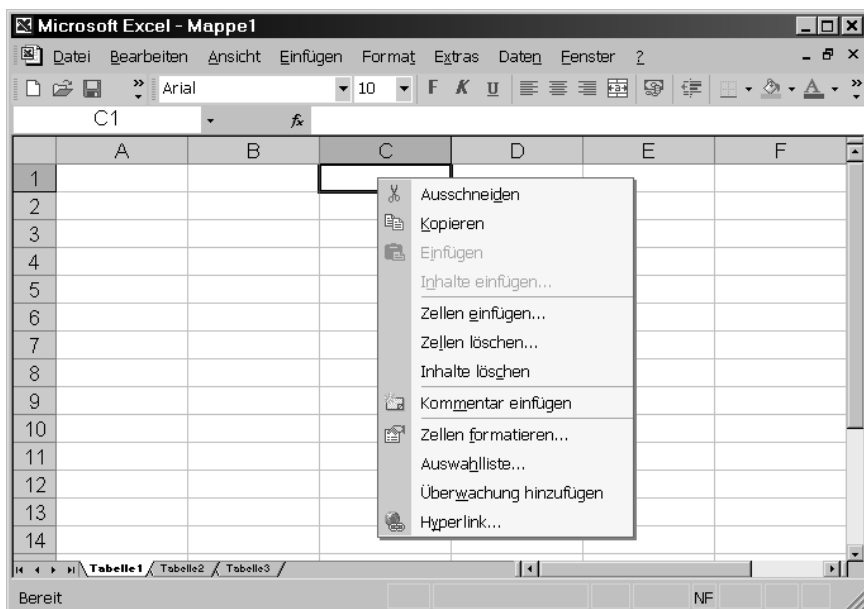


Abbildung 1.12:
Zellen-Kontextmenü

1.2.2 Symbolleisten

Das Wichtigste vorweg: Menübefehle und Symbole sind für Excel nahezu identisch. Dass diese Kommandoknöpfechen im einen Fall mit Text beschriftet sind, im anderen mit einem Bildchen, teilweise auch mit beidem gleichzeitig (siehe Befehl DATEI|NEU...) ist ein rein optischer Unterschied und kann sehr leicht von Ihnen geändert werden. Die folgenden Manipulationstechniken können somit auf all diese Knöpfechen in gleicher Weise angewendet werden!



Symbolleisten vereinfachen die Bedienung von Excel, da für die am häufigsten benutzten Menübefehle Symbole zur Verfügung stehen, die durch einfaches Anklicken aktiviert werden.

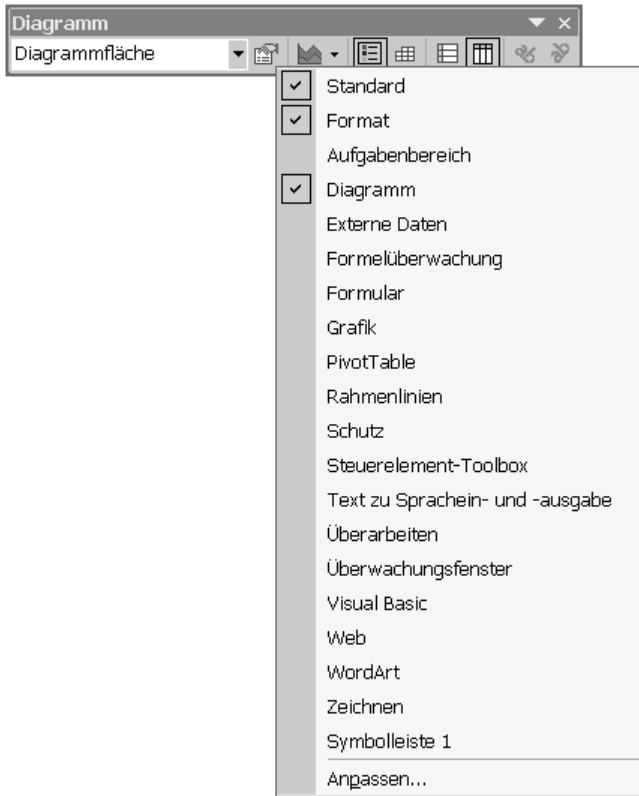
Excel besitzt verschiedene Symbolleisten für unterschiedliche Arbeitssituationen, die automatisch eingeblendet werden. Sind Sie gerade dabei, eine Grafik zu erzeugen, blendet Excel beispielsweise die Diagramm-Leiste ein, mit der Sie unter anderem per Klick ein Balken- in ein Tortendiagramm verwandeln können (Abbildung 1.13).



Abbildung 1.13:
Die Diagramm-
Leiste

Wählen Sie den Befehl ANSICHT|SYMBOLLEISTEN oder klicken Sie eine Symbolleiste an einem beliebigen Punkt mit der *rechten* Maustaste an, erscheint folgendes Menü (Abbildung 1.14).

Abbildung 1.14:
Das Symbolleisten-
Menü



**Leiste ein-/
ausblenden**

Eingeblendete Leisten sind durch Häkchen markiert. Klicken Sie den betreffenden Eintrag an, wird die Leiste wieder ausgeblendet. (Wird eine Symbolleiste in einem eigenen kleinen Fenster dargestellt, können Sie sie auch mit dem zugehörigen SCHLIESSEN-Symbol der Leiste ausblenden.)

Umgekehrt können Sie durch Anklicken eines häkchenlosen Eintrags die betreffende Leiste jederzeit einblenden.

Das durch den Befehl ANSICHT|SYMBOLLEISTEN oder das Anklicken einer Symbolleiste mit der *rechten* Maustaste geöffnete Menü enthält nicht alle Symbolleisten.

Uneingeschränkten Zugriff auf alle Symbolleisten *und zusätzlich auch auf die verschiedenen Menüleisten* erhalten Sie erst, nachdem Sie darin den Befehl ANPASSEN... wählen, um das gleichnamige Dialogfeld zu öffnen (oder den Befehl EXTRAS|ANPASSEN...).

Im Register *Symbolleisten* des Dialogfelds sind alle verfügbaren Leisten aufgeführt. Durch Aktivierung/Deaktivierung des zugehörigen Kontrollkästchens können Sie beliebige Leisten ein- oder ausblenden (Abbildung 1.15).



Abbildung 1.15:
Register »Symbol-
leisten«

Blenden Sie viele Leisten ein, sieht der Bildschirm etwas überfüllt aus. Einige der Leisten werden in einem eigenen Fenster mit Überschrift und Rahmen dargestellt (unverankerte Leisten), andere wiederum als fensterlose Leisten am unteren oder oberen Rand des Excel-Fensters (verankerte Leisten) (Abbildung 1.16).

Um aus einer verankerten eine unverankerte Leiste zu machen, doppelklicken Sie einfach auf irgendeine Stelle der Leiste, an der sich kein Symbol befindet, beispielsweise auf eine der dünnen Trennlinien zwischen Symbolen bzw. zwischen zwei benachbarten Symbolleisten.

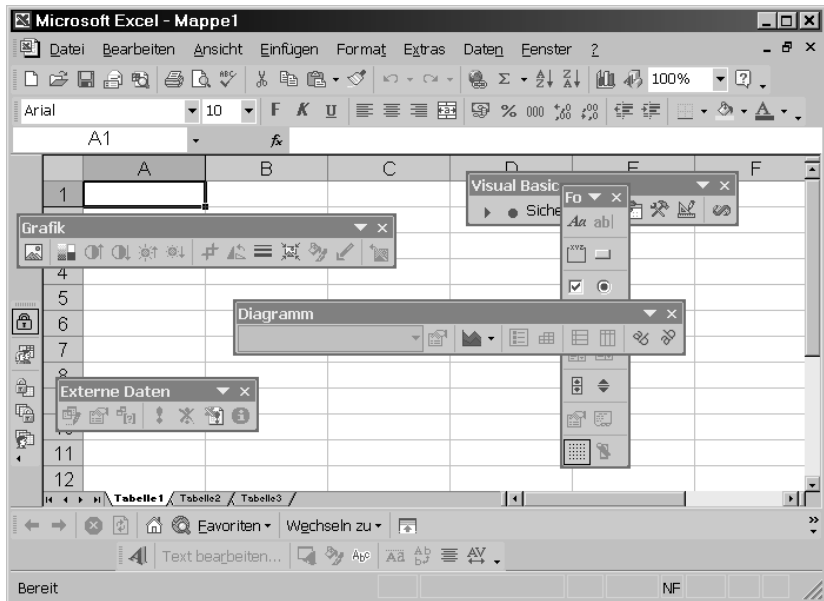
Alle Leisten können Sie frei verschieben: bei unverankerten Leisten klicken Sie wie gewohnt den Fenstertitel an, zum Beispiel *Grafik*, und ziehen daran. Bei verankerten Leisten ziehen Sie stattdessen am abgebildeten Griff (schwebt er darüber, so verändert der Cursor seine Form), der sich am linken Rand der Symbolleiste befindet.



Ziehen Sie eine Leiste zu einem der Fensterränder, so wird sie dort als fensterlose verankerte Leiste eingefügt, analog zur Standardleiste. Verschieben Sie sie in das Excel-Fenster, bekommt sie ihr eigenes Fenster, wird also zur unverankerten Leiste.

Abbildung 1.16:

Verankerte und
unverankerte
Symbolleisten



Die Größe unverankerter Leisten können Sie durch Anfassen an den Fens-
terrändern und Ziehen mit der Maus manipulieren: Ziehen Sie am unteren
Leistenrand nach unten, wird die Leiste höher und die Symbole über eine
zusätzliche Zeile verteilt. Schieben Sie dagegen den unteren Leistenrand
nach oben, wird sie niedriger und dafür breiter.

Weitere Symbole

Bei verankerten Leisten können Sie die Breite verändern, indem Sie den
Griff packen und nach rechts oder links ziehen. Ist eine Symbolleiste
anschließend zu schmal, um alle darin enthaltenen Symbole anzuzeigen
(oder das Excel-Fenster einfach zu klein dafür), fehlen in der Leiste zwangs-
läufig einige Symbole.

Als Hinweis darauf erscheint am rechten Rand der Leiste ein Doppelpfeil
>>. Klicken Sie darauf, so werden die fehlenden Symbole angezeigt und Sie
können nun auf das Gewünschte klicken (Abbildung 1.17).

Bewegen Sie den Cursor auf *Schaltflächen hinzufügen oder entfernen*,
erscheint zunächst eine Liste der Symbolleisten, die sich in der betreffenden
Zeile befinden, beispielsweise *Standard* und *Format*. Klicken Sie auf eine
davon, wird eine Liste aller Symbole der betreffenden Leiste eingeblendet
(Abbildung 1.18).

Klicken Sie auf eines der Symbole, so wird es nicht mehr in der Leiste ange-
zeigt und das Häkchen davor verschwindet. Das Symbol bleibt jedoch in
dieser Liste erhalten, sodass Sie es durch erneutes Anklicken wieder einblen-
den können.



Abbildung 1.17:
Weitere Symbole anzeigen

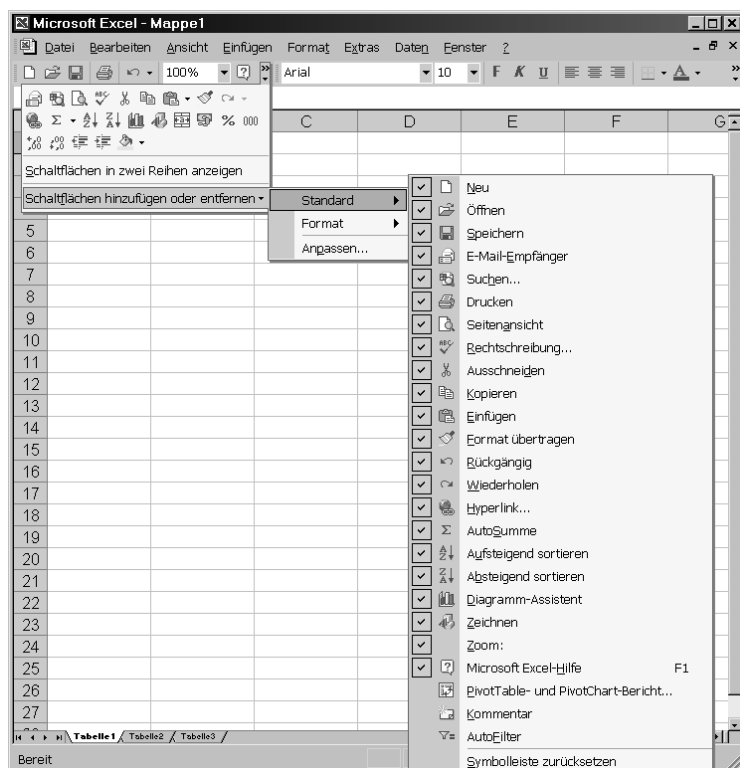


Abbildung 1.18:
Symbole ein-/ausblenden

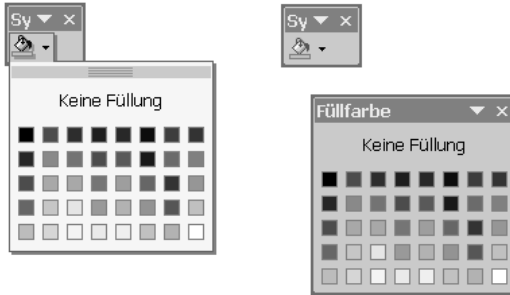
Symboleiste zurücksetzen stellt den ursprünglichen Zustand der Symbolleiste wieder her.

Der Befehl *ANPASSEN...* am Ende der Symbolleiste öffnet das bereits zuvor gezeigte Dialogfeld zum Anpassen der Symbolleisten.

Zu verschiedenen Symbolen gehört ein DropDown-Pfeil. Klicken Sie ihn an, so öffnet sich ein *Paletten*-Listenfeld, aus dem Sie beispielsweise eine Farbe oder ein Muster aussuchen können (Abbildung 1.19).

*Wegziehbare
Paletten*

Abbildung 1.19:
Palette wegziehen

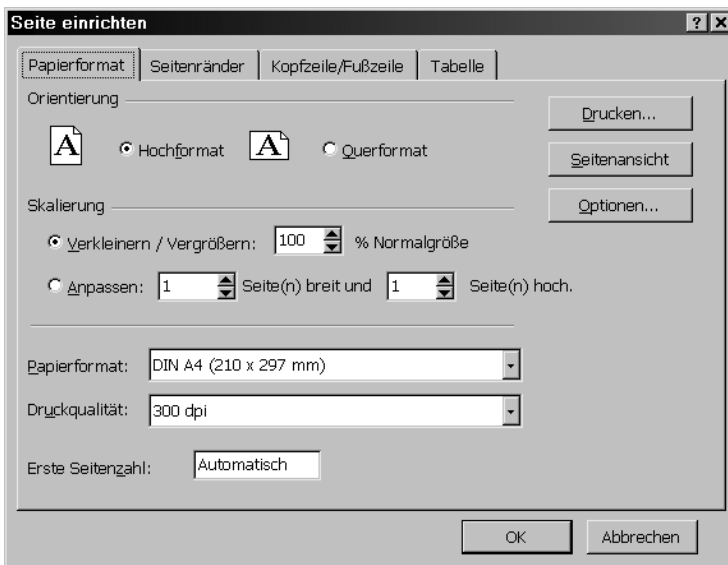


Normalerweise schließt sich das Listenfeld wieder, nachdem Sie darin irgendeinen Eintrag selektiert haben. Benötigen Sie eines dieser Listenfelder sehr oft und soll es daher ständig geöffnet bleiben, so klicken Sie den oberen Rand der Palette an und ziehen sie bei gedrückter Maustaste vom zugehörigen Symbol weg (rechter Teil der Abbildung). Die Palette wird dadurch zum unverankerten Fenster, das geöffnet bleibt, bis Sie es mit einem Klick auf das Kästchen in der oberen rechten Ecke schließen.

1.2.3 Dialogfelder

Menübefehle, die wie DATEI|SEITE EINRICHTEN... mit drei Pünktchen enden, öffnen ein Dialogfeld (Abbildung 1.20).

Abbildung 1.20:
Dialogfeld



**Dialogfeld-
elemente**

Jedes Dialogfeld besitzt einen ganz oben angezeigten Titel, den Dialogfeldnamen. Im abgebildeten Dialogfeld *Seite einrichten* legen Sie fest, wie eine Tabelle ausgedruckt wird: ob im Hoch- oder im Querformat, welche Druck-

qualität verwendet werden soll und so weiter. Die harmlosesten Elemente dieses Dialogfelds sind:

- ➔ Eingabefelder wie *Erste Seitenzahl*, in die Texte oder Zahlen eingegeben werden;
- ➔ Drehfelder wie *Anpassen*, bei denen zwei Bildlaufpfeile mit einem Eingabefeld verbunden sind. Klicken auf den aufwärts gerichteten Pfeil erhöht den im zugehörigen Eingabefeld angezeigten Wert und Klicken auf den abwärts gerichteten Pfeil vermindert ihn. Per Dauerfeuer auf einen der beiden Pfeile können Sie die verfügbaren Werte zügig durchblättern;
- ➔ runde Optionsfelder, die Gruppen bilden, in denen jeweils eine Option ausgewählt ist, z.B. die Gruppe *Orientierung*, die aus den Optionen *Hochformat* und *Querformat* besteht. Klicken Sie auf eine Option wie *Querformat*, wird diese aktiviert und gleichzeitig die zuvor aktive Option der Gruppe deaktiviert.

Teilweise sind mehrere dieser Elemente miteinander kombiniert. Zum Drehfeld rechts neben der Option *Anpassen* gehört beispielsweise ein Eingabefeld, in dem Sie den gewünschten Wert auch direkt eintippen können.

In jedem Fall aktiviert Anklicken mit der Maus das betreffende Element. Klicken Sie beispielsweise auf das zu *Erste Seitenzahl* gehörende Eingabefeld, wird der Cursor auf das angeklickte Zeichen dieses Feldes gesetzt und Sie können anschließend den Feldinhalt editieren.

Viele Dialogfelder enthalten offene oder geschlossene Listenfelder, in der vorhergehenden Abbildung beispielsweise die Felder *Papierformat* und *Druckqualität*. Geschlossene Listenfelder sind mit offenen praktisch identisch, werden jedoch (wie in der folgenden Abbildung *Papierformat*) erst durch Anklicken des zugehörigen Pfeils geöffnet (Abbildung 1.21).

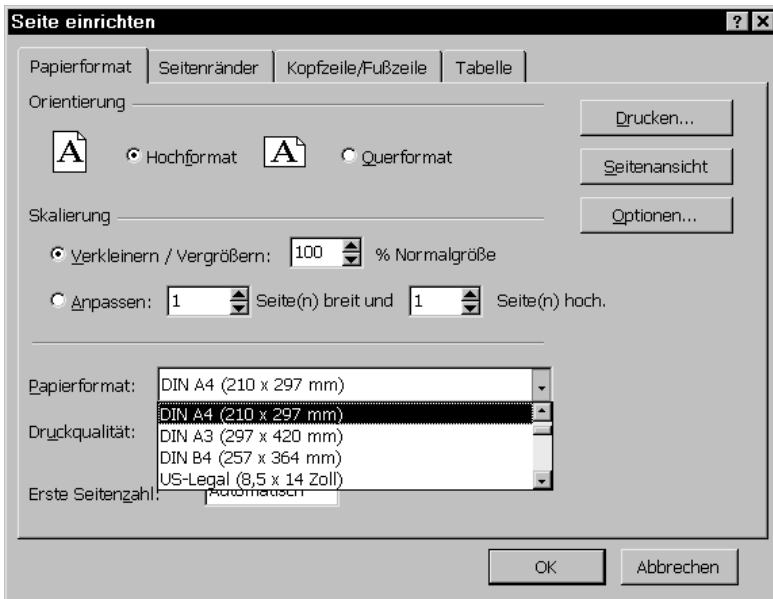
Listenfelder

Ist das Listenfeld zu klein, um alle verfügbaren Einträge auf einmal anzuzeigen, dann befindet sich am rechten Rand der Liste eine vertikale Bildlaufleiste und Sie können die Liste durchblättern, indem Sie auf die zugehörigen Bildlaufpfeile klicken.

Die Suche nach einem bestimmten Eintrag können Sie über dessen Anfangsbuchstaben beschleunigen. In der Abbildung können Sie beispielsweise einfach die Taste **A** drücken, um den ersten Eintrag zu selektieren, der mit diesem Buchstaben beginnt (»A4 (210 x 297 mm)«).

Egal, ob ein Listenfeld offen ist oder erst per Mausklick geöffnet werden muss: In beiden Fällen selektieren Sie den gewünschten Eintrag per Anklicken.

Abbildung 1.21:
Geöffnetes
Listenfeld



:-) TIPP

In einigen offenen Listenfeldern wie zum Beispiel jenem, das der Befehl **DATEI|ÖFFNEN...** öffnet, können Sie mehrere Einträge gleichzeitig selektieren. Dazu halten Sie einfach beim Anklicken jedes weiteren Eintrags die **[Strg]**-Taste gedrückt: Sie selektieren den ersten interessierenden Eintrag per Anklicken, klicken bei gedrückter **[Strg]**-Taste den zweiten Eintrag an, der Sie interessiert, bei ebenfalls gedrückter **[Strg]**-Taste den dritten Eintrag und so weiter.

Unmittelbar aufeinander folgende Einträge selektieren Sie, indem Sie auf den obersten Eintrag klicken, der Sie interessiert, und danach bei gedrückter **[⇧]**-Taste auf den untersten Eintrag.

Registerkarten

Die meisten Dialogfelder besitzen am oberen Rand »Registerkartenzungen«. Klicken Sie darauf, so wird eine andere »Registerkarte« (kurz: Register) aufgeschlagen, praktisch ein anderes Dialogfeld.

Klicken Sie im zuvor abgebildeten Dialogfeld *Seite einrichten* auf *Seitenränder*, so wird entsprechend das Register zum Einstellen der Seitenränder aktiviert (Abbildung 1.22).

Klicken Sie darin auf *Papierformat*, wird wieder das vorhergehende Register aktiviert.

Kontrollkästchen

Am unteren Rand des Registers *Seitenränder* befinden sich zwei eckige Kontrollkästchen. Ist darin ein Häkchen zu sehen, dann ist es aktiviert, sonst ist es deaktiviert. Durch Anklicken können Sie den Zustand umschalten.

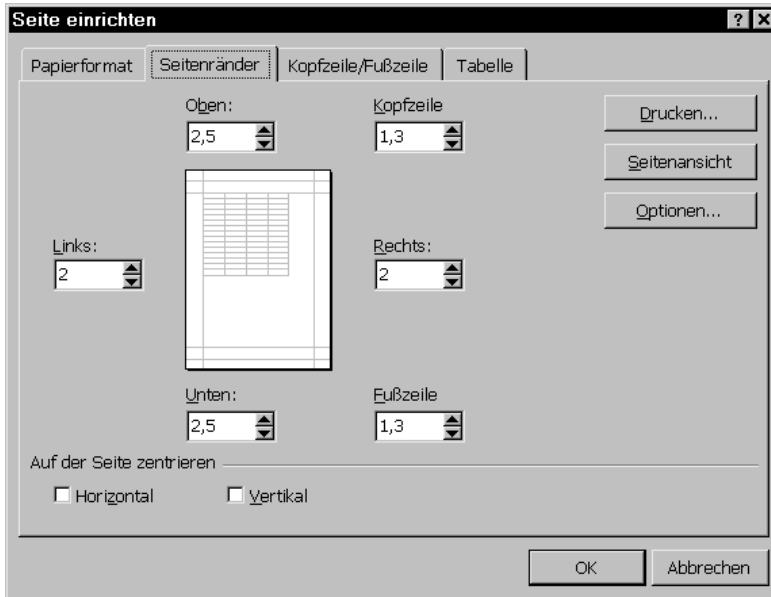


Abbildung 1.22:
Register aktivieren

Im Gegensatz zu runden Optionsfeldern, die Gruppen bilden, in denen jeweils genau eine Option aktiviert ist, sind Kontrollkästchen *unabhängig voneinander* einstellbar. Sie können daher sowohl das Kontrollkästchen *Horizontal* als auch das Kästchen *Vertikal* durch Anklicken aktivieren.

Jedes Dialogfeld verfügt zumindest über die beiden Schaltflächen *OK* und *Abbrechen*, die Sie durch Anklicken aktivieren. *OK* übernimmt die aktuellen Einstellungen und schließt das Dialogfeld.

Schaltflächen

Abbrechen schließt das Dialogfeld ebenfalls, allerdings werden dabei alle von Ihnen vorgenommenen Änderungen wieder rückgängig gemacht. Das abgebildete Symbol (in der rechten oberen Fensterecke) besitzt die gleiche Funktion.



Häufig enthalten Dialogfelder Felder wie *Druckbereich* (im Register *Tabelle*), in die ein Zellbezug einzugeben ist, was am einfachsten durch Überstreichen des betreffenden Zellbereichs mit der Maus (bei gedrückter linker Maustaste) erfolgt. Excel fügt daraufhin die Koordinaten des betreffenden Zellbereichs automatisch ein (Abbildung 1.23).

Ist das Dialogfeld bei dieser Zeigeaktion im Weg, können Sie es mit dem Symbol am rechten Feldrand verkleinern (Abbildung 1.24).

Zeigen Sie nun auf die Zelle oder den Bereich und klicken Sie danach erneut das Symbol am rechten Feldrand an, um die ursprüngliche Dialogfeldgröße wiederherzustellen.

Abbildung 1.23:
Zellbezugseingabe

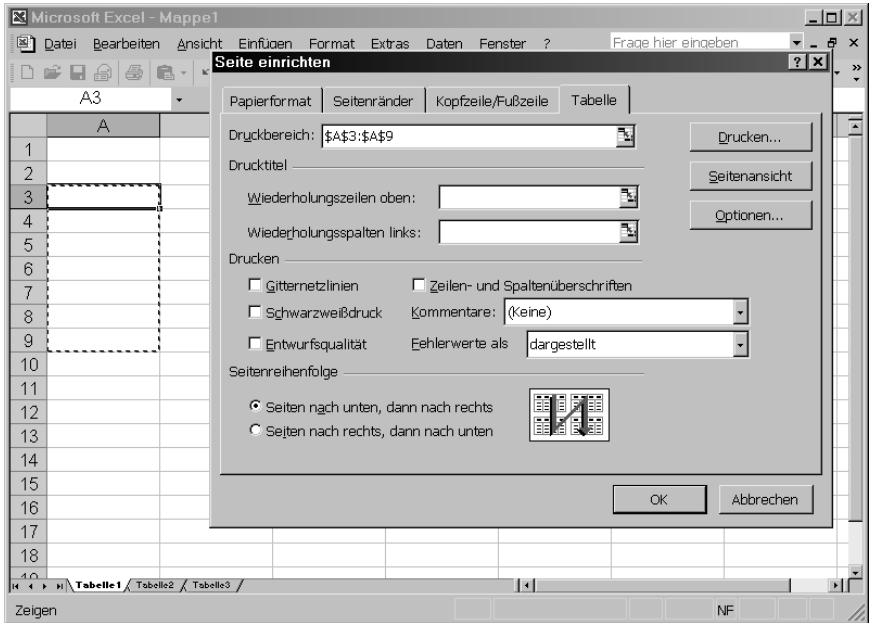


Abbildung 1.24:
Verkleinertes
Dialogfeld



Manchmal enthalten Dialogfelder zusätzliche Schaltflächen (im Beispiel *Drucken...* und *Optionen...*), deren Aufschrift mit drei Punkten endet. Klicken Sie darauf, so öffnet sich ein thematisch verwandtes Dialogfeld.

:-) TIPP

Einige Hinweise zur Bedienung von Dialogfeldern per Tastatur:

- ➔ Mit bewegen Sie sich von Element zu Element vorwärts und mit zurück.
- ➔ Mit der Taste schalten Sie den Zustand einer Option oder eines Kontrollkästchens um, aktivieren oder – bei erneutem Drücken – deaktivieren es also.
- ➔ Innerhalb einer runden Optionsgruppe wählen Sie mit und die gewünschte Option aus (oder mit bzw. .
- ➔ Sie können ein bestimmtes Element eines Dialogfeldes direkt anwählen, indem Sie die **[Alt]**-Taste zusammen mit dem im betreffenden Elementnamen unterstrichenen Buchstaben drücken. Handelt es sich um ein Kontrollkästchen, wird sein Zustand dabei umgeschaltet. Ist es eine Option, wird sie ausgewählt.

➔ Den Inhalt von Eingabefeldern editieren Sie mit folgenden Tasten:

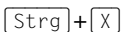
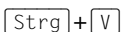
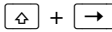
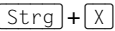
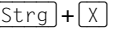
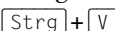
Taste	Funktion
	Ein Zeichen nach rechts
	Ein Zeichen nach links
 + 	Ein Wort nach rechts
 + 	Ein Wort nach links
	Sprung zum Anfang der Bearbeitungsleiste
	Sprung zum letzten eingegebenen Zeichen
	Zeichen links entfernen
 +  oder 	Aktuelles Zeichen löschen
 + 	Zuletzt markierte und dann gelöschte Zeichen einfügen
 + 	Markieren

Tabelle 1.2:

Tastenfunktionen
zum Editieren von
Eingabefeldern

- ➔ Halten Sie beim Umherwandern mit diesen Tasten die -Taste gedrückt, so werden die betreffenden Zeichen markiert.
- ➔ Drücken Sie anschließend  +  oder einfach , so werden statt eines einzigen Zeichens alle markierten Zeichen gelöscht.
- ➔ Mit  +  können Sie die zuletzt mit der -Taste markierten und dann mit  +  gelöschten Zeichen an einer beliebigen Stelle wieder in den Text einfügen.
- ➔ Mit  +  kopieren Sie Zeichen: Markieren Sie die zu kopierende Zeichenkette und drücken Sie  + . Kümmern Sie sich nicht darum, dass scheinbar nichts passiert, sondern bewegen Sie den Cursor zur gewünschten Position und fügen Sie die Zeichen dort mit  +  ein.
- ➔ Nach der Selektion eines Eingabefeldes mit  ist es komplett markiert. Drücken Sie jetzt eine Buchstabentaste, dann geht Excel davon aus, dass Sie die Vorgabe *komplett* verwerfen und einen völlig anderen Text eingeben wollen, und löscht daher die Vorgabe. Drücken Sie stattdessen eine Cursortaste (, , , ), nimmt Excel an, dass Sie die Vorgabe zumindest teilweise übernehmen wollen, und löscht sie nicht.
- ➔ Statt die Schaltfläche *Abbrechen* mit  zu selektieren und danach mit  zu aktivieren, können Sie übrigens einfach  drücken. Analog dazu aktiviert  die Schaltfläche *OK* – vorausgesetzt, Sie

fokussierten nicht zuvor mit  eine der anderen Schaltflächen, da  sonst diese aktiviert! Allgemein:  aktiviert immer *die momentan fokussierte Schaltfläche*, jene, die Excel durch einen dickeren Rahmen gegenüber den restlichen Schaltflächen hervorhebt.

- ➔ Mit  können Sie auch Listenfelder aktivieren. Mit den Tasten  und  selektieren Sie anschließend einen der darin enthaltenen Einträge. Erreichen Sie dabei den oberen beziehungsweise unteren Rand des Listenausschnitts, so verschiebt Excel den gezeigten Ausschnitt in die entsprechende Richtung.
- ➔ Felder wie *Druckbereich*, die Zellbezüge enthalten: Mit den Richtungstasten legen Sie zunächst die Ausgangszelle fest. Danach legen Sie mit den gleichen Tasten bei gleichzeitig gedrückter -Taste den interessierenden Zellbereich fest.

1.2.4 Aufgabenbereich

Mehrere Excel-Befehle öffnen statt eines herkömmlichen Dialogfelds den einen beträchtlichen Teil des Excel-Fensters einnehmenden Aufgabenbereich, beispielsweise der Befehl DATEI|NEU... (Abbildung 1.25).

Abbildung 1.25:
Aufgabenbereich



Der Aufgabenbereich ist eine Mischung aus verschiedenen Komponenten:

- ➔ Menüs: Die aufgelisteten Kommandos werden wie »Hyperlinks« dargestellt: Schwebt der Cursor über einem Kommando wie *Leere Arbeitsmappe*, wird es unterstrichen. Klicken Sie darauf, wird das Kommando ausgeführt, in diesem Fall also wird eine neue Arbeitsmappe erstellt.

- ➔ **Dialogfelder:** Wie ein Dialogfeld enthält der Aufgabenbereich verschiedene Register. Um ein anderes aufzuklappen, klicken Sie auf den abwärts weisenden Pfeil am oberen Fensterrand. Die verfügbaren Register werden aufgelistet und das momentan angezeigte per Häkchen hervorgehoben. Um ein anderes Register wie *Zwischenablage* aufzuschlagen, klicken Sie auf den betreffenden Eintrag (Abbildung 1.26).



Abbildung 1.26:
Register wechseln

- ➔ **Webbrowser:** Mit den beiden nach rechts bzw. links gerichteten Pfeilen schlagen Sie das zuvor geöffnete Register (Pfeil nach links) bzw. anschließend das danach geöffnete Register wieder auf (Pfeil nach rechts).

Da er sehr viel Platz einnimmt, empfehle ich Ihnen, den Aufgabenbereich zu schließen, wenn er nicht benötigt wird, indem Sie auf das Kreuz in der rechten oberen Ecke klicken. Wie gesagt, bei Auswahl eines Befehls, zu dessen Durchführung er benötigt wird, erscheint dieser Bereich automatisch.

Davon unabhängig können Sie ihn jederzeit mit **ANSICHT|AUFGABENBEREICH** einblenden.

*Der Aufgabenbereich erscheint beim Start von Excel automatisch. Wollen Sie das abstellen, klappen Sie im Aufgabenbereich das Register **Neue Arbeitsmappe** auf und deaktivieren das Kontrollkästchen **Beim Start anzeigen** (oder deaktivieren das Kontrollkästchen **Startaufgabenbereich im Register Ansicht des Befehls EXTRAS|OPTIONEN...**).*





Doppelklicken Sie auf die Titelleiste des Aufgabenbereichs, wird er zu einem eigenständigen, frei verschiebbaren und vergrößer-/verkleinerbaren Fenster und per erneutem Doppelklick auf die Titelleiste wieder zu einem am rechten Rand des Access-Fensters integrierten Abschnitt.

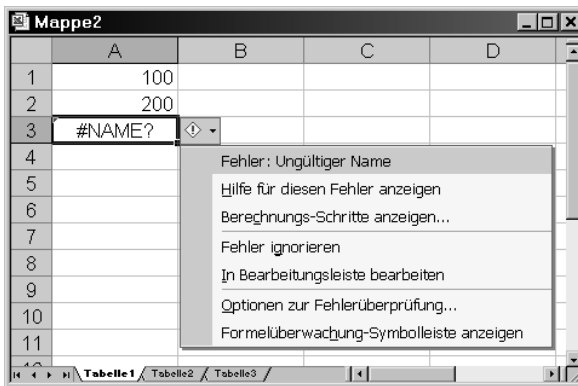
1.2.5 Smart Tags

Ein Smart Tag ist eine Art »Achtung-Schild« an bzw. neben einer Zelle. Zunächst ist es recht dezent in Form eines kleinen Dreiecks in der linken oberen Ecke der Zelle, dem »Smart Tag-Indikator«.

Tun Sie ihm den Gefallen, auf die Zelle zu klicken, erscheint neben ihr das abgebildete »Fehler-Smart Tag«, ein deutlich hervorgehobenes Ausrufezeichensymbol.

Klicken Sie darauf, öffnet sich ein Kontextmenü, das zur Fehlerbehebung nützliche Befehle enthält (Abbildung 1.27).

Abbildung 1.27:
Fehler-Smart Tag



Außer diesem Fehler-Smart Tag gibt es noch weitere Smart Tags, die in anderen Situationen erscheinen, beispielsweise das beim Kopieren und Einfügen erscheinende Einfügen-Smart Tag: Nach dem Einfügen des Inhalts der Zwischenablage mit BEARBEITEN|EINFÜGEN bzw. dem zugehörigen Symbol (aber nicht nach dem Einfügen per ) erscheint ebenfalls ein Smart Tag.

Klicken Sie darauf, erscheint ein Kontextmenü mit den meistgebrauchten Einfügeoptionen des Befehls BEARBEITEN|INHALTE EINFÜGEN..., das es Ihnen erspart, extra diesen Befehl wählen zu müssen, wenn Sie eine der (teilweise sehr häufig verwendeten) speziellen Einfügeoptionen benötigen (Abbildung 1.28).

Wie Kontextmenüs bieten auch *Smart Tags* nichts grundlegend Neues, sondern stellen Abkürzungen dar, um schneller ans Ziel zu gelangen. Der Unterschied zu Kontextmenüs ist die Automatik, dass Excel auf die Verfügbarkeit des betreffenden Kontextmenüs selbstständig hinweist.

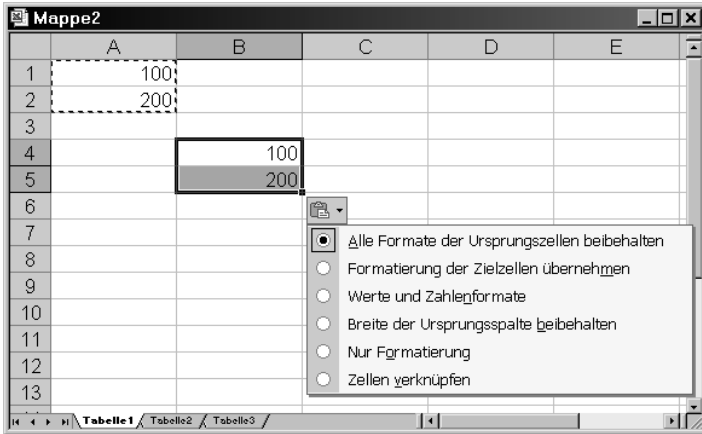


Abbildung 1.28:
Einfügen-Smart Tag

1.3 Fenster

1.3.1 Bildlauf

Excel-Tabellen können erheblich größer sein als der kleine Ausschnitt, der in einem Arbeitsfenster sichtbar ist. Diesen Ausschnitt können Sie mit den vier Bildlaufpfeilen am Rand des Arbeitsfensters beliebig verschieben. Nehmen wir folgendes Beispiel (Abbildung 1.29).

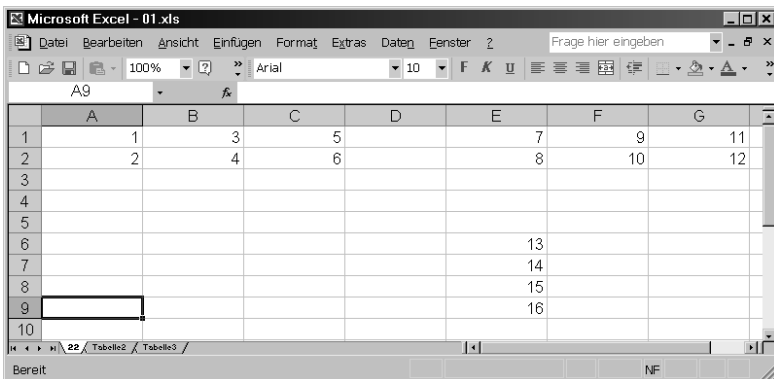


Abbildung 1.29:
Bildlauf

Die Zahlen geben Sie ein, indem Sie die betreffende Zelle anklicken oder den Zellcursor mit den vier Pfeiltasten (Cursortasten) dorthin bewegen, die zugehörige Zahl eintippen und  drücken.

Klicken Sie kurz auf den nach unten weisenden Pfeil am rechten Rand des Arbeitsfensters, bewegen Sie den Ausschnitt um eine Zeile nach unten: statt der Zeilen 1 bis 10 sind darin anschließend die Zeilen 2 bis 11 der Tabelle zu sehen.

Bildlaufpfeile

Mit dem nach oben weisenden Pfeil wird der Ausschnitt genau umgekehrt eine Zeile nach oben bewegt.

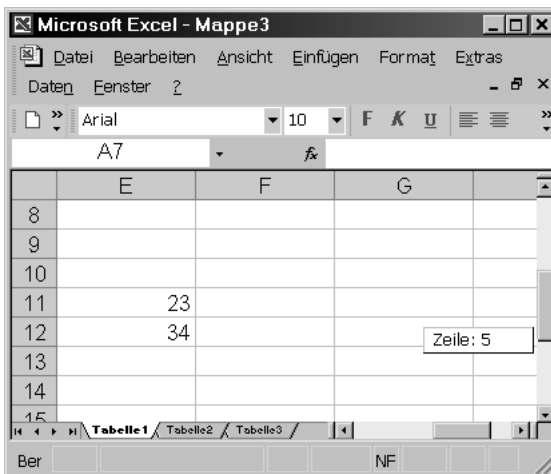
Die nach rechts beziehungsweise links weisenden Bildlaufpfeile bewirken einen entsprechenden seitlichen Bildlauf.

Bildlauffelder

Größere Strecken überwinden Sie am einfachsten mit den Bildlauffeldern, den länglichen Kästchen, die sich zwischen je zwei entgegengesetzt ausgerichteten Bildlaufpfeilen befinden. Je kürzer dieser Kasten ist, desto größer ist der bis ganz unten zurückzulegende Weg, sodass die Länge grob über die Gesamtgröße einer Tabelle informiert.

Klicken Sie die Tabelle mit der Maus an und ziehen Sie sie bei gedrückter linker Maustaste in die gewünschte Richtung. Lassen Sie die Maustaste los, so wird ein Bildlauf in diese Richtung durchgeführt, der um so weiter ausfällt, je mehr Sie das Bildlauffeld verschieben (Abbildung 1.30).

Abbildung 1.30:
Zeilenanzeige beim
Bildlauf



Der Bildlauf wird übrigens erheblich beschleunigt, wenn Sie während des Ziehens mit der Maus die -Taste gedrückt halten.

Ähnliche Sprünge wie durch Ziehen an einem Bildlauffeld bewirken Sie auch durch Klicken auf den Leerraum zwischen dem Bildlauffeld und den zugehörigen Richtungspfeilen.

Mit den folgenden Tastenkombinationen kommen Sie teilweise erheblich schneller zum Ziel als mit der Maus:

- ➔ Bewegen Sie den Zellcursor mit den Cursortasten zu einem der vier Ränder des Arbeitsblatts und dann darüber hinaus, so erfolgt ebenfalls ein Bildlauf in die betreffende Richtung.



- ➔ **[Bild↓]** rollt den Ausschnitt um eine Seite nach unten. Statt den Zeilen 1 bis 10 werden nun die Zeilen 11 bis 20 angezeigt. **[Bild↑]** rollt den Bildschirm entsprechend um eine Seite nach oben.
- ➔ **[Alt] + [Bild↓]** blättert eine Seite nach rechts, **[Alt] + [Bild↑]** eine Seite nach links.
- ➔ Enthält Ihre Tabelle Zahlenblöcke, können Sie blockweise hin- und herspringen. Befinden Sie sich beispielsweise auf Zelle A1 und drücken Sie mehrmals nacheinander **[Strg] + [→]**, kommen Sie von Zelle A1 zu C1, dann zu E1, G1 etc.

Allgemein: **[Strg] + [→]** bewegt den Cursor von Datenblock zu Datenblock nach rechts, zum Anfang eines Blocks, dann zum Ende des Blocks, zum Anfang des nächsten Blocks, zu dessen Ende und so weiter. Wenn rechts nur noch leere Zellen vorhanden sind, gelangen Sie ans äußerste rechte Ende des Arbeitsblatts, zur 255. Spalte, die die Bezeichnung IV trägt. **[Strg] + [←]** bewegt den Cursor entsprechend blockweise nach links, **[Strg] + [↓]** nach unten und **[Strg] + [↑]** nach oben.

- ➔ **[Pos1]** führt immer zur ersten Zelle der aktuellen Zeile, zum Beispiel zu A7, wenn Sie sich gerade in Zeile 7 befinden, **[Strg] + [Pos1]** zur oberen linken Ecke A1 des Arbeitsblatts und **[Strg] + [Ende]** zur rechten unteren Ecke der Tabelle, in unserem Fall zu G9, dem Schnittpunkt zwischen der letzten besetzten Spalte G und der untersten besetzten Zeile 9.

Taste	Funktion
[→]	Spalte bzw. Bildlauf nach rechts um eine Spalte
[←]	Spalte bzw. Bildlauf nach links um eine Spalte
[↓]	Zeile bzw. Bildlauf nach unten um eine Zeile
[↑]	Zeile bzw. Bildlauf nach oben um eine Zeile
[Strg] + [→]	Blockweise nach rechts springen
[Strg] + [←]	Blockweise nach links springen
[Strg] + [↓]	Blockweise nach unten springen
[Strg] + [↑]	Blockweise nach oben springen
[Pos1]	Zur ersten Zelle der aktuellen Zeile springen
[Strg] + [Pos1]	Zu Zelle A1 springen
[Strg] + [Ende]	Zur rechten unteren Tabellenecke springen

Tabelle 1.3:
Steuerung des
Bildlaufs und des
Zellcursors

Tabelle 1.3:
Steuerung des
Bildlaufs und des
Zellcursors
(Forts.)

Taste	Funktion
	Eine Seite nach unten blättern
	Eine Seite nach oben blättern
	Eine Seite nach rechts blättern
	Eine Seite nach links blättern

1.3.2 Fenstermodi

Ein Fenster kann drei Zustände besitzen: Es kann a) als echtes Fenster mit Rahmen und Bildlaufleisten dargestellt werden, b) als Vollbild, in dem es den gesamten Bildschirm einnimmt, oder c) als Symbol.

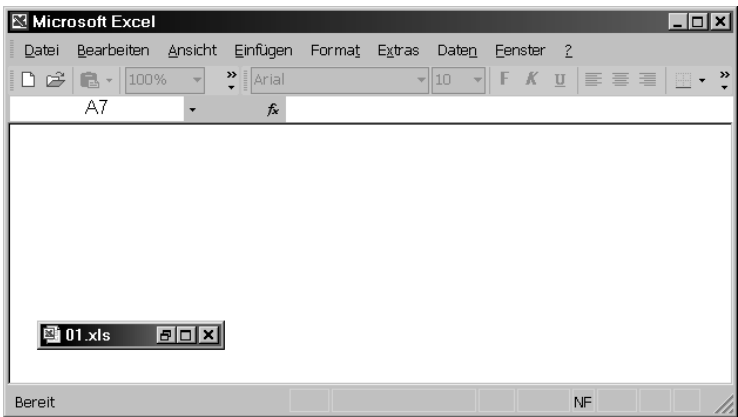
Nach dem Aufruf von Excel befinden sich zwei Fenster auf dem Bildschirm: Das Excel-Fenster wird als Vollbild dargestellt und das darin enthaltene Tabellenfenster als Fenster mit einem eigenen Rahmen und je einer Bildlaufleiste am rechten bzw. unteren Fensterrand.

Durch das Anklicken verschiedener Knöpfe in der rechten oberen Ecke eines Fensters lösen Sie spezifische Fensterfunktionen aus. Diese Knöpfe enthält sowohl das Tabellenfenster als auch das Excel-Fenster selbst.



Dieser Knopf verkleinert das Tabellenfenster zum Symbol (Abbildung 1.31).

Abbildung 1.31:
Tabellenfenster als
Symbol



Klicken Sie auf den gleichen Knopf des umgebenden Excel-Fensters, wird dieses Fenster inklusive des darin enthaltenen Tabellenfensters zum Symbol.

Das Gegenteil zur Symbol- ist die durch den abgebildeten Knopf aktivierte Vollbilddarstellung. Das Excel-Fenster selbst befindet sich bereits in diesem Modus und nimmt den gesamten Bildschirm ein. Wollen Sie möglichst viel von einer Tabelle sehen, benutzen Sie diesen Modus auch für das Tabellenfenster.



In diesem Modus werden die Symbolknöpfe des Tabellenfensters unmittelbar unter die entsprechenden Symbole des Excel-Fensters selbst verschoben, sodass Sie nun beim Knopfdrücken aufpassen müssen, ob Sie einen Excel-Fensterknopf drücken oder einen Knopf des Tabellenfensters!

Egal, ob das Tabellenfenster oder das gesamte Excel-Fenster momentan als Symbol oder als Vollbild dargestellt wird: Klicken Sie auf dieses Symbol, so wird es in jedem Fall als Fenster dargestellt: Das Tabellenfenster wird wieder als eigenständiges Fenster innerhalb des Excel-Fensters dargestellt und wenn Sie auf das gleiche Symbol des Excel-Fensters klicken, wird dieses ebenfalls ein wenig verkleinert und zu einem echten Fenster mit eigenem Rahmen (Abbildung 1.32).

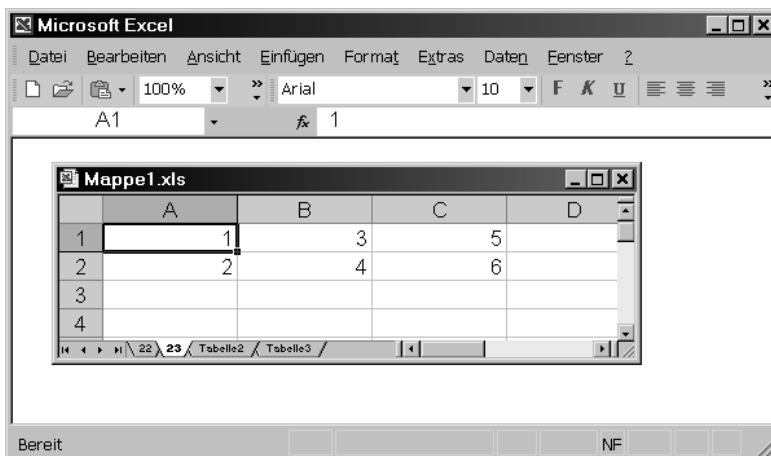


Abbildung 1.32:
Tabellenfenster als
echtes Fenster
innerhalb des Excel-
Fensters

Am einfachsten schalten Sie per Doppelklick auf die Titelleiste eines Fensters zwischen der Darstellung als echtes Fenster bzw. als Vollbild um.

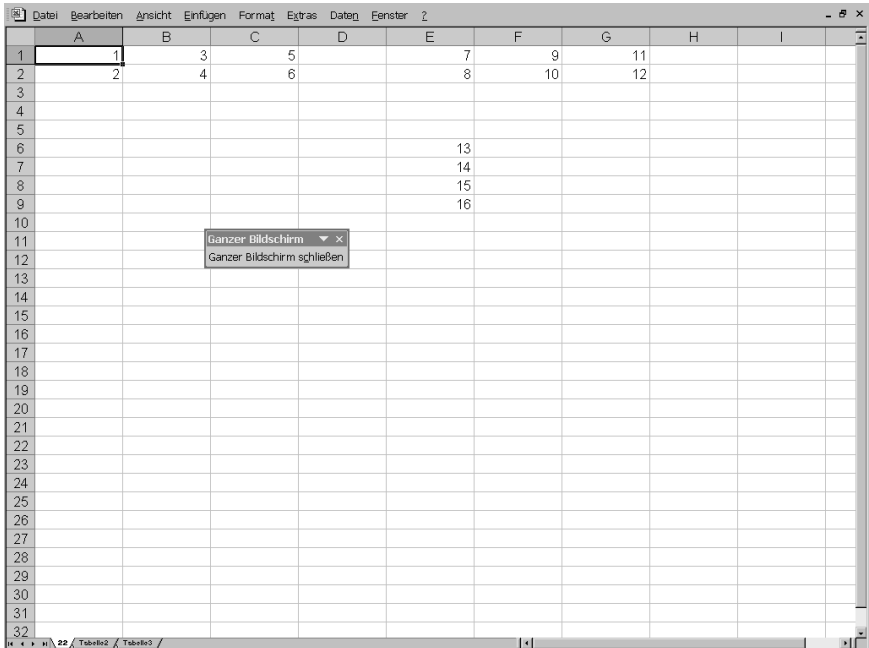
Dieses Symbol schließt das betreffende Fenster. Angewandt auf das Tabellenfenster ist Excel danach zwar weiter vorhanden, die Tabelle jedoch verschwunden. Angewandt auf das Excel-Fenster, wird Excel geschlossen und damit beendet, äquivalent zum Befehl BEENDEN im DATEI-Menü.

Eine Spezialität von Excel ist der Befehl ANSICHT|GANZER BILDSCHIRM, der das Excel- und das Tabellenfenster soweit wie irgend möglich vergrößert (Abbildung 1.33).



Ganzer Bildschirm

Abbildung 1.33:
Maximale Tabellen-
fenstergröße



Wählen Sie den Befehl erneut oder klicken Sie das automatisch eingeblendete Symbol an, wird wieder die gewohnte Darstellung aktiviert.

1.3.3 Größe und Position

Ein echtes Fenster können Sie beliebig vergrößern oder verkleinern, indem Sie irgendeinen Punkt des Fensterrahmens packen (darauf klicken) und die Maus – bei weiter gedrückter Maustaste – in die gewünschte Richtung ziehen bzw. schieben.

Ziehen/schieben Sie dabei an einem der vier Eckpunkte, so können Sie die Breite und Höhe des Fensters gleichzeitig manipulieren, wenn Sie nicht an einem Eckpunkt ziehen, dagegen nur die Breite oder die Höhe.

Um ein Fenster zu verschieben, klicken Sie einen beliebigen Punkt seiner Titelleiste an und ziehen/schieben die Maus in die gewünschte Richtung.

Systemmenü

All diese Fenstermanipulationen sind auch mit der Tastatur und dem Systemmenü möglich, dem kleinen Bildchen in der linken oberen Fensterecke des Tabellen- bzw. des Excel-Fensters. Um es zu öffnen, drücken Sie **[Alt]** + **[F4]** (Systemmenü des Excel-Fensters öffnen) oder klicken einfach auf das zugehörige Excel-Symbol (Abbildung 1.34).

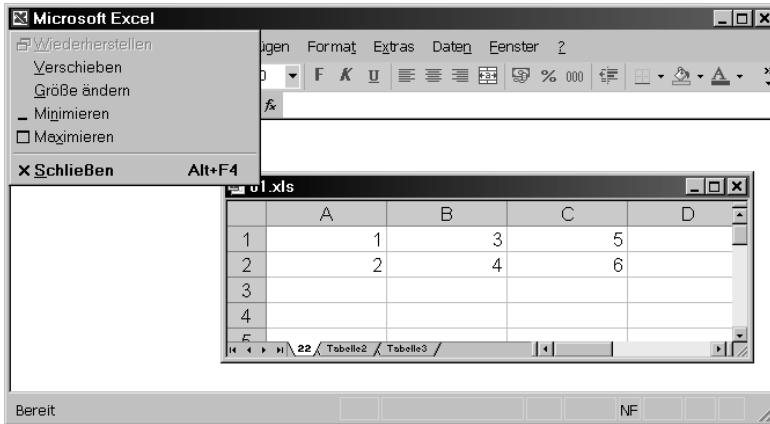


Abbildung 1.34:
Das Systemmenü

Wählen Sie **GRÖSSE ÄNDERN**, dann können Sie das Fenster anschließend mit den Cursortasten in verschiedene Richtungen ziehen und damit vergrößern oder verkleinern und die Größenänderung mit  abschließen.

:-) **TIPP**

Nach Anwahl von **VERSCHIEBEN** können Sie das Fenster mit den Cursortasten über den Bildschirm verschieben. Ist die gewünschte Position erreicht, schließen Sie auch diesen Befehl mit  ab.

MINIMIEREN verkleinert das Arbeitsfenster zum Symbol.

WIEDERHERSTELLEN stellt die alte Fenstergröße wieder her.

MAXIMIEREN aktiviert die Vollbilddarstellung.

SCHLIESSEN schließt das betreffende Fenster.

1.3.4 Mit mehreren Fenstern arbeiten

Dieses Kapitel enthält eine Einführung in den Umgang mit Fenstern. Mehr darüber (Unterfenster, Fensterfixierung etc.) erfahren Sie in Kapitel 8.5, »Fenster->Handling«.

 **REF**

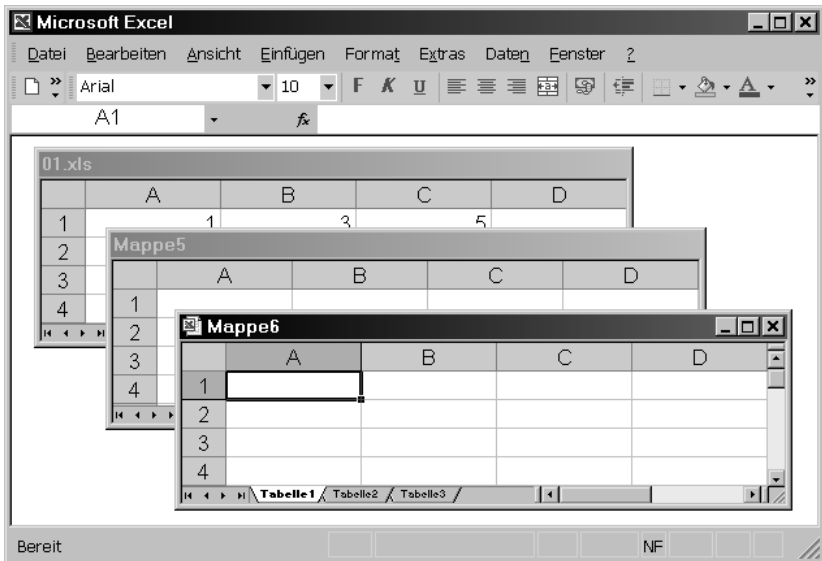
Sie können in Excel beliebig viele Fenster gleichzeitig bearbeiten, die beispielsweise verschiedene Tabellen oder Grafiken enthalten. Die einzige Grenze stellt die verfügbare Speicherkapazität dar.

Klicken Sie auf das abgebildete Symbol, so wird ein neues Fenster angelegt, genauer: eine neue Arbeitsmappe. Stattdessen können Sie auch den Befehl **DATEI|NEU...** wählen und im Aufgabenbereich auf *Leere Arbeitsmappe* klicken.



Erzeugen Sie zwei zusätzliche Fenster, verkleinern Sie beide ein wenig und verschieben Sie sie, so ergibt sich etwa folgender Zustand (Abbildung 1.35).

Abbildung 1.35:
Drei Tabellenfenster



Aktives Fenster

Excel-Befehle wie DATEI|SCHLIESSEN beziehen sich immer auf das oberste Fenster, das aktive Fenster. DATEI|SCHLIESSEN schließt es, alle anderen geöffneten Fenster bleiben jedoch erhalten.

Wenn Sie den Zellcursor bewegen oder Eingaben vornehmen, werden diese Aktionen ebenfalls immer im aktiven Fenster ausgeführt. Sie können aber jederzeit eines der im Stapel weiter unten liegenden Fenster aktivieren, indem Sie einen momentan nicht verdeckten Punkt dieses Fensters anklicken (Tastatur: wiederholtes Drücken von **Strg** + **F6** schaltet der Reihe nach zwischen den verschiedenen Fenstern um).



INFO

Jedes Excel-Unterfenster wird in der Windows-Taskleiste aufgeführt. Auch hier genügt wieder Anklicken, um es zu öffnen und in den Vordergrund zu bringen.

Alternativ dazu können Sie auch das FENSTER-Menü öffnen: Ganz unten finden Sie darin die Namen der momentan geöffneten Fenster. Klicken Sie einen dieser Einträge an (bzw. selektieren Sie ihn mit den Cursortasten und drücken Sie **↵**), wird das Fenster aktiv.

Der Befehl FENSTER|ANORDNEN... erleichtert die individuelle Fensteranordnung. Die Option *Überlappend* des zugehörigen Dialogfelds erzeugt einen Fensterstapel. *Unterteilt* führt dagegen zu folgendem Resultat (Abbildung 1.36).

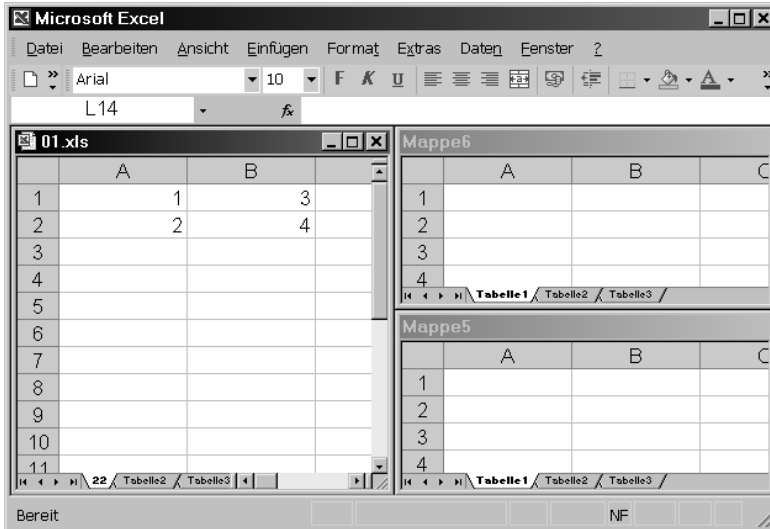


Abbildung 1.36:
Automatische Fensteranordnung durch Excel

Horizontal ordnet die Fenster untereinander an (Abbildung 1.37).

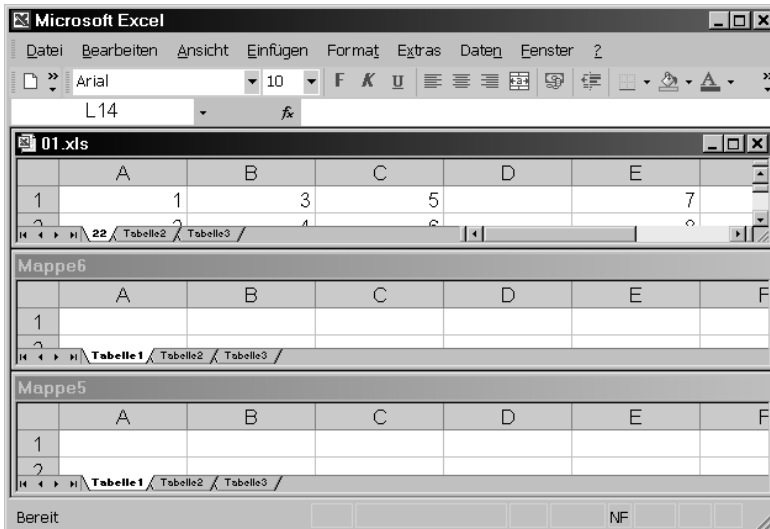
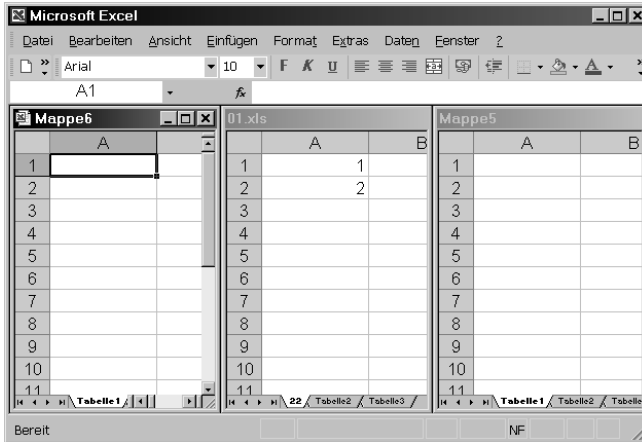


Abbildung 1.37:
Horizontale Fensteranordnung

Und *Vertikal* nebeneinander (Abbildung 1.38).

Aktivieren Sie das Kontrollkästchen *Fenster der aktiven Arbeitsmappe*, so werden nur die Fenster der aktuellen Arbeitsmappe wie beschrieben angeordnet. Dazu müssen Sie wissen, dass Sie mit dem Befehl FENSTER|NEUES FENSTER für die aktuelle Arbeitsmappe ein neues Fenster anlegen und anschließend in beiden Fenstern verschiedene Ausschnitte der gleichen Tabelle betrachten können.

Abbildung 1.38:
Vertikale Fensteran-
ordnung



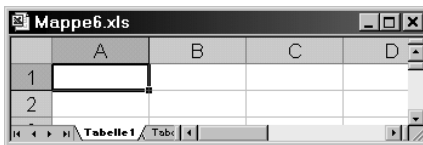
1.4 Arbeitsmappen



Mit DATEI|NEU... und nachfolgender Wahl von *Leere Arbeitsmappe* im Aufgabenbereich bzw. mit dem zugehörigen Symbol legen Sie wie erläutert eine neue Arbeitsmappe an. Eine solche Mappe enthält nicht nur eine, sondern mehrere Excel-Tabellen gleichzeitig. Die genaue Anzahl hängt von der Einstellung *Blätter in neuer Arbeitsmappe* ab, die Sie im Register *Allgemein* des Befehls EXTRAS|OPTIONEN... finden.

Für die Umschaltung zwischen den einzelnen Tabellen in einer Arbeitsmappe ist die Registerleiste am unteren Rand des Tabellenfensters zuständig (Abbildung 1.39).

Abbildung 1.39:
Arbeitsmappen-
Registerleiste



Der hervorgehobene Tabellenname »Tabelle1« weist auf die momentan aktive Tabelle der Arbeitsmappe hin. Klicken Sie einen anderen Registernamen an, wird die betreffende Tabelle aktiviert. Alternativ dazu aktiviert **Strg** + **Bild ↓** die nächste Tabelle einer Arbeitsmappe und **Strg** + **Bild ↑** die vorhergehende.

Die *Registerleiste* ist oft nicht lang genug, um die Namen aller Tabellen einer Mappe gleichzeitig anzuzeigen. Um sie zu verbreitern, klicken Sie den schmalen Querbalken rechts neben dem letzten Namen an (der Mauscursor wird zu einem Doppelpfeil, wenn Sie an der richtigen Position sind), halten die Maustaste gedrückt und ziehen den Balken nach rechts. Die Registerleiste wird

dadurch breiter und die horizontale Bildlaufleiste dafür entsprechend schmaler (Abbildung 1.40).

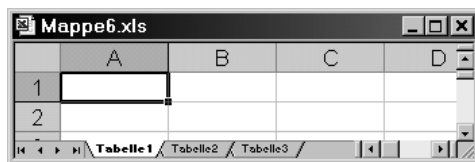


Abbildung 1.40:
Leistenbreite
verändern

Umgekehrt wird die Registerleiste schmaler, wenn Sie den Balken nach links ziehen.

Selbst bei maximaler Verbreiterung können bei vielen in der Mappe enthaltenen Blättern nicht alle Registernamen gleichzeitig angezeigt werden. Um dennoch beliebige Tabellen aktivieren zu können, müssen Sie die Pfeilsymbole am linken Leistenrand verwenden: Die beiden inneren Pfeile rollen die Leiste stufenweise nach rechts bzw. links, die beiden äußeren springen zum Leistenanfang bzw. Leistenende.

In der folgenden Abbildung aktivierte ich die Tabelle Nummer 16, indem ich auf den rechten der vier Pfeile klickte und im nun sichtbaren rechten Leistenende auf den Namen »Tabelle16« (Abbildung 1.41).

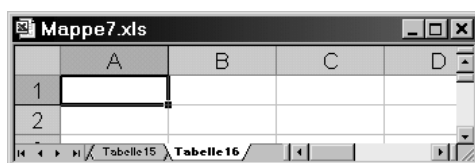


Abbildung 1.41:
»Tabelle16«
aktivieren

Die vier Pfeile besitzen noch eine andere Funktion: Klicken Sie auf einen davon mit der rechten Maustaste, so erscheint ein Kontextmenü, das die Namen aller Tabellen der aktiven Arbeitsmappe enthält, wobei die gerade aktive durch ein Häkchen markiert ist (Abbildung 1.42).

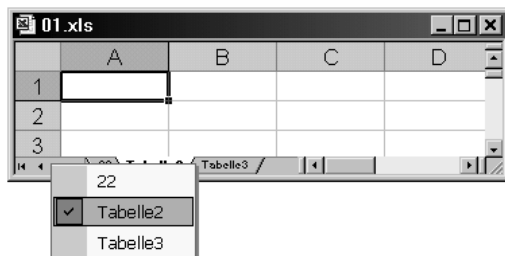


Abbildung 1.42:
Kontextmenü der
Registerleiste

Klicken Sie einen anderen Tabellennamen an, wird die betreffende Tabelle aktiviert.

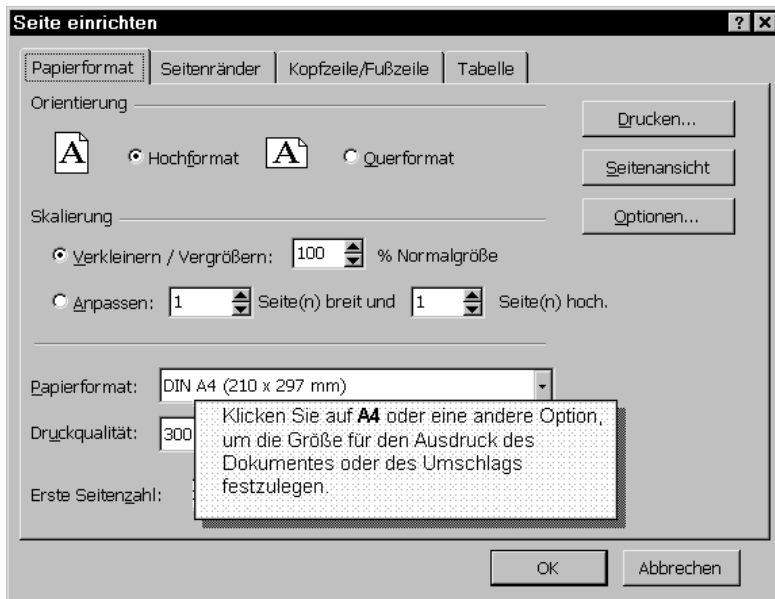
1.5 Hilfestellungen

1.5.1 Kontextsensitive Hilfe



Die kontextsensitive Hilfe öffnet ein Infocfeld mit Informationen zum momentan aktiven Objekt. Das ist vor allem in Dialogfeldern nützlich, um Hilfe zu einem bestimmten Element zu erhalten. Klicken Sie dazu auf das abgebildete Symbol, das sich in der rechten oberen Ecke jedes Dialogfelds befindet. Der Cursor schleppt daraufhin ein großes Fragezeichen mit sich herum, mit dem Sie das Objekt anklicken, zu dem Sie Informationen benötigen (Abbildung 1.43).

Abbildung 1.43:
Infocfeld



Um Hilfe zu anderen Objekten wie Menüs, bestimmten Menübefehlen oder Teilen der Bearbeitungsleiste zu erhalten, klicken Sie auf das abgebildete Symbol (in der Kategorie *Fenster u. Hilfe* (?) des Registers *Befehle* des *Anpassen*-Dialogfelds) oder wählen *?|DIREKTHILFE*.

Der Cursor wird daraufhin ebenfalls zu einem Fragezeichen, mit dem sie auf das betreffende Objekt klicken, beispielsweise auf ein Menü, das sich nun öffnet, und danach auf einen der darin enthaltenen Befehle, woraufhin Kurzinformationen zu diesem Befehl erscheinen.

1.5.2 Der Office-Assistent

Der Office-Assistent ist ein mehr oder weniger intelligenter Helfer, der Ihnen während der Arbeit über die Schulter schaut und Tipps oder Warnungen gibt. Haben Sie beispielsweise in eine Mappe Werte eingetragen und wollen Sie diese schließen, ohne sie gespeichert zu haben, erscheint folgende Warnung (Abbildung 1.44).



Abbildung 1.44:
Warnung

Klicken Sie auf *Ja*, wird die Mappe vor dem Schließen gespeichert; *Nein* bedeutet, dass Sie bewusst auf das Speichern verzichten, und *Abbrechen*, dass Sie den gesamten Vorgang (die Mappe zu schließen) abbrechen wollen.

Zusätzlich gibt der Assistent häufig Tipps zum effizienteren Arbeiten, die Sie gegebenenfalls am Erscheinen einer Glühbirne erkennen, auf die Sie dann klicken sollten (Abbildung 1.45).

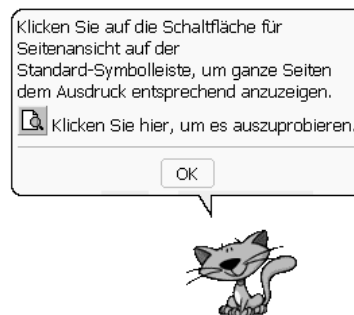
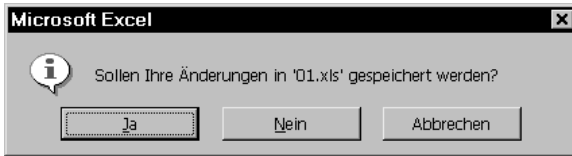


Abbildung 1.45:
Tipp

Wird Ihnen der Assistent lästig, klicken Sie mit der rechten Maustaste auf ihn und wählen Ausblenden oder wählen ?|OFFICE-ASSISTENTEN AUSBLENDEN (und blenden ihn gegebenenfalls mit ?|OFFICE-ASSISTENTEN ANZEIGEN wieder ein). Wichtige Hinweise und Warnungen wie »Wollen Sie das Formular speichern?« mit OK- und Abbrechen-Schaltflächen erscheinen weiterhin, nun jedoch in der konservativen optischen Form, die Sie von anderen Windows-Programmen gewohnt sind (Abbildung 1.46).

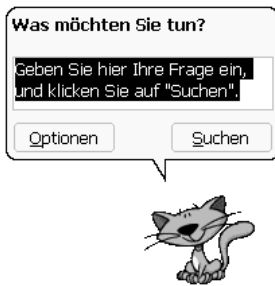


Abbildung 1.46:
Konservative
Warnungen



Wählen Sie ?|MICROSOFT EXCEL-HILFE, klicken Sie auf das abgebildete Symbol oder **[F1]** reaktiviert den Assistent, wenn er gerade nicht sichtbar ist. Dabei wird er auch gleich recht aktiv und fragt Sie, wobei genau er Ihnen helfen soll (Abbildung 1.47).

Abbildung 1.47:
Hilfestellung



Ebenso aktiv wird er übrigens auch, wenn Sie auf den Assistenten selbst klicken oder einfach **[F1]** drücken. In beiden Fällen tippen Sie eine umgangssprachliche Frage wie »Wie formatiere ich Zellen?« oder einfach »Zellen formatieren« ein und drücken **[↵]** oder klicken auf *Suchen*.

Eine Reihe zu Ihrer Frage passender Hilfethemen erscheint, aus denen Sie das Gewünschte auswählen; das *Hilfe*-Dialogfeld wird geöffnet und präsentiert einen mehr oder weniger ausführlichen Text zur betreffenden Aufgabe (Abbildung 1.48).

Die Option *Optionen* im Assistenten-Kontextmenü (mit der rechten Maustaste auf den Assistenten klicken) öffnet das *Office Assistent*-Dialogfeld (Abbildung 1.49).

Im Register *Optionen* können Sie das Verhalten des Assistenten im Detail festlegen. Zum Beispiel, ob **[F1]** den Office-Assistenten aufwecken oder das im folgenden Kapitel erläuterte konservativere *Hilfe*-Dialogfeld öffnen soll (*Den Office-Assistenten verwenden*), welche Arten von Tipps angezeigt oder ob die Tippliste geleert werden soll (*Meine Tipps zurücksetzen*).

Im Register *Katalog*, das auch durch den Befehl ASSISTENT AUSWÄHLEN... des Kontextmenüs aufgeschlagen wird, können Sie unter verschiedenen Assistenten einen auswählen (Abbildung 1.50).

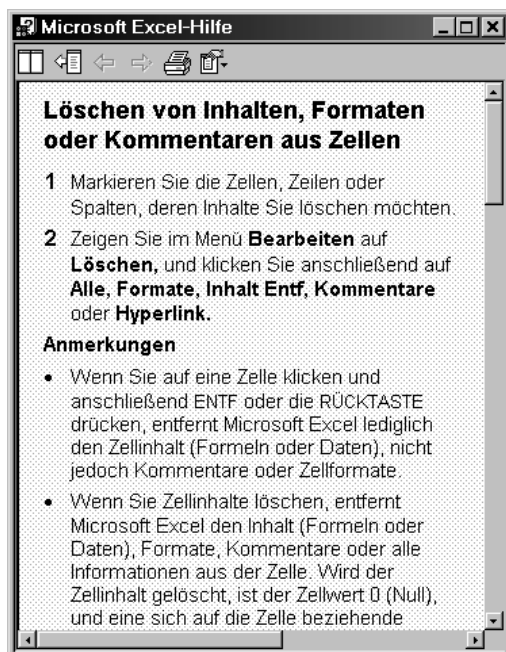


Abbildung 1.48:
Hilfetext

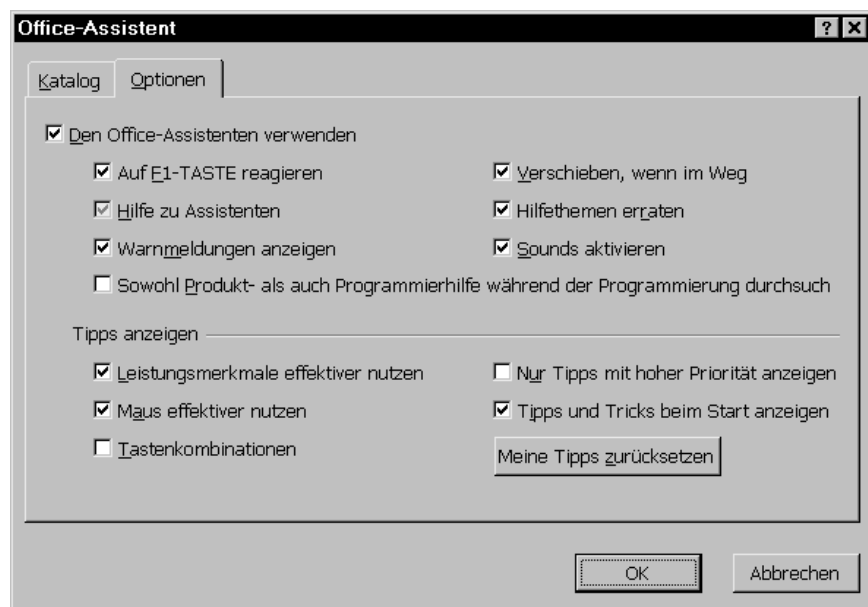
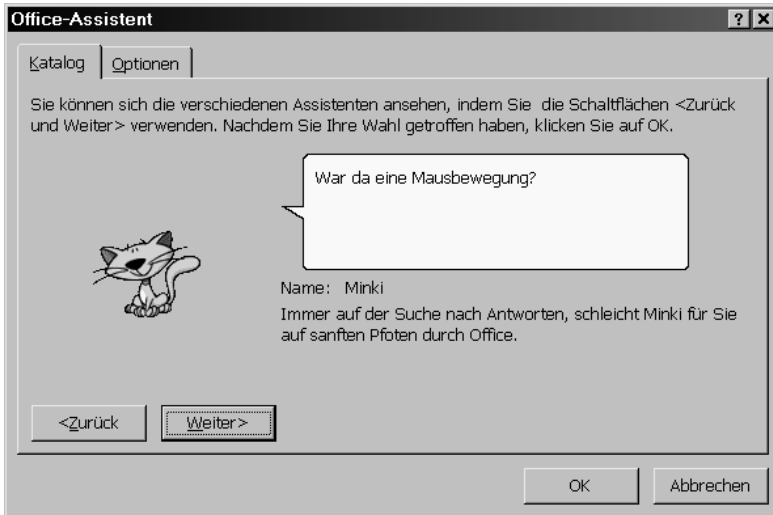


Abbildung 1.49:
Register »Optionen«

Die Assistenten-Galerie können Sie mit *<Zurück* bzw. *Weiter*> vor- oder rückwärts durchblättern.

Abbildung 1.50:
Register »Katalog«



1.5.3 Das Hilfefenster

Hilfe-Symboleiste

Klicken auf ein vom Assistenten angebotenes Thema (falls er nicht installiert ist: der Befehl ?|MICROSOFT EXCEL-HILFE öffnet das Hilfefenster, das mehrere Register enthält, ist das Fenster zu klein zur Anzeige aller Registerkarten, erscheint ein nach rechts/links weisendes Pfeilpaar, mit dem Sie zwischen den verschiedenen Registern hin- und herblättern können).

Das Hilfefenster enthält seine eigene Symbolleiste. Die Funktionen der darin enthaltenen Symbole von links nach rechts sind folgende



Das Anklicken dieses Symbols verkleinert und verschiebt das Excel-Fenster so, dass es neben dem Hilfefenster vollständig sichtbar ist (erneutes Anklicken stellt die ursprüngliche Größe des Excel-Fensters wieder her, sodass es womöglich teilweise vom Hilfefenster verdeckt wird). Ziehen Sie in diesem Zustand an der Trennungslinie, um eines der beiden Fenster zu vergrößern bzw. zu verkleinern, so wird die Größe des anderen Fensters automatisch entsprechend angepasst.



Die zweite Schaltfläche erweitert das Hilfefenster und zeigt im linken Teil eine Art Dialogfeld mit drei Registern an (erneutes Anklicken klappt den linken Teil wieder zu).



Mit der Schaltfläche *Zurück* (der Infotext, der erscheint, wenn sich der Cursor über der Schaltfläche befindet) springen Sie wieder zum zuletzt angeschauten Hilfetext zurück bzw. bei erneutem Anklicken zum Hilfetext davor.

Und mit *Vorwärts* aktivieren Sie genau umgekehrt wieder den danach angeschauten Hilfetext.



Drucken druckt den momentan angezeigten Hilfetext.



Optionen ist ein Menü, dessen Befehle vorwiegend eine Sammlung der soeben erläuterten Schaltflächen darstellt.



Blenden Sie mit der zweiten Schaltfläche den dialogfeldartigen linken Teil des Hilfefensters ein und aktivieren Sie das Register *Inhalt*, so wird darin die vollständige Liste aller verfügbaren Hilfethemen angezeigt (Abbildung 1.51).

Register Inhalt

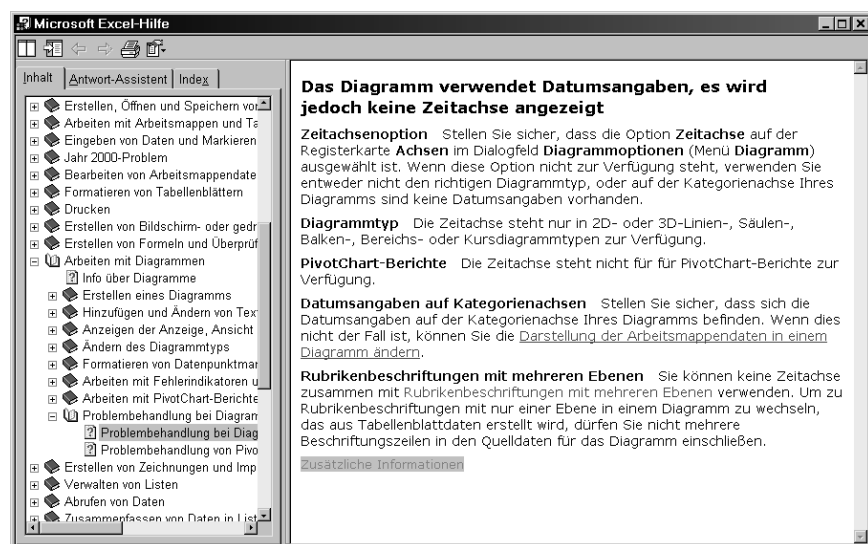


Abbildung 1.51:
Register »Inhalt«

Alle Bücher sind baumartig aufgebaut, von allgemeineren zu spezielleren Themen. Per Klick auf das »+« öffnen Sie ein Buch bzw. mit (»-«) schließen Sie es wieder. Sind Sie bei einem Fragezeichen-Symbol angelangt, erscheint beim Anklicken im rechten Fensterteil der zugehörige Hilfetext.

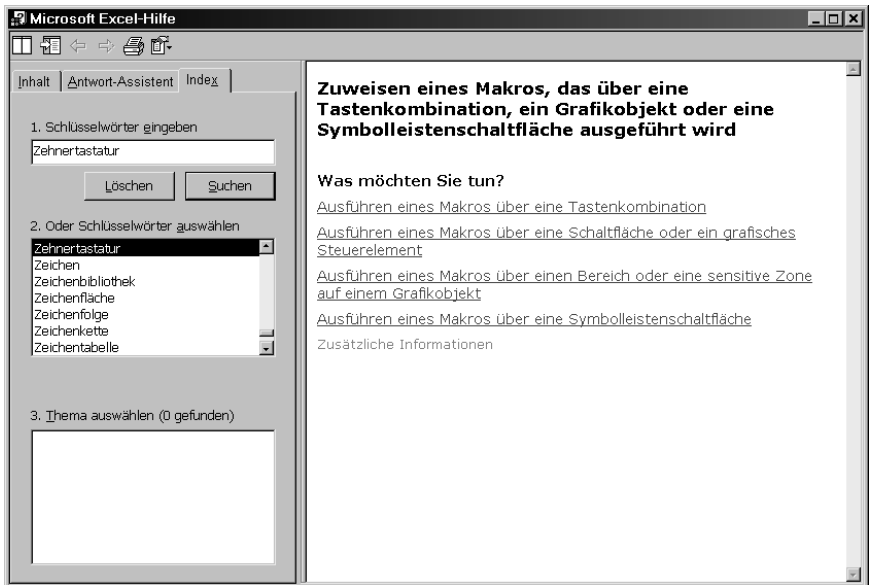
Hilfetexte enthalten häufig Querverweise auf Details oder verwandte Themen. Solche Querverweise sind immer unterstrichen und farblich hervorgehoben. Bewegen Sie die Maus zu einem solchen Verweis, ändert der Mauscursor seine Form und wird zu einer zeigenden Hand. Mit Anklicken springen Sie zum betreffenden Hilfetext.

Hyperlink-Schaltflächen

Das Register *Index* ermöglicht die gezielte Suche nach Hilfe zu einem bestimmten Begriff (Abbildung 1.52).

Register Index

Abbildung 1.52:
Register »Index«



Sie können den gesuchten Begriff unter 1. eintippen. Meist genügt es, die Anfangsbuchstaben einzugeben. Suchen Sie beispielsweise »Zellen«, wird nach Eingabe des »Z« in der Liste sofort der erste mit diesem Buchstaben beginnende Begriff hervorgehoben, beispielsweise »Zählen«. Nach Eingabe des zweiten Buchstabens »e« wird entsprechend der erste mit »Ze« beginnende Begriff hervorgehoben und so weiter.

Sie können stattdessen auch einen Begriff unter 2. auswählen. In beiden Fällen enthält 3. anschließend eine Liste aller zugehörigen Hilfethemen, aus denen Sie das gewünschte auswählen können.

Register *Antwort-Assistent*

Eine weitere Alternative bietet das Register *Antwort-Assistent*. Es ermöglicht Ihnen, im Eingabefeld umgangssprachliche Fragen zu stellen. Anschließend werden darunter alle zugehörigen Hilfethemen angezeigt (Abbildung 1.53).

Klicken Sie auf die Schaltfläche *Im Web suchen*, können Sie im rechten Fensterteil eine Frage eingeben und auf *Senden und ins Web gehen* klicken, woraufhin Ihr Webbrowser gestartet und auf einer speziellen Microsoft-Website nach einer dazu passenden Antwort gesucht wird.



Eine ähnliche Aufgabe wie der Antwort-Assistent erfüllt das listenartige Eingabefeld ganz rechts in der Menüleiste von Excel. Tippen Sie darin eine Frage wie »Wie formatiere ich Zellen?«, erscheint eine Liste mit zum Problem passenden Themen. Wählen Sie eines davon aus, wird das Hilfefenster mit diesem Thema aufgeschlagen.



Abbildung 1.53:
Register »Antwort-
Assistent«

In der Antwortliste mit »Web:...« aufgeführte Antworten befinden sich übrigens im Web, das heißt, Ihr Webbrowser wird gestartet und holt sich die Antwort von einer Microsoft-Webseite.

Öffnen Sie die Liste des Eingabefelds, erscheinen darin all Ihre vorhergegangenen Fragen, sodass Sie jederzeit die zugehörigen Antwortthemen griffbereit haben.

Teil 2 Tabellen

Kapitel 2: Grundlagen	107
Kapitel 3: Dateien	139
Kapitel 4: Selektieren	157
Kapitel 5: Editieren	163
Kapitel 6: Formatieren	213
Kapitel 7: Drucken	271
Kapitel 8: Arbeitsmappen und Vorlagen	297
Kapitel 9: Namen	331
Kapitel 10: Gliederungen	347
Kapitel 11: Funktionen	357

Das folgende Kapitel beschäftigt sich mit Tabellen, der Grundlage jeder Kalkulation mit Excel. Sie lernen, wie Tabellen aufgebaut sind, wie Zahlen und Texte in Zellen eingegeben und geändert werden und welche Hilfsmittel Excel dabei zur Verfügung stellt.

Anschließend erläutere ich alles rund ums Speichern und Öffnen von Tabellen, das Selektieren und Editieren beliebiger Tabellenbereiche und das Formatieren dieser Bereiche mit unterschiedlichen Schriftarten, Farben etc.

Danach geht es um flexible Möglichkeiten zum Ausdruck von Tabellen mit Fuß- und Kopfzeilen und die Festlegung des dabei zu verwendenden Seitenlayouts, um den Umgang mit Arbeitsmappen, die beispielsweise gemeinsam in Gruppen bearbeitet werden können, um das Ein- und Ausblenden und Fixieren von Fenstern.

Anschließend erläutere ich verschiedene Arbeitserleichterungen beim Umgang mit Tabellen: die Benennung von Zellen oder ganzen Bereichen, die Ihnen ermöglicht, Formeln wie $=C53+N4$ durch $=Brutto+MWSt$ zu ersetzen, und um den Einsatz von Gliederungen zum Ausblenden unwichtiger Tabellendetails auf Knopfdruck.

Den Abschluss dieses Teils bildet eine ausführliche Besprechung von Funktionen. Ich zeige Ihnen, wie Funktionen in der Praxis eingesetzt werden und gebe Ihnen Beispiele zum Einsatz der verschiedenen Funktionskategorien wie der mathematischen Funktionen, Suchfunktionen oder Datenbankfunktionen.

2 Grundlagen

Sie lernen nun zunächst, wie Zahlen und Texte in Tabellen eingegeben und editiert werden.

Das entscheidende Hilfsmittel jeder Kalkulation sind jedoch weder Zahlen noch Texte, sondern Formeln. Das können Formeln sein, die einfach ein paar Zellinhalte addieren, aber auch Formeln, die auf eine Testreihe komplexe statistische Verfahren anwenden.

Ich zeige Ihnen, wie Formeln aufgebaut sind und aus welchen Elementen sie bestehen. Danach geht es um die Möglichkeiten zur Eingabe von Formeln, sowohl mit der Tastatur als auch – oft einfacher – per Zeigen mit der Maus auf die zu addierenden oder sonstwie benötigten Zellen.

Excel kann Ihnen sowohl bei der Eingabe als auch bei der Korrektur von Formeln helfen, bei fehlerhaften Formeln teilweise sogar eigene Vorschläge zur Korrektur anbieten, die Sie nur noch bestätigen müssen, wenn Sie damit einverstanden sind.

Statt prinzipiell identische Formeln zur Addition von Zahlenreihen oder Ähnlichem immer wieder einzugeben, können Sie einmal eingegebene Formeln jederzeit kopieren. Dabei kommt es jedoch sehr leicht zu unerwarteten und fehlerhaften Ergebnissen, die nur zu vermeiden sind, wenn Sie sich mit den Internas von Excel auskennen.

Ich gehe daher sehr ausführlich auf Zellbezüge in Formeln ein und erläutere detailliert die verschiedenen Bezugsarten und ihre Bedeutung.

Anschließend sind Sie in der Lage, auch Tabellen mit recht komplexen Kalkulationen zu erstellen und darin problemlos Formeln zu kopieren oder zu verschieben – ohne danach unangenehme Überraschungen zu erleben, weil sich die kopierte bzw. verschobene Formel nicht wie erwartet verhält!

2.1 Dateneingabe

Ein Excel-Arbeitsblatt ist in Spalten und Zeilen unterteilt. Jede Spalten-/Zeilenkombination kennzeichnet eindeutig eine Zelle. Beispielsweise befindet sich die Zelle in der oberen linken Ecke in Zeile 1 und in Spalte A. Die Zell-

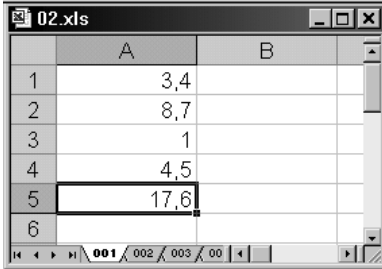
Zellen

referenz wird gebildet, indem beide Informationen hintereinandergeschrieben werden, zuerst die Spaltenbezeichnung, danach die Zeilennummer. Unsere Zelle besitzt daher die Zellreferenz A1. Die Zelle darunter befindet sich ebenfalls in Spalte A, aber in Zeile 2 und besitzt aus diesem Grund die Referenz A2.

In jede Zelle können Sie eine Zahl eintragen, aber auch beliebige Texte oder Formeln, Rechenvorschriften, die bestimmen, auf welche Weise der Wert in der Zelle ermittelt werden soll. Ein Beispiel sehen Sie in folgendem Bild (Abbildung 2.1).

Abbildung 2.1:
Abhängige und
unabhängige Zellen

=A1+A2+A3+A4



	A	B
1	3,4	
2	8,7	
3	1	
4	4,5	
5	17,6	
6		

In diese Tabelle habe ich zuerst vier Zahlen in die Zellen A1, A2, A3 und A4 eingetragen. Anschließend wurde nicht etwa in A5 die resultierende Summe 17,6 eingetippt – das hat Excel selbst erledigt. Stattdessen gab ich die in der Bearbeitungsleiste (unmittelbar unterhalb der Symbolleisten) angezeigte Formel ein: `=A1+A2+A3+A4`.

Sie bewirkt, dass Excel die Zahlen, die sich in den angegebenen vier Zellen befinden, addiert und in Zelle A5, in der sich die Formel befindet, das Ergebnis 17,6 anzeigt.

Wenn Sie in einer der vier unabhängigen Zellen A1 bis A4 einen neuen Wert eingeben, rechnet Excel die Formel in der abhängigen Zelle A5 *automatisch neu durch*. Ersetzen Sie beispielsweise die 1 in Zelle A3 durch eine 2, korrigiert Excel im gleichen Moment die in Zelle A5 enthaltene Summe und trägt selbstständig statt der alten Summe 17,6 den nun korrekten Wert 18,6 ein.

Um diese Abbildung nachzuvollziehen, geben Sie zuerst die vier Zahlen ein. Bewegen Sie den Zellcursor durch Anklicken der Zelle zu A1 und tippen Sie 3,4 ein (verwenden Sie dabei als Dezimaltrennzeichen unbedingt das Komma und nicht etwa einen Punkt!).



Sie schließen die Eingabe ab, indem Sie wahlweise a) auf das abgebildete und während der Eingabe in der Bearbeitungsleiste sichtbare Häkchen-Symbol klicken, b)  drücken oder c) eine beliebige Cursortaste betätigen.

Die -Taste beendet nicht nur die Eingabe, sondern setzt zusätzlich den Zellcursor auf die nächstuntere Zelle A2 (vorausgesetzt, im Register *Bearbeiten* des Befehls EXTRAS|OPTIONEN... ist das Kontrollkästchen *Markierung nach dem Drücken der Eingabetaste verschieben* aktiviert und als Verschieberichtung *Unten* eingestellt).

Benutzen Sie zum Beenden eine der vier Cursortasten, so wird die Eingabe ebenfalls beendet und der Cursor in die betreffende Richtung bewegt, zum Beispiel nach rechts zu B1, wenn Sie  drücken.

Geben Sie bitte auch die drei restlichen Zahlen ein. Stellen Sie bereits während der Eingabe fest, dass Sie sich vertan haben, dann brechen Sie die Eingabe ab, indem Sie  drücken oder das in der Bearbeitungsleiste während einer Eingabe sichtbare Kreuz-Symbol anklicken.



Ist die Eingabe bereits beendet, können Sie den Inhalt der betreffenden Zelle auch nachträglich löschen, indem Sie sie selektieren (Cursor dorthin steuern oder anklicken) und  oder  drücken.

2.2 Editieren

Vertippen Sie sich bei der Eingabe einer längeren Formel wie $=A1+A2+A3+A4$, dürfen Sie nicht einfach die Cursortasten verwenden, um sich in der Bearbeitungsleiste nach rechts oder links zu dem Zeichen zu bewegen, das korrigiert werden soll; die Betätigung einer Cursortaste würde die Eingabe ja wie erläutert beenden.

Korrekturen

Stattdessen klicken Sie auf das betreffende Zeichen und zwar wahlweise in der Bearbeitungsleiste oder in der Zelle selbst. In der Statusleiste steht daraufhin nicht mehr *Eingeben*, sondern *Bearbeiten*.

Sie befinden sich nun im Bearbeitungsmodus, in dem Sie wie von anderen Programmen her gewohnt mit den Cursortasten in der Eingabe herumwandern, Zeichen mit  oder  löschen oder einfügen können. Ist die Eingabe korrigiert, beenden Sie sie mit .

Sie können den Bearbeitungsmodus auch nachträglich einschalten, wenn eine Eingabe bereits beendet wurde. Dazu selektieren Sie die zu korrigierende Zelle mit einem Doppelklick. Der Cursor erscheint in der Zelle und Sie können ihn mit einem weiteren einfachen Klick exakt auf das fragliche Zeichen positionieren.

Mit der Tastatur schalten Sie den Bearbeitungsmodus mit  ein, sowohl während der Eingabe als auch bei der nachträglichen Korrektur eines Zellinhalts.

Markieren

Im Bearbeitungsmodus können Sie auch umfangreichere Textteile wie »Dies ist ein Test« schnell und einfach manipulieren. Mit der Maus ziehen Sie bei gedrückter linker Maustaste über die benötigten Zeichen oder Wörter, die daraufhin invers hervorgehoben werden. Mit der Tastatur halten Sie die -Taste gedrückt, während Sie den Cursor über den Text bewegen.

Den markierten Text können Sie anschließend mit  löschen oder mithilfe dreier Symbole der Standardleiste bzw. mit den zugeordneten Menübefehlen verschieben oder kopieren.



Das Ausschneiden-Symbol (Menübefehl BEARBEITEN|AUSSCHNEIDEN) schneidet den markierten Text ähnlich wie die -Taste aus, befördert ihn jedoch zur weiteren Verwendung in die Zwischenablage.



Bewegen Sie den Cursor nach dem Ausschneiden zu einer anderen Textposition oder zu einer anderen Zelle und klicken Sie das Einfügen-Symbol an (BEARBEITEN|EINFÜGEN), so wird der Inhalt der Zwischenablage dort eingefügt. Mehrfaches Anklicken fügt den Text mehrfach ein, kopiert ihn also.



Das Kopieren-Symbol (BEARBEITEN|KOPIEREN) schneidet markierten Text nicht aus, befördert ihn jedoch ebenso wie das Ausschneiden-Symbol in die Zwischenablage, sodass Sie ihn anschließend mit dem Einfügen-Symbol an einer anderen Position einfügen und auf diese Weise kopieren können.

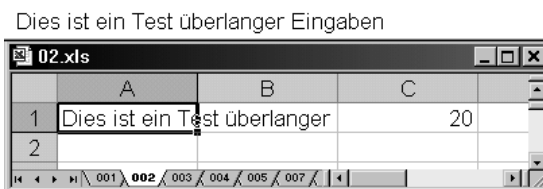
2.3 Texte, Zahlen und Symbole

Überlange Texte

Passt eine Eingabe wie »Dies ist ein Test überlanger Eingaben« nicht in eine Zelle wie A1, benutzt Excel auch benachbarte leere Zellen zur Anzeige. Der Text erstreckt sich dann zwar über die Zellen A1, B1 und C1, gehört aber dennoch zu Zelle A1.

Sie sehen das, wenn sich der Zellcursor auf einer der zur Anzeige mitverwendeten Zellen befindet: als Zellinhalt wird in der Bearbeitungszeile nichts angezeigt – B1 und C1 sind weiterhin leer. Bewegen Sie den Cursor jedoch zu A1, wird als Zellinhalt der komplette Text angezeigt, nicht nur der Teil, der in der Zelle selbst sichtbar ist (Abbildung 2.2).

Abbildung 2.2:
Überlange
Texteingaben



Wenn Sie entsprechend der Abbildung in C1 eine 20 eingeben, zeigt sich sofort, dass B1 und C1 in Wirklichkeit leer sind. Da C1 nun nicht mehr leer ist, benutzt Excel ab sofort nur noch die tatsächlich leere Zelle B1 zur Anzeige des überlangen Inhalts von A1. Löschen Sie den Inhalt von Zelle C1, wird die nun leere Zelle sofort wieder zur Anzeige mitbenutzt.

Um längere Texte übersichtlicher zu gestalten, können Sie mit **Alt** + **↵** einen Zeilenumbruch einfügen (Abbildung 2.3).

*Tabulator/Zeilen-
umbruch*

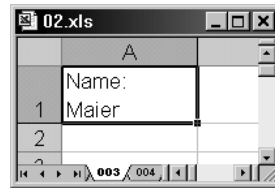


Abbildung 2.3:
Zeilenumbruch

In A1 gab ich den Text »Name:Maier« ein, wobei ich nach dem Doppelpunkt mit **Alt** + **↵** einen Zeilenumbruch einfügte und Excel die Zeilenhöhe automatisch anpasste.

Bei überlangen Zahlen verhält sich Excel völlig anders. Geben Sie in B2 bitte 120000000000 (12 mit folgenden zehn Nullen) ein. Danach ist die Zahl zwar in der Bearbeitungszeile immer noch sichtbar, in der Zelle selbst wird jedoch »1,2E+11« angezeigt.

»E«, »-«, »%«

Zur Erläuterung: Zahlen, die aufgrund ihrer Länge schlecht darzustellen sind, wandelt Excel in das Exponentialformat um. 1,2 ist die Mantisse, +11 der Exponent zur Basis 10. 1,2E+11 bedeutet daher

$$1,2E+11 = 1,2 * 10^{11}.$$

Die Exponentialschreibweise können Sie auch selbst bei Eingaben benutzen und statt 100000000000 (1 mit zehn Nullen) einfacher 1E+10 eingeben.

Sollte eine Zahl trotz dieser Kurzschreibweise immer noch zu lang für die betreffende Zelle sein, zeigt Excel einfach eine Reihe von Nummernzeichen an, etwa so: »#####«. In diesen Fällen vergrößert man wie später gezeigt die Spaltenbreite.

Natürlich können Sie bei der Eingabe von Zahlen auch ein Vorzeichen verwenden. Also statt »3,4« auch »+3,4« eingeben, was jedoch im Gegensatz zur negativen Zahl »-3,4« überflüssig ist. Eine Zahl darf auch mit einem Prozentzeichen enden, um Excel klarzumachen, dass es sich bei der Eingabe »3%« nicht um die Zahl 3, sondern um drei Prozent handelt, also um 0,03.

**Falsche
Textinterpretation**

Excel erkennt, ob Sie einen Text oder eine Zahl eingeben. Zahlen werden sofort rechtsbündig in der betreffenden Zelle ausgerichtet, im Gegensatz zu Texteingaben wie »Hallo«, die linksbündig ausgerichtet werden.

Allerdings klappt das nicht immer. Stellen Sie sich eine Tabelle vor, die in Spalte A untereinander die Jahreszahlen 2009, 2010, 2011, 2012 bis 2019 enthält. Und in Spalte B die zugehörigen Jahresumsätze Ihrer Firma. Excel wird die Jahre als Zahlen interpretieren und rechtsbündig ausrichten – auch dann, wenn Ihnen die linksbündige Ausrichtung, die bei Texten üblich ist, viel besser gefallen würde.

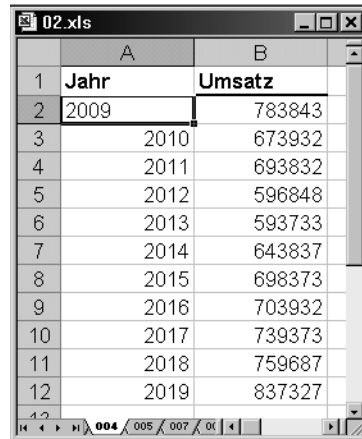
In solchen Fällen können Sie Excel dazu zwingen, die betreffende Eingabe als Text zu behandeln. Schließen Sie die Zahl in Anführungszeichen ein und beginnen Sie die Eingabe mit einem Gleichheitszeichen. Geben Sie nicht 2009 ein, sondern

= "2009"

Die Anführungszeichen selbst erscheinen nicht in der Zelle. Excel erkennt daran jedoch, dass Sie einen Text eingeben wollen, und richtet »2009« daher wie jeden Text linksbündig aus – im Gegensatz zu den Zahlen darunter (Abbildung 2.4).

Abbildung 2.4:
Als Texte
interpretierte Zahlen

= "2009"



	A	B
1	Jahr	Umsatz
2	2009	783843
3	2010	673932
4	2011	693832
5	2012	596848
6	2013	593733
7	2014	643837
8	2015	698373
9	2016	703932
10	2017	739373
11	2018	759687
12	2019	837327

Sie können die Texteingabe auch mit einem einfachen Anführungszeichen einleiten und

'2009

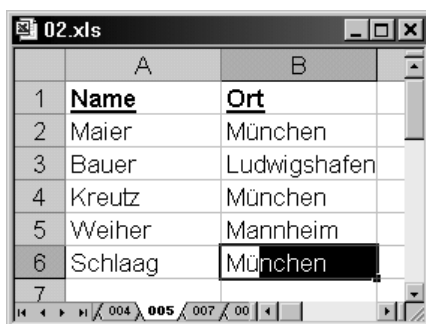
eingeben. Der Text wird erkannt und linksbündig ausgerichtet und das Anführungszeichen selbst nicht angezeigt.

Alternativ zur erläuterten Technik können Sie den betreffenden Zellen das Format Text zuweisen (siehe Kapitel, »6.4.1 Zahlenformate«).



2.4 AutoEingabe

Die einzelnen Spalten einer Tabelle enthalten häufig immer wiederkehrende Einträge, die Spalte »Ort« einer Adressliste beispielsweise mehrfach die Einträge »München« und »Ludwigshafen« (Abbildung 2.5).



	A	B
1	Name	Ort
2	Maier	München
3	Bauer	Ludwigshafen
4	Kreutz	München
5	Weiher	Mannheim
6	Schlaag	München
7		

Abbildung 2.5:
Wiederholte
Eingaben

Um diese wiederkehrenden Eingaben zu vereinfachen, sollten Sie sich vergegenwärtigen, dass im Register *Bearbeiten* des Befehls EXTRAS|OPTIONEN... das Kontrollkästchen *AutoVervollständigen für Zellwerte aktivieren* aktiviert ist.

Dann genügt es meist, die ersten Buchstaben des gewünschten Eintrags einzutippen. Excel prüft, ob sich in der gleichen Spalte ein Eintrag befindet, der mit denselben Buchstaben beginnt und ergänzt Ihre Eingabe automatisch, falls das der Fall ist.

Die Eingabe »M« würde daher automatisch zu »München« ergänzt, wenn sich irgendwo in der gleichen Spalte dieser Text befindet. Allerdings nur, wenn das »M« *eindeutig* ist und die Spalte keinen anderen Eintrag wie »Mannheim« enthält, der mit dem gleichen Buchstaben beginnt! Dann würde die Eingabe erst nach der Eingabe des zweiten Buchstabens ergänzt, nach »Mü« zu »München«, nach Eingabe von »Ma« zu »Mannheim«.

Alternativ zu dieser AutoEingabe können Sie eine Zelle auch mit der rechten Maustaste anklicken und im zugehörigen Kontextmenü AUSWAHLLISTE... wählen (Abbildung 2.6).

Wie Sie sehen, enthält diese Liste alle Einträge der betreffenden Spalte, aus der Sie sich nun den gewünschten aussuchen können.

Abbildung 2.6:
Auswahlliste



2.5 Formeln

Formeln werden mit einem Gleichheitszeichen eingeleitet. Was diesem Zeichen folgt, ist höchst unterschiedlich:

- ➔ Konstanten (13, 124)
- ➔ Zellbezüge (A1, D5)
- ➔ Namen (*AUSGABEN*, *BWK*)
- ➔ Funktionen (*SIN()*, *SUMME()*)
- ➔ Operatoren (+ - / *)

Eine Konstante ist ein fester Wert, eine bestimmte Zahl.

Ein Operator ist ein Zeichen wie »+« oder »-«, das angibt, nach welcher Vorschrift die Elemente einer Formel, die Operanden (zum Beispiel zwei Konstanten wie 34 und 21), miteinander zu verknüpfen sind. Zum Beispiel verknüpft »+«, der Additionsoperator, zwei Operanden durch eine Addition, bildet also ihre Summe.

Die restlichen mathematischen Operatoren von Excel:

Tabelle 2.1:
Mathematische
Operatoren

Operator	Funktion
^	Potenzierung
+	Addition
-	Subtraktion (Negation)

Operator	Funktion
*	Multiplikation
/	Division
%	Prozent

Tabelle 2.1:
Mathematische
Operatoren
(Forts.)

Das Zeichen »-« hat eine doppelte Bedeutung. Einmal verknüpft es zwei Konstanten durch die Rechenvorschrift Subtraktion: 3-4. Und zum anderen steht es für ein negatives Vorzeichen.

»-«

Sie können beliebig lange und komplexe Formeln eingeben, wie:

=3+4*2,5/2

Über die Berechnungsreihenfolge entscheiden die Prioritäten der einzelnen Operatoren. Gemäß der berühmten Punkt-vor-Strich-Regel werden zum Beispiel Multiplikationen und Divisionen vor Additionen oder Subtraktionen ausgeführt. In dieser Formel multipliziert Excel daher zunächst 4 mit 2,5 und dividiert das Resultat 10 anschließend durch 2. Zu diesem neuen Zwischenergebnis 5 wird zum Schluss die 3 addiert.

*Berechnungs-
reihenfolge*

Die höchste Priorität besitzt die Potenzierung (Multiplikation einer Zahl mit sich selbst). Daher wird in der Formel

=4*2^3

zuerst der Ausdruck 2^3 ausgewertet und 2 dreimal mit sich selbst multipliziert: 2*2*2. Danach wird das Resultat 8 mit 4 multipliziert.

Klammern besitzen eine höhere Priorität als alle Operatoren. Ein geklammerter Ausdruck wird daher immer zuerst ausgewertet. Wollen Sie zum Beispiel zuerst 4 und 2 addieren und anschließend das Ergebnis mit 5 multiplizieren, benötigen Sie unbedingt eine Klammer:

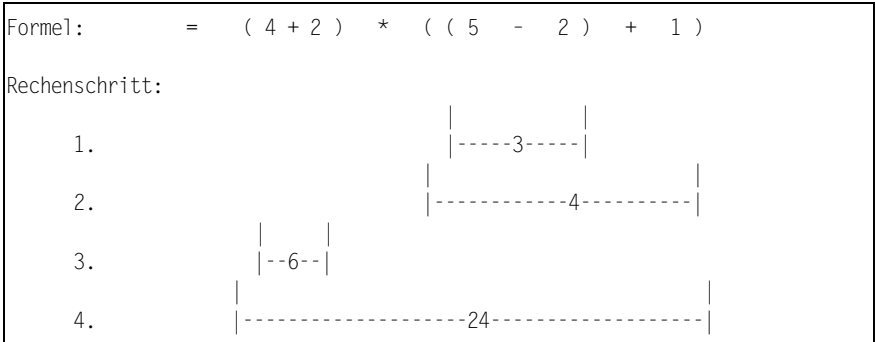
Klammern

=(4+2)*5

Ohne die Klammer würde Excel zuerst 2 mit 5 multiplizieren und anschließend die 4 addieren.

Klammern können beliebig ineinander verschachtelt werden (Abbildung 2.7).

Abbildung 2.7:
Auswertungs-
reihenfolge

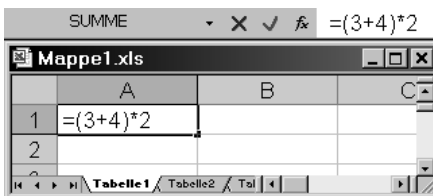


Excel beginnt eine Auswertung immer mit der innersten Klammer, hier also mit dem Ausdruck 5-2 mit dem Resultat 3. Nun wird die äußere Klammer ausgewertet und zu dieser 3 eine 1 addiert, Ergebnis: 4. Die Multiplikation wird noch nicht ausgeführt, sondern stattdessen die Addition 4+2, die ja ebenfalls geklammert ist, und nun wird das Zwischenergebnis 6 der linken mit dem Resultat 4 der beiden rechten Klammern multipliziert.

2.5.1 Hilfestellungen

Beginnen Sie, eine Formel einzugeben, indem Sie das einleitende »=« eintippen, so ändert sich der Inhalt der Bearbeitungsleiste (Abbildung 2.8).

Abbildung 2.8:
Formeleingabe



Sie enthält nun zusätzlich zwei Symbole, mit denen Sie die Formeleingabe wieder abbrechen (Kreuz) bzw., wenn Sie fertig sind, beenden (Häkchen) können. Einfacher ist es allerdings, dafür schlicht bzw. zu drücken.



Geben Sie eine Formel wie =Summe(C1:C10) ein (sie addiert die in den Zellen C1, C2, C3, ..., C9 und C10 enthaltenen Werte), erscheint nach dem Eintippen von =Summe (neben der aktuellen Zelle automatisch ein kleines Feld, das Ihnen zeigt, welche Argumente Sie der betreffenden Funktion nun angeben müssen (siehe Kapitel 2.5.5, »Funktionen«).

Excel unterstützt Sie, wenn Sie während der Eingabe der Formel offensichtliche Fehler begehen, und macht entsprechende Korrekturvorschläge. Geben Sie beispielsweise statt =Summe(C1:C10) versehentlich =Summe(C:1:C10) ein, entdeckt Excel diesen Fehler nach dem Beenden der Eingabe und schlägt als Korrektur die korrekte Formel vor (Abbildung 2.9).

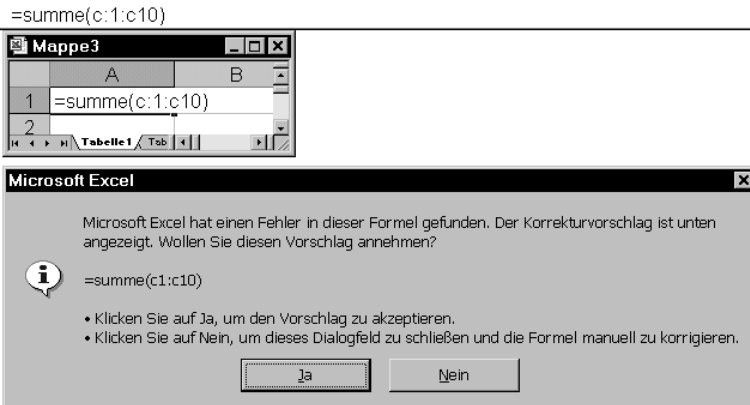


Abbildung 2.9:
AutoKorrektur

Ja korrigiert Ihre fehlerhafte Eingabe entsprechend dem Vorschlag, *Nein* lässt sie unverändert.

Ist eine Formel trotz dieser Hilfe fehlerhaft, erscheint in der Zelle statt eines Zahlenwerts ein Fehlerwert, ein Text mit einer Fehlermeldung wie *#NAME?* (Abbildung 2.10).

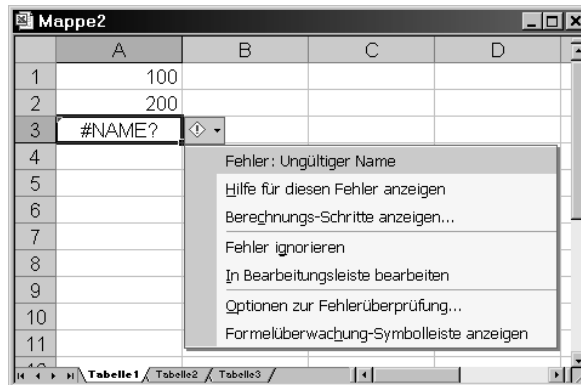


Abbildung 2.10:
Fehler-Smart Tag

Zusätzlich befindet sich am oberen linken Zellrand ein um Aufmerksamkeit heischendes kleines Dreieck. Tun Sie ihm den Gefallen, auf die Zelle zu klicken, so erscheint neben ihr das abgebildete Fehler-Smart Tag, ein deutlich hervorgehobenes Ausrufezeichen-Symbol.

Klicken Sie darauf, so öffnet sich das abgebildete Kontextmenü, das zur Fehlerbehebung nützliche Befehle enthält.

Einige davon sind recht schlicht. Beispielsweise aktiviert der Befehl *In Bearbeitungsleiste bearbeiten* die Bearbeitungsleiste, damit Sie die Formel darin editieren können – was das Gleiche ist, als würden Sie selbst auf diese Leiste klicken.

Sinnvoller ist beispielsweise die Option, im Hilfefenster die zu dieser Fehlermeldung passende Hilfestellung aufzuschlagen oder die später in diesem Buch erläuterte *Formelüberwachungs*-Symbolleiste einzublenden.



Wie ich in Anhang 7, »Excel individuell konfigurieren«, erläutere, trägt Excel manchmal sogar selbstständig eine Formel in eine Zelle ein. Passt Ihnen diese Automatik nicht, können Sie sie wie im genannten Kapitel erläutert deaktivieren.

2.5.2 Zellbezüge

Nahezu jede wirklich sinnvolle Formel enthält unter anderem Zellbezüge, die die Verbindung zu anderen Zellen herstellen. Ein Zellbezug besteht aus den Koordinaten der betreffenden Zelle, der Zellreferenz.

Zellbezüge in Formeln sind Stellvertreter für Werte, die sich aus dem Inhalt anderer Zellen ergeben. Beim Ermitteln des Wertes einer Formel ersetzt Excel alle darin enthaltenen Bezüge durch die aktuellen Inhalte der betreffenden Zellen. Ändert sich irgendwann der Inhalt einer dieser Zellen, ändert sich auch der Wert der davon abhängigen Zelle.

Probieren wir's aus. Löschen Sie mit Entf nacheinander die Inhalte aller bisher benutzten Zellen, tragen Sie anschließend in A2 die Formel

=2*A1

ein und beenden Sie die Eingabe mit ↵ (Sie können auch =2*a1 eingeben, da Excel bei Zellreferenzen nicht zwischen Groß- und Kleinschreibung unterscheidet): Excel zeigt jetzt in A2 den Wert 0 an, da diese Formel festlegt, dass sich der Wert von A2 aus dem in A1 enthaltenen Wert ergibt, multipliziert mit 2. Da A1 momentan leer ist (entspricht dem Wert 0), ermittelt Excel als Resultat der Formel 0.

Aktualisierung

Tragen Sie jetzt in A1 den Wert 123 ein. Da sich der Inhalt der im Zellbezug angegebenen Zelle verändert hat, rechnet Excel die Formel neu durch und zeigt als neues Resultat in A2 die Zahl »246« an.

Beachten Sie bitte, dass in der Zelle A2 zwar das Ergebnis 246 der Formel angezeigt wird, in der Bearbeitungsleiste als echter Zellinhalt jedoch weiterhin die Formel selbst (Abbildung 2.11).

Bezugseingabe im Zeigemodus

Statt durch Eintippen können Sie Zellbezüge wesentlich einfacher durch Zeigen auf die betreffenden Zellen eingeben: Sie beginnen die Formeleingabe mit dem Gleichheitszeichen »=«. Excel erkennt den Beginn einer Formeleingabe und zeigt in der Statusleiste *Eingeben* an.

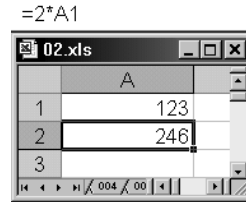


Abbildung 2.11:
Formel in A2 mit
Bezug auf Zelle A1

Klicken Sie nun eine Zelle an, zeigt Excel in der Statusleiste *Zeigen* an. Sie befinden sich nun im Zeigemodus, in dem die Zelle, auf die Sie klicken, durch einen Laufrahmen hervorgehoben und ihre Koordinaten in die Formel eingefügt werden.

Benutzen Sie die Tastatur, dann steuern Sie den Zellcursor einfach mit den Cursortasten zur betreffenden Zelle, um ihren Bezug einzufügen (funktioniert jedoch nur, wenn Sie die Eingabe per Eintippen von »=« einleiten, und nicht per Anklicken dieses Symbols).



Um beispielsweise die Formel $=A1+A2+A3+A4$ in A5 einzugeben, gehen Sie zu A5 und geben zunächst nur das Gleichheitszeichen ein. Klicken Sie danach auf A1. Der Zellcursor bleibt auf A5 und die selektierte Zelle A1 wird durch einen Laufrahmen hervorgehoben. Die Bearbeitungsleiste enthält nun $=A1$ «, außer dem Gleichheitszeichen also auch die von Excel eingefügten Koordinaten der selektierten Zelle.

Drücken Sie die +-Taste. Klicken Sie danach auf A2. Der aktuelle Stand der Formel ist:

$=A1+A2$

Drücken Sie erneut +, klicken Sie auf A3, drücken Sie + und klicken Sie auf A4. Die Formel lautet jetzt wie gewünscht $=A1+A2+A3+A4$ und Sie können die Eingabe wie gewohnt mit  beenden.

Ist im Register Bearbeiten des Befehls EXTRAS|OPTIONEN... das Kontrollkästchen Direkte Zellbearbeitung aktiviert, hebt ein Doppelklick auf eine Formelzelle, die einen Zellbezug enthält, die Bezugszellen optisch hervor.



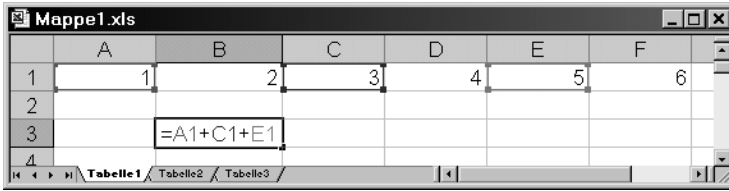
In Bild 2.11 würde ein Doppelklick auf A2 die Zelle A1 hervorheben. Befinden sich mehrere Zellbezüge in einer Formel, werden durch den Doppelklick auf die Formelzelle alle Bezugszellen hervorgehoben (Abbildung 2.12).

In dieser Tabelle enthalten die Zellen A1, B1, C1, D1, E1 und F1 je eine Zahl. B3 enthält die Formel

$=A1+C1+E1$.

Abbildung 2.12:

Hervorheben der
Bezugszellen mit
einem Doppelklick



Sie addiert jede zweite dieser Zahlen. Durch den Doppelklick auf die Formelzelle B3 werden alle drei Bezugszellen A1, C1 und E1 hervorgehoben.

Und zwar werden die betreffenden Zellen mit Rahmen in unterschiedlichen Farben hervorgehoben, entsprechend den nun ebenfalls mit unterschiedlichen Farben markierten Bezügen A1, C1 und E1 in der Formel. Sie sehen daher auf einen Blick anhand der korrespondierenden Farben, welche Zellen zu welchem Bezug gehören.

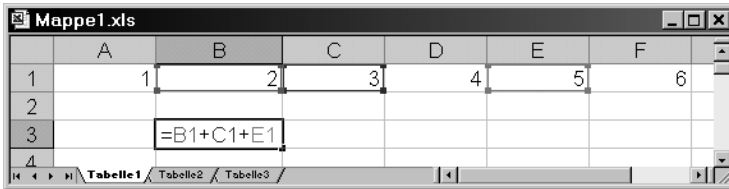


Sie können die hervorgehobenen Bezüge übrigens bequem korrigieren. Wenn Sie auf einen der Hervorhebungsrahmen klicken, beispielsweise jenen um A1 herum, können Sie ihn bei gedrückter Maustaste zu einer anderen Zelle verschieben, beispielsweise zu B1, und ihn dort fallenlassen.

Nach dem Verschieben des Rahmens wird in der Formel der Bezug A1 entsprechend durch B1 ersetzt (Abbildung 2.13)

Abbildung 2.13:

Bezüge ersetzen



2.5.3 Bezugsarten

Relative Bezüge

Bei der Angabe von Zellreferenzen haben Sie die Wahl unter drei verschiedenen Bezugsformaten: relative, absolute und gemischte Angaben.

Bisher haben wir relative Bezüge verwendet. Was das bedeutet, sehen Sie, wenn Sie EXTRAS|OPTIONEN... wählen und im Register *Allgemein* unter *Einstellungen* die momentan deaktivierte Option *Z1S1 Bezugsart* aktivieren. Angewandt auf Bild 2.11 wird die in A2 enthaltene Formel nicht mehr in der Form $=2*A1$ angezeigt, sondern (Abbildung 2.14) als

$=2*Z(-1)S.$

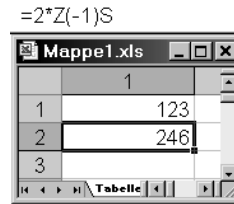


Abbildung 2.14:
Das Bezugsformat
Z1S1

Nur in dieser Darstellungsform sehen Sie die *wahre* Bedeutung relativer Bezüge! Die Zahlen in Klammern geben den *Abstand* der Bezugzelle von der Formelzelle in Spalten beziehungsweise Zeilen an. »Z(-1)« bedeutet, dass sich die Bezugzelle eine Zeile oberhalb der Formelzelle befindet. Und »S«, dass sich die Zelle in der gleichen Spalte wie die Formelzelle befindet.

Allgemein ist das Format

Z(Wert1)S(Wert2)

so aufzufassen: Positive Zahlen für »Wert1« geben an, wie viele Zeilen unterhalb der Formelzelle sich die Bezugzelle befindet, negative, wie viele Zeilen darüber.

Positive Zahlen für »Wert2« geben entsprechend an, wie viele Spalten sie sich rechts neben der Formelzelle befindet, und negative Werte, wie viele Spalten links davon.

Die Darstellung des Koordinatensystems des Arbeitsblattes hat sich ebenfalls geändert. Spalten sind nun nicht mehr durch Buchstaben gekennzeichnet, sondern wie die Zeilen ebenfalls fortlaufend als 1, 2, 3 usw. nummeriert.

Schalten Sie nun bitte wieder die ursprüngliche Darstellungsform ein, deaktivieren Sie also das Z1S1-Bezugsformat.

Absolute Bezüge geben Sie ein, indem Sie vor beiden Koordinaten, der Spalten- und der Zeilenangabe, je ein Dollarzeichen einfügen. Ändern Sie die in A2 enthaltene Formel entsprechend in

=2*\$A\$1.

Auch mit dieser absoluten Bezugsangabe wird in A2 der Wert 246 angezeigt. Auf den ersten Blick scheint es daher keinen Unterschied zwischen einem relativen und einem absoluten Bezug zu geben.

Auch beim Einfügen, Löschen oder Verschieben von Zellen gibt es zunächst keine Unterschiede zwischen beiden Bezugsarten. Egal, welche Bezugsart Sie verwenden, bei nahezu allen Editierungen des Arbeitsblattes passen sich Bezüge automatisch an die neue Situation an.

Absolute Bezüge

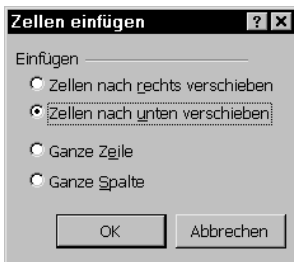
**Formelzelle
verschieben**

Ein Beispiel: Gehen wir erneut davon aus, dass A1 den Wert 123 enthält und A2 die Formel $=2*A1$ (siehe Bild 2.11).

Angenommen, Sie würden gerne in der obersten Zeile des Arbeitsblattes die Überschrift »Ein Test« eingeben. Und zwar ausgerechnet in der linken Ecke, die aber leider schon belegt ist. Dann müssen Sie die Inhalte der Zellen A1 und A2 eine Zeile nach unten verschieben in die Zellen A2 und A3.

Bewegen Sie dazu den Zellcursor zu A1 und wählen Sie EINFÜGEN|ZELLEN... (Abbildung 2.15).

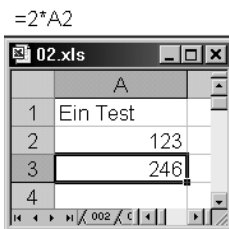
Abbildung 2.15:
Zellen nach unten
verschieben



Lassen Sie die vorselektierte Option *Zellen nach unten verschieben* selektiert und drücken Sie $\leftarrow$. Excel fügt an der aktuellen Zellcursorposition eine Zelle ein und verschiebt A1 und A2 nach unten.

Geben Sie anschließend in der nun leeren Zelle A1 die gewünschte Überschrift ein (Abbildung 2.16).

Abbildung 2.16:
Angepasste
Zellreferenz



Die Formel befindet sich durch das Verschieben nach unten nun in Zelle A3. Theoretisch dürfte darin jetzt keine 246 mehr angezeigt werden. Denn die Formel lautete $=2*A1$. Und A1 enthält ja jetzt keine Zahl mehr, sondern einen Text.

Da Excel einen Text jedoch verständlicherweise nicht mit einer Zahl wie 2 multiplizieren kann, müsste eigentlich eine Fehlermeldung erscheinen.

Stattdessen wird weiterhin 246 angezeigt. Excel hat den Bezug an die veränderte Situation angepasst, wovon Sie sich überzeugen können, wenn Sie den

Zellcursor zu A3 bewegen und sich die veränderte Formel in der Bearbeitungsleiste anschauen: Sie lautet nicht mehr $=2*A1$, sondern $=2*A2$ (siehe Bild 2.16).

Excel passte den Bezug automatisch an die durch das Einfügen der Leerzelle veränderte Situation an und sorgt dafür, dass er sich auf die gleiche Zelle wie zuvor bezieht.

Das Gleiche gilt, wenn Sie eine Formel nach oben, rechts oder links verschieben oder nur die Formel- oder die Bezugszelle.

*Bezugszelle
verschieben*

Probieren wir das aus, indem wir A2 eine Spalte nach rechts verschieben: Gehen Sie mit dem Zellcursor zu A2 und wählen Sie wieder EINFÜGEN|ZELLEN... . Selektieren Sie diesmal jedoch *Zellen nach rechts verschieben*. Excel verschiebt den Inhalt der Zelle nach rechts, in Zelle B2.

Gehen Sie mit dem Cursor zu A3 und schauen Sie sich die erneut angepasste Formel an (Abbildung 2.17).

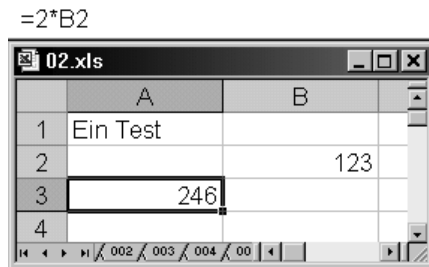


Abbildung 2.17:
Verschieben der
Bezugszelle

Der zuvor in A2 enthaltene Wert 123 befindet sich nun in Zelle B2 und Excel wandelte die Formel $=2*A2$ entsprechend in $=2*B2$ um.

Ändern Sie die Formel nun in $=2*\$B\$2$ ab, verwenden Sie also einen absoluten Bezug. Gehen Sie danach zu A2, wählen Sie BEARBEITEN|ZELLEN LÖSCHEN... und aktivieren Sie im zugehörigen Dialogfeld die Option *Zellen nach links verschieben*.

Excel verschiebt B2 nach A2. Die in A3 enthaltene Formel lautet anschließend nicht mehr $=2*\$B\$2$, sondern $=2*\$A\$2$. Excel passt den absoluten Bezug wie zuvor den relativen Bezug korrekt an die geänderte Situation an.

Allgemein: Beim Einfügen, Löschen oder Verschieben von Zellen werden immer beide Bezugsformen angepasst, relative und absolute. Der Unterschied zeigt sich erst beim in der Praxis extrem häufigen Kopieren von Formeln, bei dem Bezüge nicht angepasst werden!

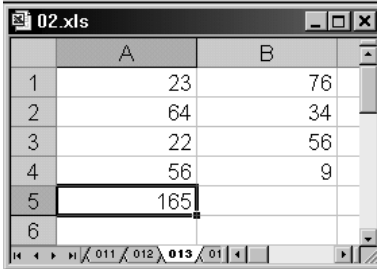


2.5.4 Formeln kopieren

Gehen wir von folgender Tabelle aus (Abbildung 2.18).

Abbildung 2.18:
Summieren von
Spalten

=A1+A2+A3+A4



	A	B
1	23	76
2	64	34
3	22	56
4	56	9
5	165	
6		

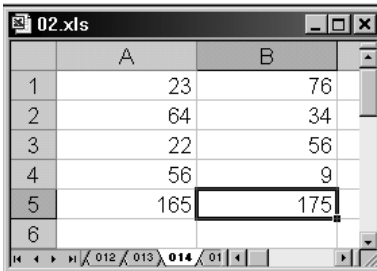
Zwei Spalten A und B enthalten jeweils vier zu addierende Werte und Zelle A5 folgende Formel:

=A1+A2+A3+A4

Gehen Sie mit dem Zellcursor zu A5 und wählen Sie BEARBEITEN|KOPIEREN. A5 ist nun von einem Laufrahmen umgeben. Gehen Sie zu B5, dem Kopierziel, und drücken Sie , damit Excel die in A5 enthaltene Formel in Zelle B5 kopiert. Das Resultat sehen Sie in Bild 2.19.

Abbildung 2.19:
Kopieren relativer
Formeln

=B1+B2+B3+B4



	A	B
1	23	76
2	64	34
3	22	56
4	56	9
5	165	175
6		

Als Inhalt von Zelle B5 wird die Formel =B1+B2+B3+B4 angezeigt, obwohl diese Formel durch Kopieren der Formel =A1+A2+A3+A4 entstand! Die Formel wurde beim Kopieren modifiziert und an die veränderte Situation angepasst.

Ursache dafür ist die Verwendung relativer Bezüge. Beide Formeln sind in Wahrheit trotz der unterschiedlichen Anzeige absolut identisch miteinander und besagen dank der relativen Bezüge, die ja in Wahrheit *Entfernungsangaben* sind: »Addiere die in der gleichen Spalte, aber vier bzw. drei bzw. zwei bzw. eine Zeile darüber enthaltenen Werte.«

Wenn Sie erneut auf das Z1S1-Anzeigeformat umschalten, sehen Sie, dass die relativen Bezüge in allen drei Formelzellen tatsächlich identisch sind. Sowohl in A5 als auch in B5 wird die gleiche Formel angezeigt:

=Z(-4)S+Z(-3)S+Z(-2)S+Z(-1)S

Fazit: Relative Bezüge stellen Abstände zur Formelzelle dar. Wird eine Formel mit relativen Bezügen kopiert, ändert sich die angesprochene Zelle, da sich die Position der Formelzelle ändert.

Ändern Sie nun die in unserer Ausgangszelle A5 enthaltene Formel in

=A\$1+A\$2+A\$3+A\$4.

Ersetzen Sie die relativen also durch absolute Bezüge und kopieren Sie die Formel anschließend wieder nach B5 (Abbildung 2.20).

=A\$1+A\$2+A\$3+A\$4

	A	B
1	23	76
2	64	34
3	22	56
4	56	9
5	165	165
6		

Abbildung 2.20:
Kopieren absoluter
Formeln

Absolute Bezüge betreffen immer eine ganz bestimmte Zelle und werden beim Kopieren von Formeln nicht angepasst. Entsprechend lautet die kopierte Formel ebenfalls

=A\$1+A\$2+A\$3+A\$4.

und addiert unglücklicherweise die gleichen Zellen in Spalte 1 wie die in A5 enthaltene Formel, was mit relativen Bezügen nicht passiert wäre.

Relative Bezüge sind offenbar weitaus flexibler als absolute, da relative Angaben meist auch nach dem Kopieren einer Formel zutreffen. Daher enthalten mindestens 90% aller in einem Arbeitsblatt verwendeten Formeln relative Bezüge.

*Einsatz absoluter
Bezüge*

Allerdings ist diese Flexibilität manchmal äußerst unerwünscht (Abbildung 2.21).

Abbildung 2.21:

Bezug auf eine bestimmte Zelle

=B4*B1

	A	B	C
1	Mehrwertsteuer:	16%	
2			
3			
4	Netto:	9367	8946
5	MWSt:	1498,72	
6			

B4 und C4 enthalten irgendwelche Nettobeträge. Darunter soll die zugehörige Mehrwertsteuer erscheinen. Der aktuelle Mehrwertsteuersatz von 16% wurde in Zelle B1 eingegeben. B5 enthält die Formel

=B4*B1

Sie besagt: »Multipliziere den in B4 enthaltenen Wert mit dem in B1 enthaltenen Prozentsatz.« Diese Berechnung soll auch in C5 ausgeführt werden.

Kopieren Sie die in B5 enthaltene Formel nach C5, wird die Formel um eine Spalte nach rechts verschoben und die darin enthaltenen relativen Bezüge werden entsprechend angepasst (Abbildung 2.22).

Abbildung 2.22:

Fehler durch Verwendung relativer Bezüge beim Kopieren

=C4*C1

	A	B	C
1	Mehrwertsteuer:	16%	
2			
3			
4	Netto:	9367	8946
5	MWSt:	1498,72	0
6			

Die Formel lautet nun

=C4*C1

Aus B4 wird C4, was ja auch in Ordnung ist, da der betreffende Nettobetrag nun in dieser Zelle enthalten ist; aus B1 wird C1 – was keinesfalls in Ordnung ist, da sich der Mehrwertsteuer-Prozentsatz ja unverändert in B1 befindet!

Das Problem ist gelöst, wenn Sie in der Ausgangszelle B5 den relativen Bezug auf die Zelle mit dem Prozentsatz durch einen absoluten Bezug ersetzen:

=B4*\$B\$1

B4 ist weiterhin ein relativer Bezug, $\$B\$1$ nun jedoch ein absoluter Bezug, der sich auch nach dem Kopieren nach C5 weiterhin auf die gleiche Zelle wie zuvor bezieht. Diesmal enthält B5 nach dem Kopieren die korrekt angepasste Formel $=C4*\$B\$1$ (Abbildung 2.23).

$=C4*\$B\$1$

	A	B	C
1	Mehrwertsteuer:	16%	
2			
3			
4	Netto:	9367	8946
5	MWSt:	1498,72	1431,36
6			

Abbildung 2.23:
Mit absolutem
Bezug

Gemischte Bezüge, bei denen der Spaltenbezug absolut und der Zeilenbezug relativ ist oder umgekehrt, spielen in der Praxis eine Nebenrolle.

*Gemischte
Bezüge*

Bei diesen Bezügen befindet sich vor genau einer der beiden Angaben ein Dollarzeichen. Zum Beispiel ist im Bezug $A\$2$ nur der Zeilenbezug absolut. Wird eine Formel kopiert, die diesen Bezug enthält, bezieht sich auch die Kopie immer auf Zeile 2, allerdings nicht unbedingt auch auf Spalte A, da die Spaltenangabe relativ ist und daher angepasst wird.

Die Taste $\boxed{F4}$ erleichtert die Änderung von Bezugsformaten erheblich. Setzen Sie in der Zelle selbst oder in der Bearbeitungszeile den Cursor rechts neben den zu ändernden Bezug und drücken Sie $\boxed{F4}$, so wird das Bezugsformat umgewandelt. Drücken Sie solange wiederholt $\boxed{F4}$, bis das gewünschte Format erreicht ist.



Sie können sogar einen Teil der Formel markieren, der mehrere Bezüge enthält und mit $\boxed{F4}$ alle Bezüge, die dieser Bereich enthält, gleichzeitig umwandeln.

2.5.5 Funktionen

Formeln können Funktionsaufrufe enthalten. Einer Funktion werden beim Aufruf in Klammern und durch Semikola voneinander getrennte Argumente übergeben; je nach Funktionstyp beispielsweise Texte oder Zahlen, die sie nach einer eingebauten Rechenvorschrift bearbeitet. Als Ergebnis übergibt die Funktion wiederum einen Funktionswert, beispielsweise ebenfalls einen Text oder ein Zahl:

Funktionswert = Funktion(Argument1; Argument2; Argument3...)

Beispielsweise ermittelt die Funktion *MITTELWERT* den Mittelwert mehrerer Zahlen. Die Formel lautet

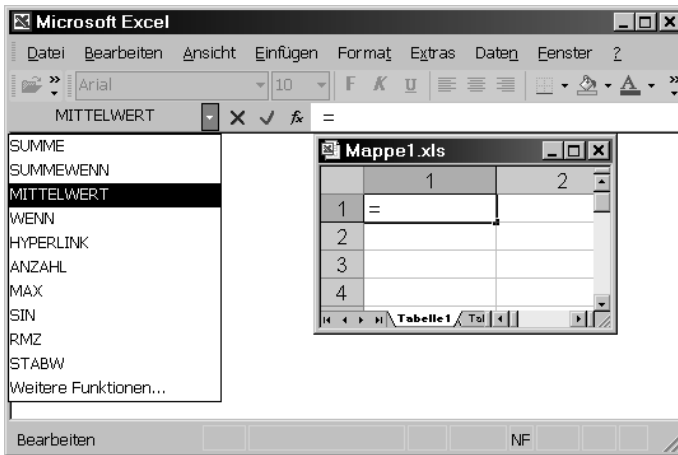
`=MITTELWERT(5;7;9).`

addiert 5, 7 und 9, teilt das Ergebnis durch drei und übergibt den Funktionswert 7. Er wird in der Zelle angezeigt, in der sich diese Formel befindet.



Sie können diese Funktion per Hand eintippen oder den Funktions-Assistenten benutzen: Nach der Eingabe des einleitenden »=« wird in der Bearbeitungsleiste ein geschlossenes Listenfeld eingeblendet, das nach dem Öffnen die am häufigsten benötigten Funktionen enthält (Abbildung 2.24).

Abbildung 2.24:
Funktionslistenfeld



Klicken Sie auf die gewünschte Funktion oder auf den Eintrag *Weitere Funktionen...*, der ein Dialogfeld öffnet, das alle verfügbaren Funktionen enthält, in Kategorien unterteilt (Abbildung 2.25).

Im unteren Listenfeld wählen Sie eine der alphabetisch aufgelisteten Funktionen aus, wobei die momentan selektierte in Stichworten beschrieben wird.

Zur näheren Einschränkung können Sie im oberen Eingabefeld eine Beschreibung der gewünschten Tätigkeit eingeben (»Standardabweichung berechnen«, »Zahl nachschlagen«, »Wert umrechnen« etc.) und auf *Start* klicken. Excel zeigt daraufhin im unteren Feld nur die seiner Meinung nach dazu passenden Funktionen an.

Alternativ dazu öffnen Sie das geschlossene Listenfeld und wählen darin eine der Funktionskategorien wie *Finanzmathematik* aus, worauf darunter nur finanzmathematische Funktionen angezeigt werden.

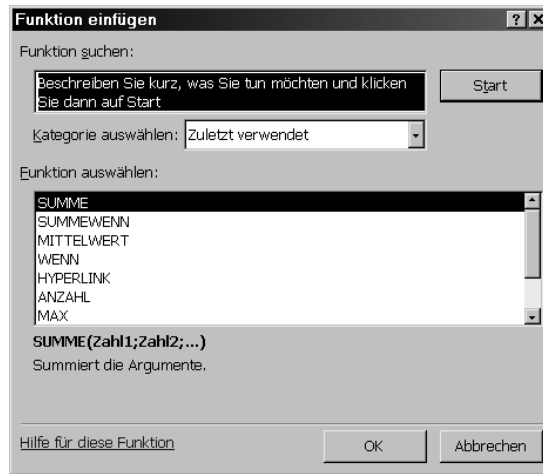


Abbildung 2.25:
Funktionsauswahl

Nach Auswahl der gewünschten Funktion, im Beispiel *MITTELWERT*, leitet OK den nächsten Schritt ein, die Angabe der Funktionsargumente.

Unter *Zahl1* tippen Sie das erste Argument ein, beispielsweise die Zahl 5. Mit  gelangen Sie zu *Zahl2*, wo Sie das zweite Argument eingeben und so weiter.

Nach jeder Argumenteingabe wird im Dialogfeld der aktuelle Funktionswert angezeigt und ein weiteres Eingabefeld erscheint darin (Abbildung 2.26).

Haben Sie die gewünschten Argumente eingegeben, beispielsweise die drei Zahlen 5, 7 und 9, und enthält die Bearbeitungsleiste die gewünschte Formel, wählen Sie OK, um die Funktionseingabe zu beenden. Die erzeugte Formel wird daraufhin in die Bearbeitungsleiste eingefügt.

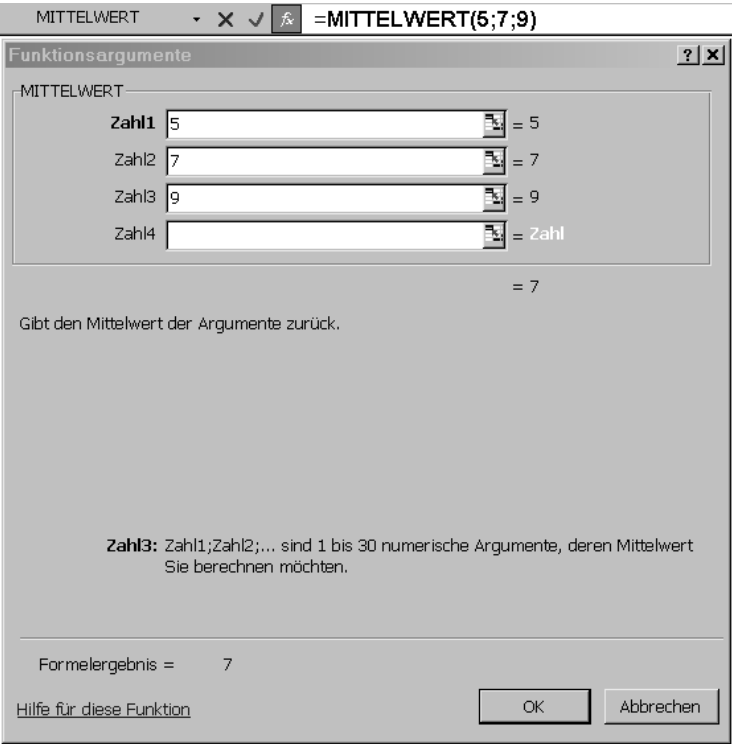
Wollen Sie die Formel nachträglich editieren, klicken Sie einfach erneut auf das Funktions-Assistent-Symbol: der Funktions-Assistent wird wieder aufgerufen und Sie können die in den einzelnen Feldern enthaltenen Werte editieren, zusätzliche Werte eingeben, oder im Funktionslistenfeld eine andere Funktion auswählen.

Sie können als Argumente nicht nur Zahlen, sondern auch Zellbezüge wie A1 oder B5 eingeben, beispielsweise, um mit der Funktion MITTELWERT und der Formel =MITTELWERT(A1;B5) den Mittelwert der beiden in A1 und B5 enthaltenen Zahlen zu ermitteln.



Zur Eingabe müssen Sie den Bezug noch nicht einmal eintippen; es genügt, per Anklicken auf die betreffende Zelle zu zeigen. Excel fügt daraufhin den betreffenden Bezug in die Formel ein.

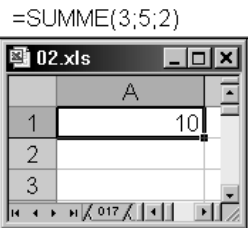
Abbildung 2.26:
Argumentübergabe



Ist das Dialogfeld dabei im Weg, klicken Sie auf das Symbol am rechten Rand des Eingabefelds, um es zu verkleinern. Zeigen Sie danach auf die Zelle und klicken Sie im nun verkleinerten Dialogfeld das darin am rechten Rand sichtbare Symbol an, das die ursprüngliche Dialogfeldgröße wiederherstellt.

SUMME-Funktion Eine äußerst nützliche Funktion ist *SUMME*. Sie addiert bis zu dreißig Zahlen, die ihr übergeben werden (Abbildung 2.27).

Abbildung 2.27:
Die SUMME-
Funktion



A1 enthält die Formel *=SUMME(3;5;2)*. *SUMME* addiert die drei Zahlen und übergibt den Funktionswert 10, der in der Zelle angezeigt wird. Außer Zahlenkonstanten können *SUMME* auch Zellbezüge übergeben werden, um die Inhalte der betreffenden Zellen zu addieren (Abbildung 2.28).

=SUMME(A1;A2;A3;A4)

The screenshot shows an Excel window titled '02.xls'. The active cell is A5, which contains the formula '=SUMME(A1;A2;A3;A4)' and the result '165'. The spreadsheet has columns A and B, and rows 1 through 6. The data in column A is: 23 (A1), 64 (A2), 22 (A3), 56 (A4), and 165 (A5). The status bar at the bottom shows '023 / 024'.

	A
1	23
2	64
3	22
4	56
5	165
6	

Abbildung 2.28:
Zellbezüge als
Argumente

Die in A5 enthaltene Formel =SUMME(A1;A2;A3;A4) addiert die in den Zellen A1, A2, A3 und A4 enthaltenen Zahlen.

2.5.6 Bereichsangaben

Außer einzelnen Zahlen oder Zellen können Funktionen häufig ganze Zellbereiche als Argumente übergeben werden. Ein solcher Bereich kann durch Angabe zweier Eckpunkte eindeutig gekennzeichnet werden:

Ecke1:Ecke2

A1:A4 meint beispielsweise den von A1 bis A4 reichenden Bereich, der aus vier Zellen besteht: A1, A2, A3 und A4. Die Angabe A1:B4 kennzeichnet entsprechend einen Bereich, der sich über vier Zeilen und zwei Spalten erstreckt und acht Zellen enthält, A1, A2, A3, A4, B1, B2, B3 und B4.

Auch der SUMME-Funktion können derartige Bereiche als Argumente übergeben werden (Abbildung 2.29).

=SUMME(A1:B4)

The screenshot shows an Excel window titled '02.xls'. The active cell is A5, which contains the formula '=SUMME(A1:B4)' and the result '340'. The spreadsheet has columns A and B, and rows 1 through 6. The data in column A is: 23 (A1), 64 (A2), 22 (A3), 56 (A4), and 340 (A5). The data in column B is: 76 (B1), 34 (B2), 56 (B3), 9 (B4). The status bar at the bottom shows '023 / 024'.

	A	B
1	23	76
2	64	34
3	22	56
4	56	9
5	340	
6		

Abbildung 2.29:
Zellbereich als
Argument

Die Formel =SUMME(A1:B4) ruft die SUMME-Funktion mit dem Argument A1:B4 auf, die daraufhin die in diesem Bereich enthaltenen acht Zahlen addiert. Diese Bereichsangabe erspart Ihnen die umständliche Formel

=SUMME(A1;A2;A3;A4;B1;B2;B3;B4).

Im Zeigemodus

Nicht nur einzelne Zellbezüge, auch Bereichsbezüge können Sie im Zeigemodus angeben. Dazu tippen Sie »=SUMME(« ein und zeigen auf den betreffenden Bereich, indem Sie auf eine Ecke des Bereichs klicken, beispielsweise auf A1, die Maustaste weiter gedrückt halten und die Maus zum entgegengesetzten Bereichsende B4 ziehen.

Excel fügt daraufhin den gewünschten Bereichsbezug A1:B4 ein und Sie vervollständigen die Formel =SUMME(A1:B4 durch Eingabe der schließenden Klammer »)«.

TIPP*Mehrere Bereiche*

Mit der Tastatur bewegen Sie den Cursor zu A1, worauf der Zellbezug eingefügt wird, tippen dahinter einen Doppelpunkt »:« ein und bewegen den Cursor zu B4, worauf auch dieser Zellbezug eingefügt wird.

Auf diese Weise können Sie beliebig viele Bereichsargumente übergeben. Befinden sich beispielsweise auch im Bereich D10:D20 zu addierende Werte, schließen Sie die Formel =SUMME(A1:B4 noch nicht mit »)« ab, sondern tippen zunächst das Zeichen »:« ein, das mehrere Funktionsargumente voneinander trennt, und zeigen danach auf diesen zweiten Bereich, damit Excel hinter dem »:« diesen Bezug einfügt (drücken Sie beim Zeigen auf den folgenden Bereich die Strg-Taste, fügt Excel »:« automatisch ein). Die Formel würde nun lauten:

=SUMME(A1:B4;D10:D20)

*Spalten, Zeilen,
Tabelle*

Sie können sogar auf komplette Spalten oder Zeilen zeigen, auf eine Zeile, beispielsweise die »10«, indem Sie die Zeilennummer anklicken. Excel fügt daraufhin den entsprechenden Zeilenbezug in der Form »10:10« ein und die SUMME-Funktion addiert alle in der betreffenden Zeile enthaltenen Zahlen.

Einen Spaltenbezug fügen Sie durch Anklicken eines Spaltenbuchstabens wie »C« ein: Excel fügt den Bezug »C:C« ein, der die gesamte Spalte C meint.

INFO

Während der Eingabe können Sie Fehler korrigieren, indem Sie an dem Rahmen ziehen, der jeden Bezug während der Eingabe hervorhebt, und ihn zur korrekten Zelle verschieben. Durch Ziehen an den Eckpunkten können Sie den angegebenen Bereich sogar vergrößern oder verkleinern.

INFO

Das kleine Fensterchen, das bei der Eingabe der Funktionsargumente hilft, können Sie per Anklicken und Ziehen verschieben, sollte es stören. Es kann allerdings ganz hilfreich sein. Klicken Sie auf den darin enthaltenen Funktionsnamen, dann wird die Hilfe zu dieser Funktion aktiviert.

Darüber hinaus zeigt es an, welche Argumente Sie bisher eingaben und welche weiteren Argumente von der Funktion erwartet werden.

Zusätzlich können Sie auf eines der bereits eingegebenen Argumente klicken, um es zu ändern. Klicken Sie beispielsweise auf das zweite Argument, so wird der zweite Bereichsbezug in Ihrer Formel entsprechend markiert und Sie können den betreffenden Bezug nun ändern.

Vertun Sie sich bei einer Bereichsangabe und lassen vermutlich miteinzubeziehende Zellen aus, erkennt Excel das häufig und bietet Ihnen an, das zu korrigieren. Angenommen, Sie wollen nur die erste Zeile summieren, markierten aber versehentlich den Bereich A2:A3 statt A1:A4 (Abbildung 2.30).

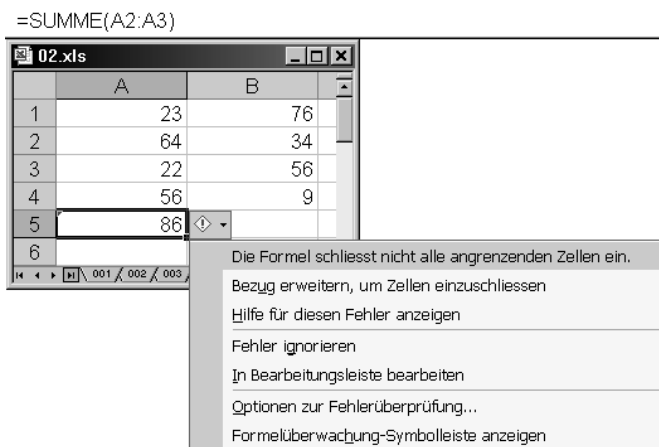


Abbildung 2.30:
Smart Tag

Excel vermutet, dass Sie möglicherweise auch die angrenzenden Zellen A1 und A4 einbeziehen wollen und zeigt in der Formelzelle A5 einen kleinen Indikator an. Folgen Sie der Aufforderung und klicken Sie auf die Zelle, erscheint ein Smart Tag, das Ihnen nach dem Klicken auf den Listenpfeil unter anderem anbietet, die Formel zu korrigieren. Wählen Sie den entsprechenden Befehl, ändert Excel Ihre Formel `=SUMME(A2:A3)` in `=SUMME(A1:A4)`, um die oben und unten angrenzenden Zellen einzubeziehen.

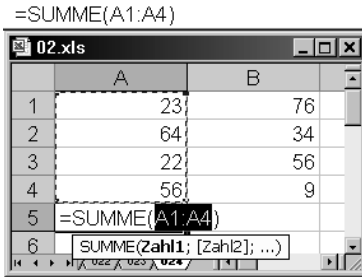
Befindet sich der interessierende Bereich rechts/links der Formelzelle, erhalten Sie diese *AutoKorrektur*-Option ebenfalls; jedoch nicht bei Bereichen, die gegenüber der Formelzelle horizontal *und* vertikal verschoben sind.

2.5.7 Summen, Mittelwerte etc.

Um den Inhalt eines oder mehrerer Bereiche zu addieren, sollten Sie einfach auf das Summe-Symbol der Standardleiste klicken. Excel fügt daraufhin das Gleichheitszeichen `=` und dahinter den vollständigen Funktionsaufruf der `SUMME`-Funktion ein und schlägt dabei als Funktionsargument die an die Formelzelle angrenzenden Zellen vor, beispielsweise den Bereich A1:A4 (Abbildung 2.31).



Abbildung 2.31:
Bereichsvorschlag



Diesen Vorschlag können Sie korrigieren, indem Sie wie erläutert mit Maus oder Tastatur auf den tatsächlich zu addierenden Bereich A1:B4 zeigen. Excel korrigiert die Formel entsprechend in `=SUMME(A1:B4)` und Sie müssen nur noch  drücken.

Auch bei Verwendung des Summe-Symbols haben Sie alle zuvor erläuterten Möglichkeiten zur Angabe weiterer Bereiche per Zeigen oder Eintippen und zum Korrigieren dieser Angaben.



Das Summe-Symbol ist ein Listenfeld. Öffnen Sie die Liste, können Sie statt SUMME eine andere Funktion anwenden, beispielsweise mit MITTELWERT den Durchschnitt der Werte im betreffenden Bereich ermitteln oder mit MAX den größten darin enthaltenen Wert.

2.5.8 Formeln testen

Excel kann Formeln auch *teilweise* berechnen und es Ihnen dadurch ermöglichen, eine Formel Schritt für Schritt zu untersuchen, um darin enthaltene Fehler zu entdecken. Nehmen wir an, Sie wollen den Ausdruck

$$B2 + \text{SUMME}(C1:C10) + 3 \cdot 4,25$$

mit 2 multiplizieren und benötigen daher die Formel

$$=2 * (B2 + \text{SUMME}(C1:C10) + 3 \cdot 4,25) .$$

Sie vergessen jedoch versehentlich die Klammer und geben stattdessen ein:

$$=2 * B2 + \text{SUMME}(C1:C10) + 3 \cdot 4,25$$

Sie werden zwar schnell entdecken, dass das Ergebnis nicht Ihren Vorstellungen entspricht, erkennen jedoch möglicherweise nicht auf Anhieb, wo der Fehler liegt.

Dann selektieren Sie in der Bearbeitungsleiste einen Teil der Formel und drücken . Excel berechnet den markierten Ausdruck und zeigt das Ergebnis an.

Selektieren Sie in der fehlerhaften Formel zum Beispiel den Ausdruck $3*4,25$ und drücken Sie **F9**, zeigt Excel an:

`=2*B2+SUMME(C1:C10)+12,75`

Der Ausdruck $3*4,25$ wird durch seinen Wert 12,75 ersetzt. Wenn Sie **Esc** drücken, wird die Teilauswertung rückgängig gemacht und die ursprüngliche Formel angezeigt. Anschließend können Sie auf die gleiche Weise beliebige andere Teile der Formel berechnen und überprüfen, bis Sie den Fehler gefunden haben.

Drücken Sie **↵** statt **Esc**, wird der angezeigte Wert übernommen und ersetzt den selektierten Ausdruck dauerhaft!

Das Ganze funktioniert nicht, wenn der Funktions-Assistent aktiv ist!



Zum Testen von Formeln können Sie auch das Formelauswertungsfenster einsetzen, das beim Nachvollziehen der schrittweisen Auswertung einer Formel recht hilfreich ist (siehe Kapitel 38.3, »Fehler in Tabellen aufdecken«, Abschnitt »Formelauswertung«).



2.5.9 AutoBerechnung

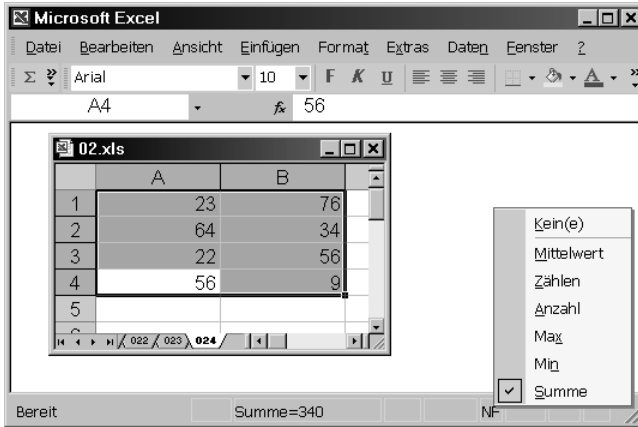
Um Formeln zu testen oder zwischendurch schnell mal Berechnungen durchzuführen, können Sie auch bei Excel die AutoBerechnung benutzen.

Angenommen, in einer Tabelle befinden sich einige zu addierende Zahlen: statt zur Addition die SUMME-Funktion zu verwenden, markieren Sie einfach den betreffenden Bereich (Abbildung 2.32).

In der Statusleiste wird daraufhin die Summe dieser Werte in der Form *Summe=xxx* angezeigt, im Beispiel 340.

Klicken Sie mit der rechten Maustaste auf die Statusleiste, so können Sie übrigens eine andere AutoBerechnungs-Funktion auswählen oder die AutoBerechnung mit *Kein(e)* ganz ausschalten (siehe Bild 2.32). Wählen Sie beispielsweise *Mittelwert*, wird in der Statusleiste ab jetzt der Mittelwert der selektierten Zellen angezeigt.

Abbildung 2.32:
AutoBerechnung



2.6 Kommentare



Zu jeder Zelle kann ein praktisch beliebig umfangreicher Kommentar gehören. Sie erstellen ihn mit EINFÜGEN|KOMMENTAR bzw. dem zugehörigen Symbol der Detektiv- oder der Überarbeiten-Symbolleiste (Abbildung 2.33).

Abbildung 2.33:
Das Kommentar-
fenster



Ihr Benutzername wird im Kommentarfenster automatisch vorgegeben; den restlichen Text müssen Sie selbst eingeben, z.B. »Im Jan vorgenommene Autoreparatur«.

Excel nimmt den Zeilenumbruch während der Eingabe selbstständig vor. Alternativ dazu können Sie mit jederzeit manuell einen Zeilenumbruch veranlassen. Zusätzlich können Sie durch das Einfügen von Bindestrichen manuell trennen, wenn Ihnen der Zeilenumbruch nicht passt.

Sie beenden die Eingabe, indem Sie auf irgendeinen Punkt außerhalb des Kommentarfensers klicken.

Um den Kommentar nachträglich zu verändern, setzen Sie den Cursor auf die zugehörige Zelle und wählen den Befehl EINFÜGEN|KOMMENTAR BEARBEITEN, der EINFÜGEN|KOMMENTAR nun ersetzt.

Um einen Kommentar zu löschen, setzen Sie den Cursor ebenfalls auf die zugehörige Zelle und wählen BEARBEITEN|LÖSCHEN|KOMMENTARE.

Noch einfacher ist die Verwendung der Befehle KOMMENTAR BEARBEITEN bzw. KOMMENTAR LÖSCHEN, die sich im Zellkontextmenü befinden (mit der rechten Maustaste auf die betreffende Zelle klicken).

Gehört zu einer Zelle ein Kommentar, ist am oberen rechten Zellrand ein kleines rotes Dreieck sichtbar. Befindet sich der Cursor auf einer solchen Zelle, wird der Kommentar automatisch sichtbar.

Um ihn dauerhaft einzublenden, wählen Sie im Kontextmenü KOMMENTAR ANZEIGEN. KOMMENTAR AUSBLENDEN bewirkt genau umgekehrt, dass der Kommentar nur angezeigt wird, solange sich der Cursor auf der Zelle befindet.

Mit ANSICHT|KOMMENTARE können Sie alle Kommentare dauerhaft ein- bzw. durch erneute Wahl dieses Befehls wieder ausblenden.

Die automatisch eingeblendete Überarbeiten-Symbolleiste enthält übrigens mehrere Symbole, die ebenfalls all diese Möglichkeiten bieten und Ihnen darüber hinaus erlauben, alle in der Tabelle vorhandenen Kommentare der Reihe nach zu durchblättern.

In Kapitel 20, »Grafikobjekte«, erläutere ich den Umgang mit Grafikobjekten wie dem Kommentarfenster und erkläre, wie Sie derartige Objekte vergrößern, verkleinern, verschieben oder den eingegebenen Text formatieren können.



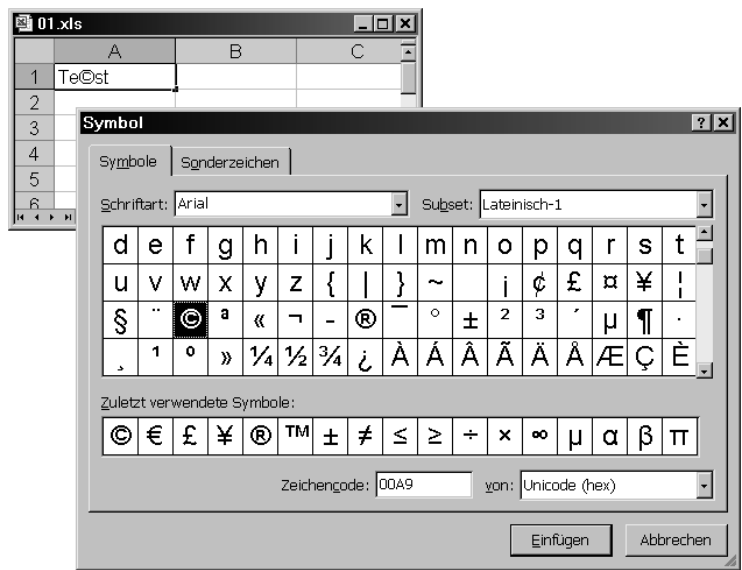
2.7 Symbole

Außer Texten und Zahlen können Sie auch Spezial-Symbole in eine Tabelle eingeben, zum Beispiel Währungs-Symbole, mathematische Symbole und so weiter. Dazu wählen Sie EINFÜGEN|SYMBOL..., im darauf erscheinenden Dialogfeld das gewünschte Symbol, und klicken auf *Einfügen* (Abbildung 2.34).

Das Symbol entweder als einziges Zeichen in eine leere Zelle eingefügt oder – bei nicht-leeren Zellen – an's Ende des darin enthaltenen Textes gesetzt bzw. – wenn der Bearbeitungsmodus aktiviert ist – genau dort in diesem Text eingefügt, wo sich gerade der Cursor befindet.

Das Dialogfeld bleibt anschließend geöffnet, sodass Sie ein weiteres Symbol auswählen und einfügen können.

Abbildung 2.34:
Symbol einfügen



Mit den beiden Listenfeldern können Sie die zu verwendende Schriftart bestimmen und unter unterschiedlichsten Symbolkategorien die gewünschte auswählen. Zusätzlich enthält das Register *Sonderzeichen* noch eine Menge weiterer spezieller Symbole.

3 Dateien

Inzwischen haben Sie sicherlich die eine oder andere Tabelle erzeugt – die Sie natürlich auch speichern wollen! Das folgende Kapitel behandelt alles rund ums Speichern von Arbeitsmappen.

Sie lernen, wie Sie Laufwerk und Verzeichnis auswählen. Ich zeige Ihnen, wie Sie mit Kennwörtern Ihre Arbeitsmappe vor unbefugten Zugriffen schützen können; wahlweise, indem Sie den Zugriff ohne Eingabe Ihres Kennworts komplett sperren oder aber nur einen lesenden Zugriff auf die Mappe gestatten, sodass ein anderer sich zwar Ihren Inhalt anschauen, die Originaldatei aber nicht überschreiben kann.

Außer einzelnen Mappen können Sie auch ganze Arbeitsbereiche speichern, das heißt, alle geöffneten Mappen gleichzeitig und zusätzlich auch die Größe und die Positionen der Fenster, in denen sich diese Mappen momentan befinden. Beim späteren Öffnen wird der Zustand aller Fenster wiederhergestellt.

Das Wiederfinden bestimmter Mappen vereinfacht der Datei-Manager. Er ermöglicht Ihnen, nur Mappen zu selektieren, die bestimmte Werte oder Texte enthalten, vor oder nach einem bestimmten Datum erstellt wurden und so weiter.

Nach der Festlegung dieser Kriterien starten Sie den Suchlauf und erhalten eine Liste aller Mappen, die die betreffenden Bedingungen erfüllen, aus denen Sie anschließend die gewünschte Mappe auswählen können.

3.1 Speichern

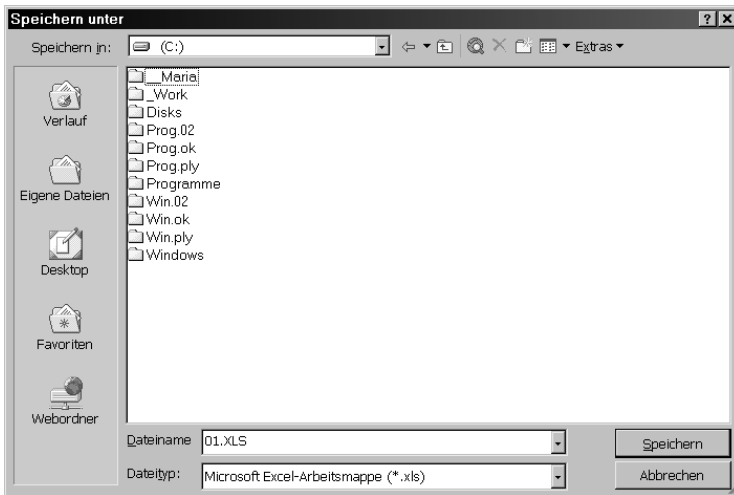
Das abgebildete Symbol der Standardleiste entspricht dem Befehl DATEI|SPEICHERN (Kurtaste **Strg**+s oder **Alt**+**F12**), den Sie zum Speichern Ihrer Tabellen verwenden.



Gespeichert wird immer die gesamte Arbeitsmappe mit allen darin enthaltenen Tabellen.

Wird eine Arbeitsmappe zum ersten Mal von Ihnen gespeichert, verlangt Excel im *Speichern unter*-Dialogfeld die Eingabe eines Dateinamens (Abbildung 3.1).

Abbildung 3.1:
Datei speichern



Dateinamen Im Feld *Dateiname* erscheint als Vorschlag der in der Titelleiste der Mappe vorgegebene Name, zum Beispiel »Mappe1«, mit dem Zusatz .XLS.

Sie können diesen Dateinamen akzeptieren oder einen nahezu beliebigen anderen Namen eingeben und die Datei danach mit  bzw. Klicken auf *Speichern* speichern.

Dateinamen können aus bis zu 255 Zeichen bestehen, wobei sogar Leerzeichen und andere (allerdings nicht alle) Sonderzeichen verwendet werden können. »Test 5.xls« ist daher ebenso ein gültiger Name wie »Tabelle von Maier vom 15.4.09«.

Erweiterung Vergeben Sie keine Erweiterung, hängt Excel an den von Ihnen vorgegebenen Dateinamen automatisch den Zusatz .XLS an, damit Sie auf einen Blick an diesem Zusatz erkennen können, dass es sich bei der betreffenden Datei um eine Excel-Datei handelt. Geben Sie »TEST« ein, wird Ihre Datei entsprechend unter dem Namen TEST.XLS gespeichert.

Wollen Sie, dass Ihre Datei ohne Erweiterung gespeichert wird, muss Ihre Eingabe mit einem Punkt enden. Mit »HALLO.« wird die Datei entsprechend unter HALLO gespeichert, ohne jede Endung.

Laufwerk und Verzeichnis

Wo eine Datei gespeichert wird, sehen Sie im oberen normalerweise geschlossenen Listenfeld *Speichern in*: »(C:)« bedeutet beispielsweise, dass die Datei im Stammverzeichnis von Laufwerk C: gespeichert wird, der Festplatte Ihres Rechners.

Steht in diesem Listenfeld stattdessen »Windows«, wird die Datei entsprechend im Ordner (=Verzeichnis) WINDOWS der Festplatte gespeichert.

Öffnen Sie dieses Listenfeld, so können Sie statt des vorgegebenen Laufwerks ein beliebiges anderes Laufwerk Ihres Rechners oder eines anderen mit diesem über ein Netzwerk verbundenen Rechners auswählen (Abbildung 3.2).

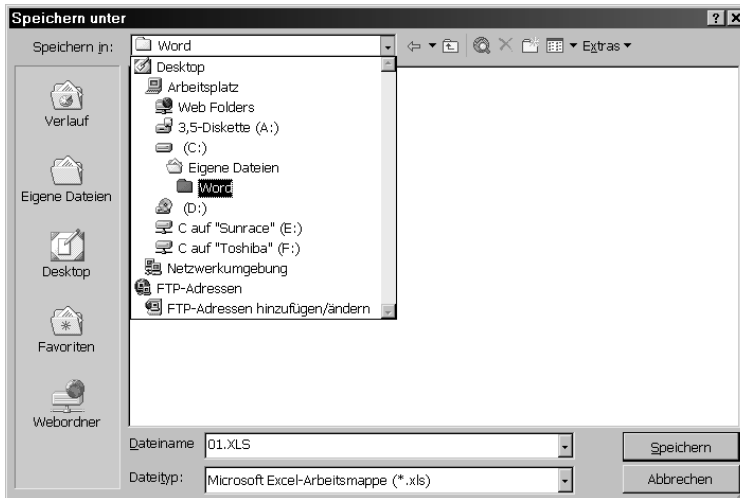


Abbildung 3.2:
Laufwerk wählen

Im großen Listenfeld wird immer der Inhalt des aktuellen Ordners angezeigt (siehe Bild 3.1). Oft enthält dieser Ordner nicht nur Dateien, sondern auch weitere Ordner; praktisch Unterordner oder Unterverzeichnisse, die selbst wiederum Dateien enthalten.

In jedem Fall genügt ein Doppelklick auf das betreffende Symbol, um in den betreffenden Ordner zu wechseln, beispielsweise vom Ordner C: (dem Stammverzeichnis der Festplatte C:) zum darin enthaltenen Ordner EIGENE DATEIEN (Pfad C:\EIGENE DATEIEN), der außer Dateien gegebenenfalls selbst wieder Ordner enthält, in die Sie sich wiederum per Doppelklick begeben können.

Umgekehrt kommen Sie durch Anklicken des abgebildeten Symbols wieder zur nächsthöheren Ordner-Ebene, beispielsweise vom Ordner C:\EIGENE DATEIEN wieder in den Ordner C:.



Sie können das Dateiauswahl-Dialogfeld wie ein Fenster durch Ziehen an einem der Ränder oder einer Ecke vergrößern bzw. – in Grenzen – verkleinern, worauf im großen Listenfeld entsprechend mehr bzw. weniger Einträge dargestellt werden.



Der linke Teil des Dialogfelds bietet eine alternative Technik zum Festlegen des Speicherorts. Jedes Symbol steht für einen bestimmten häufig benutzten Speicherort, der beim Anklicken des Symbols geöffnet wird:

- ➔ *Verlauf*: Dieser Ordner enthält die zuletzt von Ihnen benutzten Dateien.
- ➔ *Eigene Dateien*: In diesem Ordner sollten Sie nach Meinung von Microsoft Ihre Dateien vorzugsweise speichern.
- ➔ *Desktop* stellt den Inhalt des Desktops dar. Speichern Sie eine Datei in diesem Ordner, so erscheint Sie direkt auf dem Windows-Desktop.
- ➔ *Favoriten* ist ein Spezialordner, der Verknüpfungen zu beliebigen Dateien und Ordnern enthält. Wie ich noch zeigen werde, können Sie in diesem Ordner Verknüpfungen zu Ihren wichtigsten Dateien und Ordnern ablegen. Öffnen Sie diesen Ordner später per Anklicken von *Favoriten*, so sehen Sie darin auf einen Blick Ihre wichtigsten Verknüpfungen und können das betreffende Objekt sofort durch Anklicken öffnen.



Webordner ist ebenfalls ein Spezialordner, der Verknüpfungen zu allen von Ihnen definierten Webserver-Verzeichnissen enthält (die Möglichkeiten von Excel, Dateien im Internet zu speichern, erläutere ich in Kapitel 26, »Excel-Tabellen im Internet/Intranet«).

Alternativ zu diesem interaktiven Festlegen von Laufwerk und Ordner können Sie beides auch direkt zusammen mit dem Dateinamen im Feld *Dateiname* eintippen. Die Eingabe »C:\Eigene Dateien\test« würde die aktuelle Tabelle beispielsweise unter dem Namen TEST.XLS im Ordner EIGENE DATEIEN von Laufwerk C: speichern.



Die Eingabe des ersten Zeichens eines Ordner- oder Dateinamens genügt, um ihn zu selektieren. Tippen Sie »e« ein, bewegt Excel die Markierung zum ersten Verzeichnisnamen, der mit »e« beginnt, beispielsweise zu EIGENE DATEIEN.



Selektieren Sie eine Datei oder einen Ordner und klicken Sie (kurz warten, um einen Doppelklick zu vermeiden) erneut darauf, so können Sie den Datei-/Ordnernamen anschließend ändern.



Klicken Sie mit der rechten Maustaste auf eine Datei oder einen Ordner, erscheint ein Kontextmenü, das unter anderem Befehle zum Löschen und zum Kopieren enthält.

Im Listenfeld *Dateityp* haben Sie die Wahl unter verschiedenen Dateiformaten. Vorselektiert ist immer »Microsoft Excel-Arbeitsmappe«, das Excel-eigene Format, mit dem das Programm am besten und schnellsten umgehen kann. Die anderen Formate sind zum Datenaustausch gedacht und werden von mir im Kapitel 19.2, »Dateiformate«, erläutert.

Dateitypen

In Kapitel 8.8, »Arbeitsmappen als Vorlagen verwenden und automatisch öffnen«, erläutere ich, wie Sie mithilfe des Startordners XLSTART dafür sorgen können, dass ständig benötigte Arbeitsmappen beim Start von Excel automatisch geöffnet werden. Vor allem aber sehen Sie hier, wie Sie Arbeitsmappen in den verschiedenen Vorlagenverzeichnissen als Mustervorlagen speichern und anschließend als Muster zum Erzeugen identisch aufgebauter neuer Mappen verwenden können!



Mehrere Symbole am oberen Rand des Dialogfelds erleichtern den Umgang mit Dateien und Ordnern:

Symbole

Wechselt in den Ordner, in dem Sie sich zuletzt befanden. Klicken Sie auf den Listenpfeil, so öffnet sich eine Liste mit allen Ordnern, in denen Sie sich zuletzt befanden.



Wechselt in den Ordner oberhalb des aktuellen Ordners



Aktiviert Ihren Webbrowser und ermöglicht Ihnen die Auswahl eines Speicherorts im Internet



Löscht die selektierte Datei



Erzeugt im aktuellen Ordner einen neuen Ordner und fragt Sie nach dem Namen, den der neue Ordner erhalten soll



Öffnet eine Liste mit verschiedenen Befehlen. Die wichtigsten:



- ➔ GROSSE SYMBOLE zeigt Dateien/Ordner als große Symbole an.
- ➔ KLEINE SYMBOLE zeigt Dateien/Ordner als kleine Symbole an.
- ➔ LISTE zeigt Dateien/Ordner entsprechend Bild 3.1 in Listenform an.
- ➔ DETAILS: In mehreren Spalten werden außer den Namen von Dateien/Ordnern zusätzliche Informationen wie die Dateigröße, die Dateiart und das Datum der letzten Veränderung angezeigt.
- ➔ EIGENSCHAFTEN zeigt die wichtigsten Dateieigenschaften an (Name des Autors, Dateigröße, Erstellungsdatum etc.).
- ➔ VORSCHAU zeigt einen Ausschnitt des Inhalts der betreffenden Excel-Datei an.

Ein Menü mit verschiedenen weiteren Befehlen. Die nützlichsten:

- ➔ ZU FAVORITEN HINZUFÜGEN fügt die selektierte Datei bzw. den Ordner den Favoriten hinzu, genauer: es legt in diesem Ordner eine Verknüpfung zum betreffenden Objekt ab. Das ermöglicht den schnellen Zugriff auf die betreffenden Objekte, da wie gesagt ein Klick auf *Favoriten* im linken Dialogfeldteil genügt, um diesen Ordner zu öffnen und die Verknüpfungen anzuzeigen.
- ➔ NETZLAUFWERK VERBINDEN... ermöglicht Ihnen eine dauerhafte Verbindung zu einem Laufwerk eines anderen Rechners herzustellen und diesem Laufwerk einen Laufwerksbuchstaben wie E: zu geben, unter dem Sie ab jetzt darauf zugreifen können.



WEBOPTIONEN... sind Optionen, die das Verhalten von Excel beim Speichern bzw. Öffnen von Dateien im Internet festlegen (siehe Kapitel 30.1, »Weboptionen«).

ALLGEMEINE OPTIONEN...: (siehe Kapitel 3.1.1, »Speicheroptionen«)

BILD KOMPRIMIEREN... steuert die Komprimierung in Tabellen eingebetteter Bilder, die vor allem bei der Veröffentlichung von Excel-Dokumenten als Webseiten im Internet sinnvoll ist (siehe Kapitel 30.2, »Bilder komprimieren«).

3.1.1 Speicheroptionen

Der Befehl ALLGEMEINE OPTIONEN... des Symbols *Extras* öffnet das folgende Dialogfeld (Abbildung 3.3).

Sicherungsdatei

Aktivieren Sie *Sicherungsdatei erstellen*, so prüft Excel beim Speichern, ob eine Datei dieses Namens bereits existiert; beispielsweise eine Datei TEST.XLS, wenn Sie diesen Namen für Ihre Arbeitsmappe wählen.

Falls ja, haben Sie die Mappe zuvor schon einmal gespeichert. Die Erweiterung dieser Datei wird in .BAK umbenannt, in unserem Fall also in TEST.BAK.

Erst danach wird die Tabelle unter dem Namen TEST.XLS gespeichert – ohne die zuletzt gespeicherte Version zu überschreiben, die ja nun einen anderen Namen besitzt, nämlich TEST.BAK statt TEST.XLS. Dieser Vorgang wiederholt sich bei jedem Speichervorgang. Daher gilt:

.XLS = Aktuelle Version der Datei
 .BAK = Zuletzt gespeicherte Version

Löschen Sie Ihre Datei irgendwann und stellen am nächsten Tag fest, dass Sie sie eigentlich noch gebraucht hätten, dann öffnen Sie einfach die immer noch vorhandene zuletzt gespeicherte Version, also TEST.BAK.

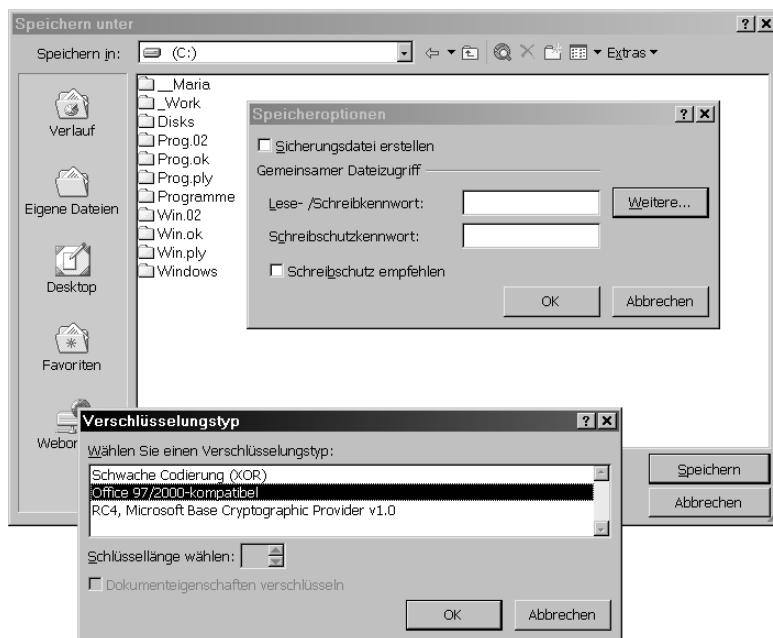


Abbildung 3.3:
Speicheroptionen

Im Feld *Lese-/Schreibkennwort* können Sie einen bis zu 15 Zeichen langen Namen vergeben, der erfragt werden soll, wenn diese Datei irgendwann später geladen wird.

Kennwortschutz

Das Eintippen erfolgt blind – Sie sehen nicht, was Sie tippen, da Excel für jedes eingegebene Zeichen nur ein Sternchen anzeigt. Nach der Eingabe und Aktivierung von *OK* müssen Sie das Kennwort sicherheitshalber ein zweites Mal eingeben.

Angenommen, Sie geben ein: »Testwort«. Und Ihr neugieriger Kollege am benachbarten Arbeitsplatz versucht am nächsten Tag, Ihre Datei zu öffnen. Dann wird er leider Pech haben, da ein Dialogfeld erscheint, in dem ihn Excel auffordert, das Kennwort zu nennen (Abbildung 3.4).



Abbildung 3.4:
Öffnen einer kenn-
wortgeschützten
Datei

Ist die Eingabe falsch, wird die Datei nicht geöffnet. Stattdessen erscheint der Hinweis »Ungültiges Kennwort«.



Excel unterscheidet bei Kennwörtern zwischen Groß- und Kleinschreibung. Die Eingabe »testwort« wäre daher falsch, wenn »Testwort« als Kennwort benutzt wurde!

Einen Kennwortschutz können Sie jederzeit wieder aufheben, indem Sie die Datei mit **SPEICHERN UNTER...** erneut speichern, das Kennwort jedoch wieder entfernen, indem Sie die einzelnen Sternchen löschen.

Verschlüsselung

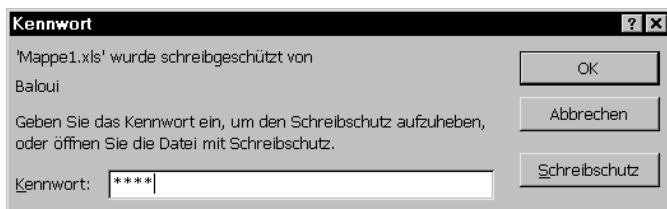
Klicken Sie auf die Schaltfläche *Weitere...*, so erscheint ein Dialogfeld, in dem Sie unter mehreren Verschlüsselungsverfahren wählen und beispielsweise im harmlosesten Fall die schlichte XOR-Verschlüsselung (bitweise »entweder oder«-Verknüpfung) selektieren können.

Die zu speichernde Datei wird daraufhin nicht im Klartext gespeichert, sondern mit dem betreffenden Verfahren verschlüsselt, was erhöhte Sicherheit bringt – allerdings nur, wenn diese Option zusammen mit dem Kennwortschutz verwendet wird, da sonst jeder andere Excel-Anwender die verschlüsselte Datei ebenfalls problemlos öffnen kann!

Schreibschutz

Das Schreibschutz-Kennwort dient zum Schutz einer Datei vor versehentlichem Überschreiben, sowohl durch Ihre Kollegen als auch durch Sie selbst. Vergeben Sie ein solches Kennwort, erscheint bei jedem Versuch, die Datei zu öffnen, folgendes Dialogfeld (Abbildung 3.5).

Abbildung 3.5:
Schreib- und
kennwortgeschützte
Datei



Sie können das Kennwort eingeben und **OK** wählen, um den Schreibschutz aufzuheben.

Oder Sie verzichten auf die Kennworteingabe und aktivieren *Schreibschutz*: Dann erscheint nach dem Öffnen in der Titelleiste hinter dem Dateinamen die Bemerkung »[Schreibgeschützt]«, um Sie darauf hinzuweisen, dass Sie diese Datei nicht verändern können.

Wählen Sie **DATEI|SPEICHERN**, wird nicht wie sonst die alte Datei – zum Beispiel **TEST.XLS** – einfach überschrieben, sondern Sie werden nach einem neuen Namen gefragt.

Geben Sie dennoch den alten Dateinamen **TEST.XLS** ein, weist Excel darauf hin, dass diese Datei nur gelesen werden kann, Änderungen an **TEST.XLS**

jedoch unzulässig sind. Das Speichern wird nur erlaubt, wenn Sie einen neuen Dateinamen wählen und die alte Tabelle somit nicht überschrieben wird.

Aktivieren Sie das Kontrollkästchen *Schreibschutz empfehlen*, erhalten Sie beim späteren Öffnen der Datei eine Warnung, dass sie im Schreibschutz-Modus geöffnet werden sollte – außer, Sie haben diese Option beim Öffnen sowieso aktiviert.

*Schreibschutz-
Empfehlung*

Es handelt sich gewissermaßen um einen Hinweis, den Sie sich selbst geben, um sich daran zu erinnern, dass diese Datei wichtige Daten enthält und vor dem Überschreiben geschützt werden sollte.

Excel gibt Ihnen diesen Hinweis und fordert Sie auf, zu wählen, ob die Datei im Normal- oder im Schreibschutz-Modus geöffnet werden soll (Abbildung 3.6).

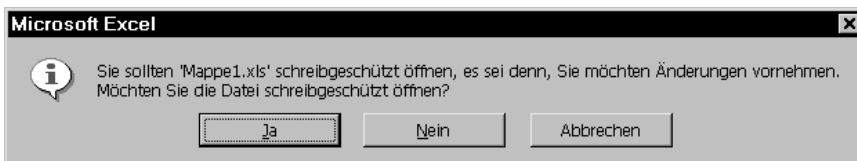


Abbildung 3.6:
Öffnen einer Datei
mit Schreibschutz-
Empfehlung

3.1.2 Dateieigenschaften

Ist im Register *Allgemein* des Befehls EXTRAS|OPTIONEN... das Kontrollkästchen *Anfrage nach Dateieigenschaften* aktiviert, erscheint unmittelbar vor dem Speichervorgang noch folgendes Dialogfeld (Abbildung 3.7).

Alle Informationen, die Sie im Register *Zusammenfassung* eingeben, werden zusammen mit der Datei gespeichert.

*Übrigens: Nur wenn Sie in diesem Register das Kontrollkästchen Vorschau-
grafik speichern aktivieren, ist im Öffnen-Dialogfeld, das ich in Kürze
erläutern werde, die Dateivorschau verfügbar!*



Wird die betreffende Datei später wieder geöffnet und wählen Sie DATEI|EIGENSCHAFTEN, erscheint das gleiche Dialogfeld, sodass Sie diese Datei-Informationen jederzeit einsehen oder auch nachträglich verändern können.

Die Hyperlink-Basis ist die Pfadangabe, auf die sich Hyperlinks mit relativen Adressen beziehen. Mit der Basis C:\EXCEL-BUCH verweist beispielsweise ein Hyperlink mit der relativen Adressangabe DEMOS\TEST.XLS auf die Datei C:\EXCEL-BUCH\DEMOS\TEST.XLS (siehe Kapitel 25, »Hyperlinks«).



Abbildung 3.7:
Dateieigenschaften

Mappe1.xls Eigenschaften

Tab: **Allgemein** | Zusammenfassung | Statistik | Inhalt | Anpassen

Titel:

Thema:

Autor:

Manager:

Firma:

Kategorie:

Stichwörter:

Kommentar:

Hyperlink-basis:

Vorlage:

☐ Vorschau grafik speichern

OK Abbrechen

Beide Register *Allgemein* und *Statistik* enthalten Informationen über die Dateieigenschaften: die Größe der Datei, den Typ etc., das Datum, an dem sie zuletzt verändert oder ausgedruckt wurde und so weiter.

Das Register *Inhalt* enthält die Namen aller Tabellen der Arbeitsmappe.

Das Register *Anpassen* ermöglicht die Festlegung benutzerdefinierter Dateieigenschaften (Abbildung 3.8).

Im Listenfeld *Name* können Sie unter verschiedenen Alternativen die gewünschte Eigenschaft auswählen, beispielsweise *Sprache*, *Kunde* oder *Abteilung*.

Darunter, im Feld *Typ*, selektieren Sie den Typ der Eigenschaft, *Text*, *Datum*, *Zahl* etc.

Im Feld *Wert* geben Sie den Wert der Eigenschaft für die aktuelle Mappe ein.

Um bei einem Projekt zusammen mit der Mappe Informationen über den Auftraggeber zu speichern, können Sie beispielsweise entsprechend der Abbildung als Eigenschaftsname »Kunde« selektieren, die vorgegebene Typeinstellung *Text* belassen und als Wert dieser Eigenschaft »Maier« eingeben, den Namen des Kunden.

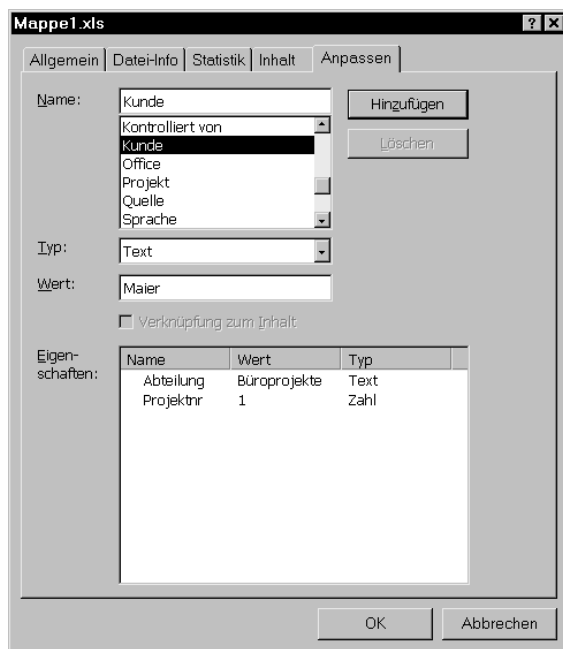


Abbildung 3.8:

Register
»Anpassen«

Hinzufügen fügt diese Informationen analog zu den dort bereits vorhandenen Informationen in das große Listenfeld ein.

Klicken Sie in diesem Listenfeld auf den Namen einer Eigenschaft, erscheinen die zugehörigen Daten in den oberen Feldern.

Sie können die betreffenden Informationen nun darin editieren und mit *Ändern* die geänderten Informationen im großen Listenfeld speichern oder sie daraus entfernen, indem Sie auf *Löschen* klicken.

Statt im Listenfeld Name einen der vorgegebenen Eigenschaftsnamen auszuwählen, können Sie darin auch nicht vorgesehene individuelle Bezeichnungen wie im Beispiel »Projektnr« eintippen.



Sie können Dateieigenschaften mit bestimmten Zellen der Arbeitsmappe verknüpfen.

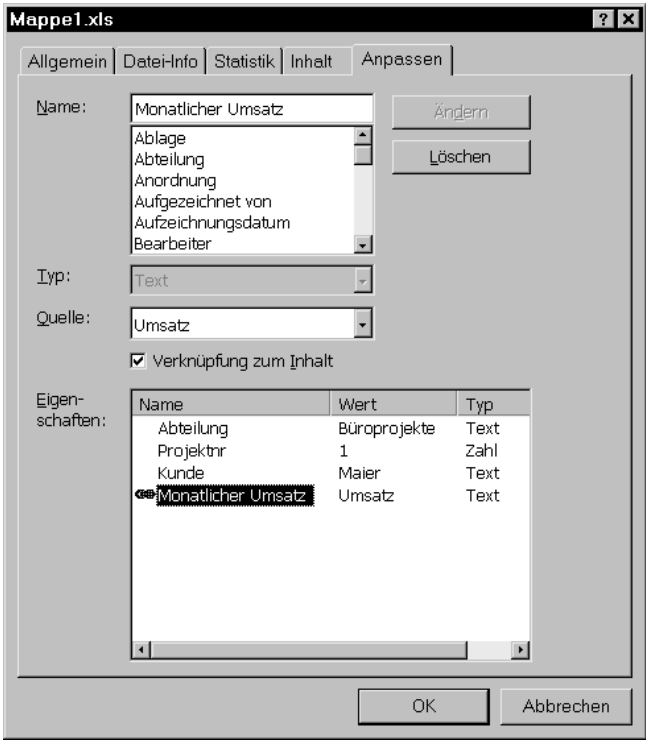
*Verknüpfung zum
Inhalt*

Voraussetzung dazu ist, dass die ausgewählten Zellen einen Namen besitzen (siehe Kapitel 9, »Namen«).



Nehmen wir an, Sie gaben der Zelle A2 einer Mappe den Namen »Umsatz«. Anschließend definieren Sie im *Eigenschaften*-Dialogfeld den benutzerdefinierten Namen »Monatlicher Umsatz« und aktivieren das Kontrollkästchen *Verknüpfung zum Inhalt* (Abbildung 3.9).

Abbildung 3.9:
Verknüpfung zum
Inhalt



Das Feld *Typ* ist nun deaktiviert und das Feld *Wert* heißt jetzt *Quelle*. Darin geben Sie die *Quelle der Eigenschaft* an, die Sie »Monatlicher Umsatz« nannten, also den Namen »Umsatz«, den Sie der Zelle A2 gaben.

Sie können diesen Namen aus dem Listefeld auswählen, da es alle in der Mappe vergebenen Namen enthält.

Hinzufügen fügt diese Eigenschaft in das große Listefeld ein, wobei das zusätzliche Symbol auf eine Verknüpfung dieser Eigenschaft mit dem Inhalt der Mappe hinweist. In der Spalte *Wert* wird der aktuelle Wert der Eigenschaft angezeigt, also der aktuelle Inhalt der Zelle A2.



Beim Öffnen von Dateien können Sie nach Dateien suchen, die bestimmte Eigenschaften aufweisen, genauer: nach den Werten der betreffenden Eigenschaften. Beispielsweise nach Dateien, die als Eigenschaft Kunde den Wert »Maier« enthalten.

3.1.3 Speichern unter...

Nach dem Speichern einer Datei zeigt Excel in der Titelleiste den vollständigen Dateinamen an, statt »Mappe2« zum Beispiel MAPPE2.XLS.

Speichern Sie die Datei später erneut, erscheint das erläuterte Dialogfeld nicht mehr. Excel geht davon aus, dass Sie diesen Namen auch bei folgenden Speichervorgängen verwenden wollen und überschreibt den alten Inhalt der Datei einfach mit dem aktuellen Inhalt der Tabelle.

Wollen Sie die Datei später dennoch unter einem anderen Namen speichern, wählen Sie DATEI|SPEICHERN UNTER... (Kurzaste **F12**), womit das gleiche Dialogfeld geöffnet wird, in dem beim ersten Speichervorgang nach dem Dateinamen gefragt wurde (siehe Bild 3.1).

Speichern unter...

Sie können nun einen anderen Namen eingeben und haben nach Aktivierung von *Speichern* zwei identische Dateien mit unterschiedlichen Namen auf der Festplatte.

Wenn Sie Excel beenden, nachdem Sie Ihre Mappe verändert hatten, ohne danach gespeichert worden zu sein, fordert Excel Sie automatisch zum Speichern auf, je nachdem, ob der Assistent aktiviert ist oder nicht, in einer moderneren oder eher konservativen Form (Abbildung 3.10).

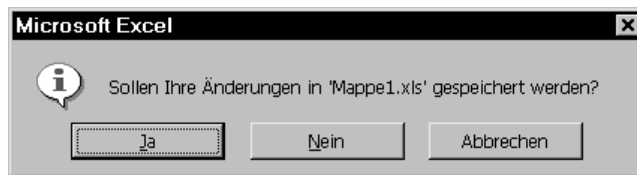


Abbildung 3.10:
Aufforderung zum
Speichern

Wählen Sie *Ja*, erscheint das *Speichern unter*-Dialogfeld, wenn die Datei noch nie gespeichert wurde. Ansonsten wird wie gewohnt die zuletzt gespeicherte Version überschrieben.

Nein beendet Excel, ohne zu speichern, und *Abbrechen* bricht den gesamten Vorgang zum Beenden von Excel ab.

Klicken Sie mit der rechten Maustaste auf die Titelleiste eines Tabellenfensters, erscheint ein Kontextmenü, das ebenfalls die Befehle SPEICHERN und SPEICHERN UNTER... enthält.



3.1.4 Arbeitsbereich speichern

Wählen Sie statt SPEICHERN oder SPEICHERN UNTER... den Befehl DATEI|AUFGABENBEREICH SPEICHERN..., so werden Sie wieder aufgefordert, einen Dateinamen einzugeben.

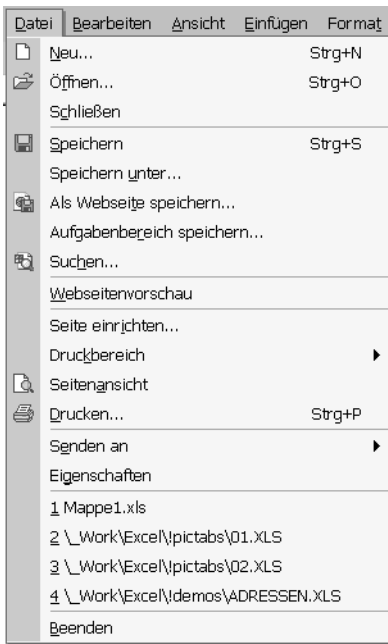
Geben Sie beispielsweise »Test« ein, wird diesmal jedoch eine Datei namens TEST.XLW angelegt, eine Arbeitsbereichsdatei, die *alle* momentan geöffneten Arbeitsmappen enthält, inklusive der aktuellen Fenstergröße und -position.

Öffnen Sie diese Datei am nächsten Tag, werden alle darin enthaltenen Mappen gleichzeitig geöffnet und erscheinen wieder an der alten Position und in der zuletzt eingestellten Fenstergröße.

3.2 Öffnen

Ganz unten im Menü DATEI werden die Namen der zuletzt benutzten Dateien angezeigt (ebenso wie im Register *Neue Arbeitsmappe* des Aufgabenbereichs). Klicken Sie einen an, öffnet Excel die betreffende Datei (Abbildung 3.11).

Abbildung 3.11:
Zuletzt bearbeitete
Dateien



Wollen Sie auf diese Anzeige verzichten, deaktivieren Sie das Kontrollkästchen Liste zuletzt geöffneter Dateien (im Register Allgemein des Befehls EXTRAS|OPTIONEN...).



Allerdings funktioniert diese Art des Öffnens nur mit den zuletzt verwendeten Dateien. Wesentlich allgemeiner ist der Befehl DATEI|ÖFFNEN... bzw. das Anklicken des zugehörigen Symbols der Standardleiste oder das Klicken auf den Eintrag *Weitere Arbeitsmappen...* des Aufgabenbereichs, da all diese Techniken das gleiche Dialogfeld zur Dateiauswahl öffnen (Abbildung 3.12).

Dieses Dialogfeld ist identisch mit dem beim Speichern von Dateien verwendeten, auch die Symbole im linken Dialogfeldteil und am oberen Rand sind identisch. Ihnen stehen somit dieselben Möglichkeiten zur Verfügung, den Ort auszuwählen, an dem sich die zu öffnende Datei befindet.

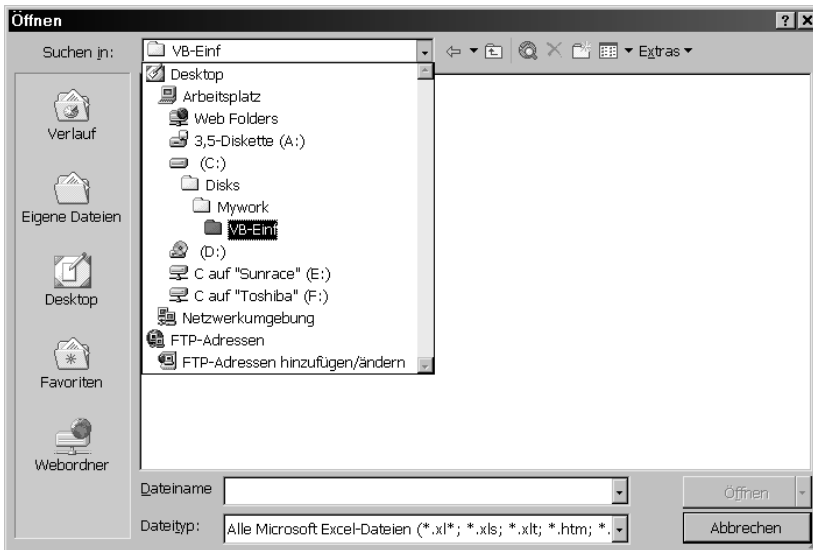


Abbildung 3.12:
Datei öffnen

Ich beschränke mich daher auf die Erläuterung der Besonderheiten beim Öffnen von Dateien.

Statt eine einzige Datei auszuwählen und per Schaltfläche Öffnen oder per Doppelklick auf die Datei zu öffnen, können Sie mehrere Dateien gleichzeitig selektieren und danach mit Öffnen automatisch alle selektierten Dateien nacheinander öffnen (siehe Kapitel 1.2.3, »Dialogfelder«).



3.2.1 Optionen beim Öffnen

Die Schaltfläche **Öffnen** ist zugleich ein Listenfeld. Öffnen Sie die Liste per Anklicken des Pfeils am rechten Schaltflächenrand, stehen Ihnen außer **Öffnen** folgende zusätzliche Optionen zur Verfügung:

- ➔ **SCHREIBGESCHÜTZT ÖFFNEN** öffnet die Datei genauso, als wäre sie beim Speichern mit einem Schreibschutz versehen worden. Beim späteren Speichern wird sie also nicht einfach überschrieben, sondern Sie werden nach einem neuen Namen gefragt.
- ➔ **ALS KOPIE ÖFFNEN** öffnet eine Kopie der Originaldatei, die anschließend unter einem beliebigen Namen gespeichert werden kann, unter anderem auch unter dem Namen der Originaldatei (den Sie dazu jedoch ausdrücklich angeben müssen, sodass ein versehentliches Überschreiben des Originals dadurch ebenfalls verhindert wird).
- ➔ **IM BROWSER ÖFFNEN** bezieht sich nicht auf Excel-Dateien, sondern auf Webseiten (Endung HTM oder HTML) und öffnet diese im daraufhin gestarteten Webbrowser.

➔ **ÖFFNEN UND REPARIEREN** öffnet ein Dialogfeld, in dem Sie wählen können, ob Excel versuchen soll, die betreffende Datei zu reparieren oder – erst sinnvoll, wenn das scheiterte – soweit möglich die darin enthaltenen Daten zu extrahieren.

Dateitypen Im Feld *Dateityp* wählen Sie aus, welche Dateien angezeigt werden. Der vorgegebene Eintrag »Microsoft Excel-Dateien...« führt dazu, dass nur Excel-Dateien angezeigt werden, das heißt Dateien, die eine der von Excel verwendeten Endungen besitzen.

Stattdessen können Sie auch eine beliebige andere Selektionsmaske auswählen, beispielsweise »Alle Dateien (*.*)«, um alle Dateien anzeigen zu lassen.

Dateiname Alternativ dazu können Sie im Feld *Dateiname* eine Selektionsmaske eintippen. »*.XL*« zeigt beispielsweise ausschließlich Excel-Dateien an, das heißt Dateien, bei denen die ersten beiden Buchstaben der Erweiterung XL lauten. »**.TXT« würde nur Dateien mit der Endung .TXT anzeigen und »**.« alle Dateien ohne Endung.

Zusätzlich können Sie auch den Suchpfad eintippen. Beispielsweise zeigt

C:\EXCEL\DATEN\TEST.*

nur Dateien an, die den Namen TEST und eine beliebige Erweiterung besitzen und sich im Ordner EXCEL\DATEN von Laufwerk C: befinden. Und

C:\EXCEL\DATEN\TEST.XLS

würde nur die Datei TEST.XLS dieses Ordners anzeigen (der Zusatz .XLS ist übrigens überflüssig, da Excel ihn automatisch an den von Ihnen angegebenen Namen anhängt, wenn dieser keine Erweiterung besitzt).

Haben Sie den Dateinamen beim Speichern mit einem Punkt abgeschlossen (»TEST.«), um zu verhindern, dass Excel eine Endung ergänzt, müssen Sie diesen Punkt auch beim Laden angeben.

Mit der Eingabe »HALLO« können Sie eine unter diesem Namen gespeicherte Datei nicht laden, da Excel automatisch nach HALLO.XLS suchen wird. Erst die Eingabe »HALLO.« macht Excel klar, dass der Dateiname tatsächlich HALLO ohne jede Endung ist.

Sie können auch den Namen eines Verzeichnisses eingeben, in dem Sie Ihre Datei vermuten, zum Beispiel C:\EXCEL\DATEN. Dann zeigt Ihnen Excel alle Dateien dieses Verzeichnisses, die der aktuellen Selektionsmaske entsprechen.

Die Eingabe eines Pfades und einer Maske können Sie auch kombinieren und sich zum Beispiel mit »c:\excel\daten*. *« alle Dateien dieses Verzeichnisses zeigen lassen.

3.2.2 Dateien wiederherstellen

Waren Sie nach vorgenommenen Änderungen nicht mehr in der Lage, eine Datei zu speichern (Stromausfall, Rechnerabsturz etc.), oder trat gar während des Speicherns ein Problem aus, bietet Ihnen Excel an, beim nächsten Start alle davon betroffenen Dokumente wiederherzustellen – soweit möglich (Abbildung 3.13).

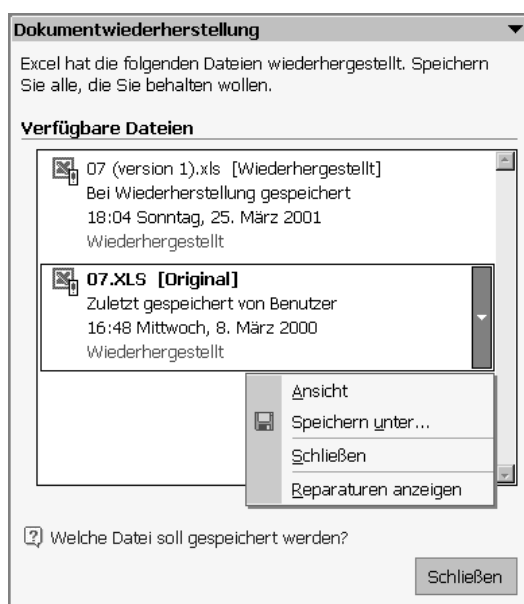


Abbildung 3.13:
Dokumente wiederherstellen

Die betroffenen Dokumente werden aufgelistet und Sie klicken auf das gewünschte, um es zu öffnen, oder auf den Listenpfeil eines Dokuments, um die zusätzlichen Optionen zu erhalten, das betreffende Dokument zu speichern oder die vorgenommenen Reparaturen einzusehen (in beiden Fällen wird es zusätzlich auch geöffnet). Anschließend verschwindet das wiederhergestellte Dokument aus der Liste.

4 Selektieren

Fast alles, was Sie mit Excel anstellen können, beruht auf dem Auswählen einzelner Zellen oder mehr oder weniger umfangreicher Zellbereiche. Entsprechend geht es nun um dieses Auswählen oder auch »Selektieren«.

Zellbereiche können sowohl per Tastatur als auch mit der Maus selektiert werden – wobei es jedoch nicht ganz ohne die Tastatur geht, wenn Sie komplexe Gebilde aus nicht zusammenhängenden Zeilen und Spalten auswählen wollen.

Darüber hinaus können Sie per Mausklick auch das gesamte Arbeitsblatt selektieren, beispielsweise um mit einem Schlag allen Zellen der Tabelle eine andere Schriftart oder Schriftgröße zuzuweisen.

4.1 Bereiche

Sie selektieren einen Bereich, indem Sie

- ➔ entweder die erste Bereichsecke und danach *bei gleichzeitig gedrückter* -Taste die entgegengesetzte Ecke anklicken
- ➔ oder aber die Maus über den betreffenden Bereich ziehen *und dabei die linke Maustaste gedrückt halten*.

In beiden Fällen wird die Zellselektion ausgeweitet, was Excel durch eine dunkle Tönung der markierten Fläche anzeigt.

Probieren Sie das aus, indem Sie den Bereich B5 bis C10 selektieren, in der Excel-Kurzform ausgedrückt:

B5:C10

Klicken Sie B5 an, steuern Sie den Mauscursor zu C10 und klicken Sie diese Zelle bei gedrückter -Taste an (Methode 1) oder ziehen Sie die Maus bei gedrückter -Taste über den Bereich (Methode 2) (Abbildung 4.1).

Der Ausgangspunkt B5 ist zwar heller hervorgehoben, gehört aber dennoch zum selektierten Bereich! Die Hervorhebung bedeutet, dass B5 die aktive Zelle im selektierten Bereich ist.

Abbildung 4.1:
Bereich B5: C10
selektieren

	A	B	C	D
1	23	76	23	76
2	64	34	64	34
3	22	56	22	56
4	56	9	56	9
5	165	175	165	175
6	23	76	23	76
7	64	34	64	34
8	22	56	22	56
9	56	9	56	9
10	165	175	165	175
11	22	56	22	56
12	56	9	56	9
13	299	249	299	249
14				

Sowohl mit der Maus als auch mit der Tastatur können Sie Selektionen über den sichtbaren Ausschnitt der Tabelle hinaus ausdehnen. Excel scrollt den Ausschnitt automatisch in die gewünschte Richtung, wenn Sie einen Fensterrand erreichen.



Mit den Cursortasten wird ein Tabellenbereich selektiert, indem Sie den Zellcursor zu einer Ecke des Bereichs bewegen, die -Taste drücken und diese Taste gedrückt halten, während Sie anschließend den Cursor bewegen.

Bewegen Sie den Zellcursor daher zu B5, drücken Sie die -Taste und *halten Sie sie gedrückt*, während Sie den Zellcursor nun zur entgegengesetzten Bereichsecke steuern, zu C10. Haben Sie diese Zelle erreicht, lassen Sie die -Taste wieder los.

Alle Möglichkeiten zur schnelleren Bewegung im Arbeitsblatt können Sie natürlich auch beim Selektieren von Bereichen nutzen.



*Sie können zum Markieren nicht nur die vier Pfeiltasten zur Cursorsteuerung einsetzen, sondern auch **[Strg]+[↓]**, **[Strg]+[Bild↓]** etc. (siehe Kapitel 1.3.1, »Bildlauf«), um größere Strecken schnell zurückzulegen und dabei alle übersprungenen Zellen zu markieren. Sie müssen nur darauf achten, gleichzeitig immer die -Taste zu drücken, was manchmal zu etwas akrobatischen Dreitasten-Kombinationen führt.*

Notebook-Tastaturen

Wenn Sie ein Notebook benutzen, funktioniert die Selektion per -Taste möglicherweise nicht. Ursache dafür sind die teilweise deutlich vom Standard abweichenden Notebook-Tastaturen.

Die Lösung: Sie begeben sich zu einer Ecke des gewünschten Bereichs, drücken die Taste **[F8]** und lassen sie wieder los. In der Statusleiste erscheint die

Meldung *ERW*, die darauf hinweist, dass Sie sich nun im erweiterten Modus befinden, in dem Sie mehr als nur eine Zelle selektieren können.

Dazu steuern Sie einfach wie zuvor beschrieben den Zellcursor in die gewünschte Richtung – ohne gleichzeitig irgend eine weitere Taste zu drücken! Der erweiterte Modus bleibt auch so erhalten.

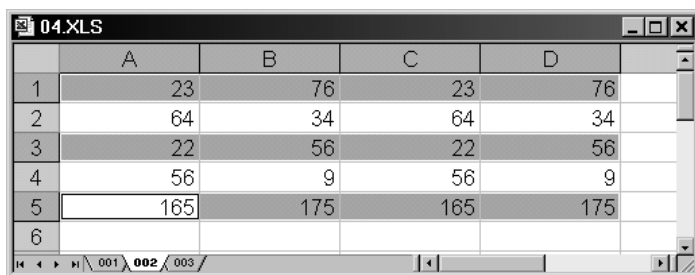
Wollen Sie die Selektion aufheben, drücken Sie erneut **[F8]**. Jede Betätigung der Cursorasten hebt nun die Markierung auf und bewegt den Zellcursor wieder normal.

4.2 Mehrfachselektion

Mehrfachselektion bedeutet eine Auswahl mehrerer rechteckiger Bereiche. Selektieren Sie zunächst mit der Maus den ersten Bereich, sagen wir A1:D1.

Um nun als zweiten Bereich A3:D3 zu selektieren, drücken Sie zunächst die **[Strg]**-Taste. Klicken Sie dann den Bereichsanfang A3 an. Halten Sie die Maustaste gedrückt und selektieren Sie den Bereich A3:D3. Wenn Sie den Beginn des zweiten Bereichs angeklickt haben, können Sie die **[Strg]**-Taste während der Selektion ruhig wieder loslassen.

Selektieren Sie nun als dritten Bereich A5:D5. Drücken Sie **[Strg]**, klicken Sie A5 an und ziehen Sie die Maus bei gedrückter linker Maustaste bis D5 (Abbildung 4.2).



	A	B	C	D
1	23	76	23	76
2	64	34	64	34
3	22	56	22	56
4	56	9	56	9
5	165	175	165	175
6				

Abbildung 4.2:
Mehrfachselektion

Eine Alternative zu dieser **[Strg]**-Technik ist folgende Methode: Sie ziehen die Maus bei gedrückter Maustaste über den ersten Bereich, um ihn zu selektieren.

Dann drücken Sie – einmal, nicht gedrückt halten – **[⇧] + [F8]**, um den Hinzufügemodus einzuschalten. Anschließend bewegen Sie den Mauspfel zum zweiten Bereich und ziehen die Maus bei gedrückter Maustaste auch über diesen Bereich, danach über den dritten, vierten Bereich und so weiter, beliebig oft.

Bei dieser Technik bleibt der Hinzufügemodus solange eingeschaltet, bis Sie ihn explizit wieder ausschalten, indem Sie noch einmal $\boxed{\text{⇧}} + \boxed{\text{F8}}$ oder einfach $\boxed{\text{Esc}}$ drücken.



Mit der Tastatur selektieren Sie den ersten Bereich. Diese Selektion wird jedoch sofort wieder aufgehoben, wenn Sie nun – ohne gedrückte $\boxed{\text{⇧}}$ -Taste – den Cursor bewegen, zum Beispiel zum Beginn des zweiten Bereichs.

Beibehalten wird diese Selektion nur, wenn Sie die Auswahl mit $\boxed{\text{⇧}} + \boxed{\text{F8}}$ hinzufügen. Drücken Sie diese Taste genau einmal, erscheint in der Statusleiste ADD als Hinweis, dass Sie dem selektierten Bereich nun einen weiteren hinzufügen können.

Gehen Sie zum Beginn dieses zweiten Bereichs, selektieren Sie ihn, drücken Sie erneut $\boxed{\text{⇧}} + \boxed{\text{F8}}$ und selektieren Sie auch den dritten Bereich.

4.3 Spalten, Zeilen und das Arbeitsblatt

Eine komplette Spalte selektieren Sie durch Anklicken des Spaltenbuchstaben, eine Zeile entsprechend durch Anklicken der Zeilennummer. Durch Anklicken des kleinen Kästchens in der oberen linken Fensterecke, zwischen den Zeilennummern und den Spaltenbuchstaben, selektieren Sie die gesamte Tabelle.

Die Spalten B und D und zusätzlich die Zeile 3 selektieren Sie daher, indem Sie

- ➔ den Spaltenbuchstaben B anklicken,
- ➔ bei gedrückter $\boxed{\text{Strg}}$ -Taste den Spaltenbuchstaben D anklicken
- ➔ und bei gedrückter $\boxed{\text{Strg}}$ -Taste die Zeilennummer 3 anklicken (Abbildung 4.3).

Abbildung 4.3:
Mehrfachselektion
von Zeilen und
Spalten

	A	B	C	D
1	23	76	23	76
2	64	34	64	34
3	22	56	22	56
4	56	9	56	9
5	165	175	165	175
6				



Wollen Sie das mit der Tastatur machen, so bewegen Sie den Zellcursor in die gewünschte Spalte oder Zeile und drücken Sie $\boxed{\text{↵}} + \boxed{}$, um die Zeile zu selektieren, beziehungsweise $\boxed{\text{Strg}} + \boxed{}$ für die Spalte. $\boxed{\text{Strg}} + \boxed{\text{↵}} + \boxed{}$ selektiert das gesamte Arbeitsblatt.

Selbstverständlich können Sie für Mehrfachselektionen auch hier wieder mit $\boxed{\text{↵}} + \boxed{\text{F8}}$ den Hinzufügemodus einschalten.

Allerdings verhält sich Excel dabei etwas merkwürdig. Angenommen, es sollen entsprechend der Abbildung die Spalten B und D und zusätzlich die Zeile 3 selektiert werden, dann selektieren Sie zuerst Spalte B, schalten mit $\boxed{\text{↵}} + \boxed{\text{F8}}$ den Hinzufügemodus ein und selektieren die Spalte D.

Im Gegensatz zu vorhin bleibt diesmal das **ADD** erhalten. Also verlassen Sie sich darauf und bewegen den Zellcursor in Zeile 3, um die komplette Zeile zu selektieren. Aber im gleichen Moment, in dem Sie eine Cursortaste betätigen, wird die Markierung von Spalte D aufgehoben!

Meine Versuche ergaben, dass nach der Markierung zweier Spalten oder Zeilen zwar **ADD** in der Statusleiste steht, der Hinzufügemodus tatsächlich jedoch ausgeschaltet ist und mit $\boxed{\text{↵}} + \boxed{\text{F8}}$ explizit wieder eingeschaltet werden muss. Gehen Sie daher so vor:

- ➔ Bewegen Sie den Zellcursor zu irgendeiner Zelle von Spalte B und drücken Sie $\boxed{\text{Strg}} + \boxed{}$, um diese Spalte zu selektieren.
- ➔ Da Ihre Selektion noch nicht beendet ist, schalten Sie mit $\boxed{\text{↵}} + \boxed{\text{F8}}$ den Hinzufügemodus ein.
- ➔ Gehen Sie zu einer Zelle von Spalte D und drücken Sie wieder $\boxed{\text{Strg}} + \boxed{}$, um zusätzlich diese Spalte zu selektieren.
- ➔ Glauben Sie dem **ADD** in der Statusleiste nicht, sondern drücken Sie $\boxed{\text{↵}} + \boxed{\text{F8}}$: **ADD** verschwindet. Drücken Sie erneut $\boxed{\text{↵}} + \boxed{\text{F8}}$: **ADD** erscheint wieder und diesmal ist der Hinzufügemodus tatsächlich eingeschaltet.
- ➔ Gehen Sie in eine der Zellen von Zeile 3 und selektieren Sie diese Zeile mit $\boxed{\text{↵}} + \boxed{}$.

5 Editieren

Selektierte Zellen oder Zellbereiche können Sie nach Belieben editieren, also ihren Inhalt verändern, sie verschieben, kopieren und so weiter. All diese Möglichkeiten zeige ich Ihnen anhand des Beispiels »Kostenübersicht«, einer Tabelle, die harmlos beginnt und Schritt für Schritt weiterentwickelt und verfeinert wird.

Zunächst erläutere ich, wie Sie sich die Anwahl all dieser Aktionen per Kontextmenü vereinfachen können. Danach geht es um die Eingabe von Daten in einen zuvor selektierten Zellbereich, bei der der Cursor automatisch der Reihe nach jede Zelle dieses Bereichs abläuft, sodass keinerlei Cursortasten zur Steuerung benötigt werden.

Anschließend zeige ich Ihnen, wie Sie Zellbereiche verschieben, kopieren und löschen, Bereiche aus Leerzellen einfügen oder den Inhalt eines Bereichs löschen.

(Fast) alles ist außer per Tastatur und Menükommandos auch mithilfe verschiedenster Maus-Ziehaktionen machbar, wobei es in allen Fällen ausreicht, nach der Selektion des Bereichs an einer Bereichsrandlinie zu ziehen (=Bereich verschieben) bzw. das kleine Kästchen in der rechten unteren Bereichsecke entweder in den selektierten Bereich hinein- oder daraus herauszuziehen.

Auf das Kopieren gehe ich besonders detailliert ein, da es in der Praxis wohl die am häufigsten benutzte Editieraktion ist – und die umfangreichsten Möglichkeiten bietet!

Sie können Bereiche nämlich nicht nur 1:1 kopieren, sondern nahezu beliebig auswählen, ob die Kopie nur die Zahlen und Formeln des Originals enthalten soll oder auch die gleichen Formatierungen (kursiv, fett etc.) bzw. ob auch mit bestimmten Zellen des Originalbereichs verbundene Kommentare kopiert werden sollen.

Besonders interessant ist, dass beim Kopieren die Anordnung der Spalten und Zeilen vertauscht werden kann (Transponieren eines Bereichs) und dass die zu kopierenden Werte mithilfe verschiedenster Rechenoperationen mit Werten verknüpft werden können, die sich bereits im Zielbereich befinden, beispielsweise um sie zu diesen Werten zu addieren.

Sogar dynamische Verknüpfungen mit dem Originalbereich sind möglich, die bewirken, dass jede spätere Änderung im Originalbereich automatisch auf die Kopie übertragen wird, wodurch die Kopie ständig aktuell und ein erneutes Kopieren des geänderten Originalbereichs überflüssig ist.

Oft sehr nützlich ist das Füllen von Bereichen, das Ihnen ermöglicht, einmal begonnene Reihen wie »2000«, »2001«, »2002« oder »Jan«, »Feb«, »Mae« automatisch fortsetzen zu lassen – wobei Excel auch weitaus komplexere Reihen fortsetzen kann, beispielsweise geometrische Reihen – und sogar lineare und exponentielle Trendreihen ermittelt!

Zum Editieren gehört auch das Suchen und Ersetzen bestimmter Zellinhalte und die Auswahl bestimmter Datenarten, beispielsweise aller Zellen eines Bereichs, die Formeln enthalten.



Einige der in den folgenden Kapiteln gezeigten Symbole befinden sich in der Standardleiste. Den Rest finden Sie in den Kategorien Bearbeiten bzw. Einfügen im Register Befehle des Anpassen-Dialogfelds (Befehl ANSICHT|SYMBOLLEISTEN|ANPASSEN...).

5.1 Rückgängig machen und Wiederholen



Ist Ihnen während einer Editieraktion ein Fehler unterlaufen, können Sie diesen unmittelbar danach noch mit dem Befehl BEARBEITEN|RÜCKGÄNGIG:... beziehungsweise dem entsprechenden Symbol der Standardleiste ungeschehen machen.

An das Symbol ist übrigens ein Listenfeld gebunden. Im geöffneten Listenfeld sind die zuletzt durchgeführten Aktionen aufgelistet, die neueste oben und die älteste unten. Durch das Ziehen der Maus nach unten selektieren Sie mehrere Aktionen und wählen damit aus, ob nur die letzte oder auch die vorletzte bzw. die davor erfolgte Aktion gemacht werden soll.



Umgekehrt können Sie die letzte Aktion mit BEARBEITEN|WIEDERHOLEN:... beziehungsweise dem entsprechenden Symbol der Standardleiste erneut ausführen, wobei sich mit dem an das Symbol gebundenen Listenfeld erneut mehrere Aktionen wiederholen lassen.

5.2 Kontextmenüs

Unter dem Editieren einer Tabelle versteht man das Löschen, Verschieben, Kopieren oder Sortieren einzelner Zellen oder ausgewählter Bereiche. Dafür sind vorwiegend die Befehle des Menüs BEARBEITEN zuständig. Sie können

sich die Anwahl dieser Befehle jedoch mit mehreren Kontextmenüs ersparen. Um eine einzelne Zelle zu editieren, klicken Sie sie einfach mit der *rechten* Maustaste an.

Das Zellen-Kontextmenü erscheint, das jene Befehle der Menüs BEARBEITEN und EINFÜGEN enthält, die erfahrungsgemäß am häufigsten beim Editieren einzelner Zellen verwendet werden (Abbildung 5.1).



Abbildung 5.1:
Das Zellen-Kontextmenü

ZELLEN EINFÜGEN... entspricht beispielsweise dem Befehl EINFÜGEN|ZELLEN... und ZELLEN LÖSCHEN... dem Befehl BEARBEITEN|ZELLEN LÖSCHEN... .

Wollen Sie einen Zellbereich editieren, selektieren Sie ihn wie gewohnt, lassen die Maustaste los und klicken anschließend *irgendeine* Zelle dieses Bereichs mit der rechten Maustaste an.

Wollen Sie eine komplette Spalte oder Zeile editieren, können Sie sie selektieren und *gleichzeitig* das zugehörige Spalten- beziehungsweise Zeilen-Kontextmenü öffnen, indem Sie den Spaltenbuchstaben bzw. die Zeilennummer mit der *rechten* Maustaste anklicken.

Noch nützlicher ist das Ziehen-Kontextmenü. Praktisch alle nachfolgend erläuterten Techniken zum Verschieben, Kopieren oder Einfügen von Bereichen werden dadurch vereinfacht.

**:-)
TIPP**

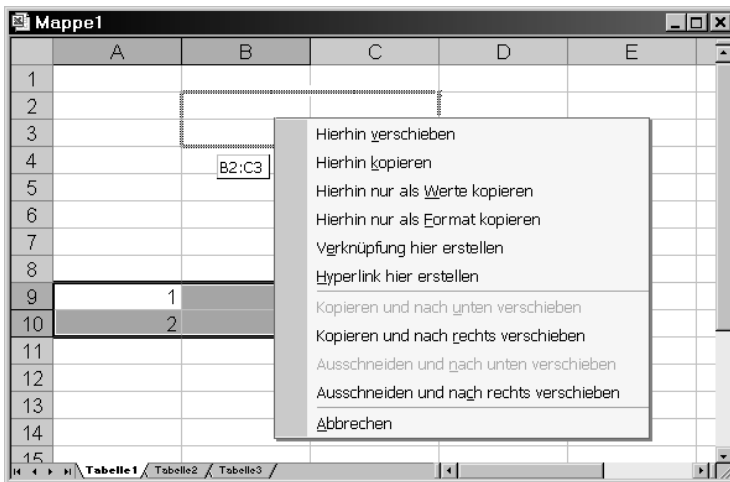
Diese Techniken sollten Sie zunächst anhand der erläuterten Menübefehle nachvollziehen. Sind Sie damit vertraut, wird es jedoch Zeit, umzusteigen: Klicken Sie auf die Sie interessierende Zelle bzw. markieren Sie den entsprechenden Bereich und lassen Sie die Maustaste wieder los.

Bewegen Sie den Mauscursor anschließend zu einem der Bereichsränder, wo er seine Form ändert und zum Cursorpfeil wird. Nun klicken Sie den Rand mit der *rechten* Maustaste an und ziehen den Bereich zur gewünschten Position, um ihn dort fallen zu lassen.

Im automatisch erscheinenden Kontextmenü können Sie den passenden Befehl auswählen, um den Bereich zur aktuellen Position zu verschieben oder ihn dorthin zu kopieren.

Wollen Sie vermeiden, dort vorhandene Zellen zu überschreiben, können Sie diese zuvor noch in eine beliebige Richtung verschieben lassen (Abbildung 5.2).

Abbildung 5.2:
Ziehen-Kontext-
menü



5.3 Demo »Kostenübersicht«

In das Thema *Editieren* werde ich Sie anhand einer kleinen Anwendung einführen, die sich mit einer fiktiven Klein- oder Einpersonenfirma beschäftigt und Schritt für Schritt weiterentwickelt wird. Jede Entwicklungsstufe ist in einer Tabelle der Mappe EDIT.XLS gespeichert, die sich auf der Begleit-CD befindet.

Kopieren Sie diese Mappe bitte auf Ihre Festplatte. *Sie sollten niemals mit der Originaldiskette arbeiten!*

Immer dann, wenn wir einen Schritt weiter sind, zeige ich Ihnen anhand einer Abbildung, wie Ihr Arbeitsblatt inzwischen aussehen soll: Unter vielen dieser Abbildungen finden Sie den Namen der zugehörigen Datei, in der dieser Zustand der Tabelle gespeichert ist und die Sie jederzeit aktivieren können, um sich näher über ihren Aufbau zu informieren oder sie als Ausgangspunkt für die Weiterentwicklung zu verwenden.

Sie besitzen eine Kleinfirma, die EDV-Produkte vertreibt, die Sie aus Taiwan importieren: Rechner, Drucker, Monitore und so weiter. Sie wollen sich mit Excel einen Überblick über die Entwicklung der verschiedenen Kosten im gerade abgelaufenen Jahr verschaffen. Sie interessieren sich speziell für

Datenstruktur

- ➔ Einkaufskosten
- ➔ Lagermiete
- ➔ Werbekosten
- ➔ Kosten für Büromaterial und Büromöbel
- ➔ Büromiete

Diese Posten wollen Sie nach Monaten getrennt erfassen. Fünf Posten, geordnet nach zwölf Monaten, diese Datenstruktur verlangt geradezu danach, für jeden Posten eine eigene Spalte zu verwenden, wobei die ersten zwölf Zeilen dieser Spalten den Monaten Januar bis Dezember entsprechen.

Zum Beispiel werden in die Zellen A1:A12 die monatlichen Einkaufskosten eingegeben: in A1 die Einkaufskosten vom Januar, in A2 die Einkaufskosten vom Februar und so weiter. In den Bereich B1:B12 wird entsprechend der monatliche Umsatz eingetragen und so weiter.

5.4 Dateneingabe in selektierten Bereichen

Die Eingabe größerer Datenmengen können Sie sich vereinfachen, indem Sie zunächst die Bereiche Ihrer Tabelle selektieren, in denen die Eingabe erfolgen soll, im Beispiel den Bereich A1:E12 (Abbildung 5.3).

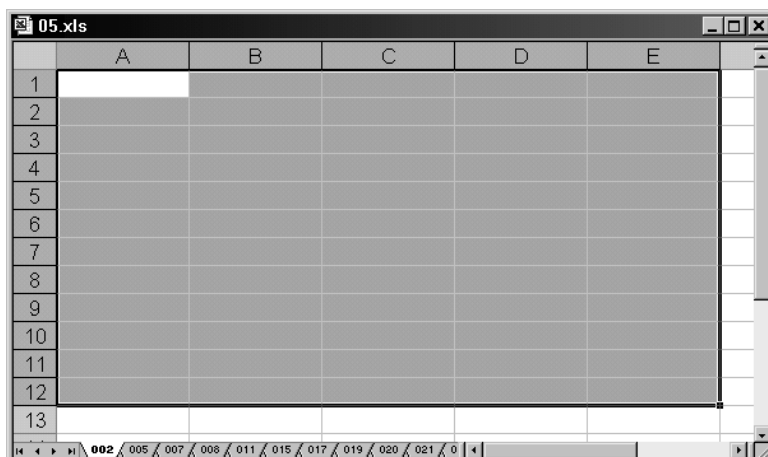


Abbildung 5.3:
Bereich A1: E12
selektieren

Die aktive A1 dieses Bereichs (hell hervorgehoben) ist Ihr Ausgangspunkt. Die Bereichsmarkierung wird nicht aufgehoben, wenn Sie in diese Zelle Daten eingeben!

Schließen Sie eine Eingabe in einen selektierten Bereich mit  ab, wird der Zellcursor wie üblich automatisch auf die darunter liegende Zelle gesetzt, sodass Sie die Dateneingabe direkt fortsetzen können – und zwar auch dann, wenn *Markierung nach dem Drücken der Eingabetaste verschieben* deaktiviert ist (im Register *Bearbeiten* von EXTRAS|OPTIONEN...)!

Wenn Sie jedoch das untere Bereichsende bei A12 erreichen und nach der Eingabe erneut  drücken, bewegt sich der Zellcursor *zur obersten Zelle der nächsten markierten Spalte*, also zu B1. Sie verlassen den markierten Bereich daher nicht, sondern *durchwandern ihn automatisch in der richtigen Reihenfolge*.



Eines dürfen Sie während der Eingabe keinesfalls: eine Cursortaste betätigen oder eine Maustaste drücken, da die Selektion in diesem Fall sofort aufgehoben wird!

Benutzen Sie daher zum Ändern eines Zelleninhalts , um den Bearbeitungsmodus einzuschalten, und steuern Sie den Zellcursor mit den folgenden Tastenkombinationen:

Tabelle 5.1:
Cursorsteuerung im
selektierten Bereich

Taste	Funktion
	Eine Zelle nach unten
 + 	Eine Zelle nach oben
	Eine Zelle nach rechts
 + 	Eine Zelle nach links
 +, 	Nächste Ecke des Bereichs



Um zur nächsten Bereichsecke zu gelangen, müssen Sie – zusammen mit  – die Komma-Taste des (aktivierten) Zahlenblocks Ihrer Tastatur benutzen – nicht die der normalen alphanumerischen Tastatur (Abbildung 5.4).

Geben Sie bitte auch in H1:I15 die abgebildeten Zahlen ein. Sie haben zwar nichts mit der eigentlichen Anwendung zu tun, sind jedoch in Kürze für das Thema *Editieren* wichtig.

Ist Ihnen das Eintippen zu aufwendig, öffnen Sie die zugehörige Datei EDIT.XLS und aktivieren das Blatt »Tabelle1« dieser Arbeitsmappe (Klicken auf den entsprechenden Namen in der Registerleiste).

	A	B	C	D	E	F	G	H	I
1	107822	1047	8560	834	1200			1	16
2	95927	1047	7600	302	1200			2	17
3	97264	1080	9304	280	1200			3	18
4	122846	1080	8560	150	1200			4	19
5	103846	1080	7890	274	1320			5	20
6	114963	1080	10845	720	1320			6	21
7	118370	1200	9570	549	1320			7	22
8	130827	1200	13065	839	1320			8	23
9	133840	1200	12085	224	1450			9	24
10	128746	1200	18563	172	1450			10	25
11	163790	1800	19750	642	1450			11	26
12	174972	1800	21510	830	2700			12	27
13								13	28
14								14	29
15								15	30
16									

Abbildung 5.4:
Edit.xls, »Tabelle1«

Wurden die Kosten wie hier in eine übersichtliche Form gebracht, ist die Auswertung relativ einfach und beispielsweise klar zu erkennen, dass die Werbekosten (Spalte C), die in den ersten neun Monaten jeweils mäßig stiegen, in den letzten drei Monaten auf einmal schlagartig stiegen.

Entweder wurden die Werbeaktivitäten in diesen Monaten erhöht oder Ihre Werbeagentur wird allmählich zu teuer und Sie sollten einen Wechsel in Betracht ziehen.

5.5 Verschieben

Für eine Analyse der Zahlen ist es denkbar ungünstig, wenn nicht auf den ersten Blick hervorgeht, welche Spalte welche Kostenart enthält.

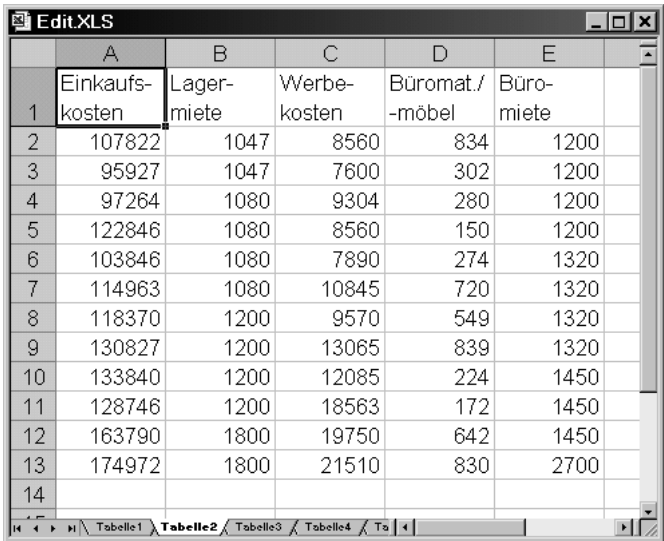
Wir benötigen für jede Spalte eine aussagekräftige Überschrift. Also müssen wir den gesamten Datenblock um eine Zeile nach unten verschieben, um am oberen Rand des Arbeitsblatts eine Leerzeile einzufügen.

Am einfachsten per Drag&Drop (Ziehen und Ablegen):

- ➔ Selektieren Sie den zu verschiebenden Bereich A1:E12 und lassen Sie die Maustaste los.
- ➔ Bewegen Sie den Mauscursor zu einem der Bereichsränder. Dort ändert er seine Form und wird zu einem Pfeil.
- ➔ Nun klicken Sie den Rand an und ziehen den Bereich zur gewünschten Position, in unserem Fall eine Zeile nach unten.
- ➔ Lassen Sie die Maustaste wieder los und den Bereich dadurch an der neuen Position fallen.

Danach geben Sie die benötigten Überschriften ein. Die Zeilenumbrüche erzielen Sie, indem Sie während der Eingabe an der entsprechenden Position (nach den Bindestrichen) die Tastenkombination **[Alt] + [↵]** betätigen (Abbildung 5.5).

Abbildung 5.5:
Edit.xls, »Tabelle2«

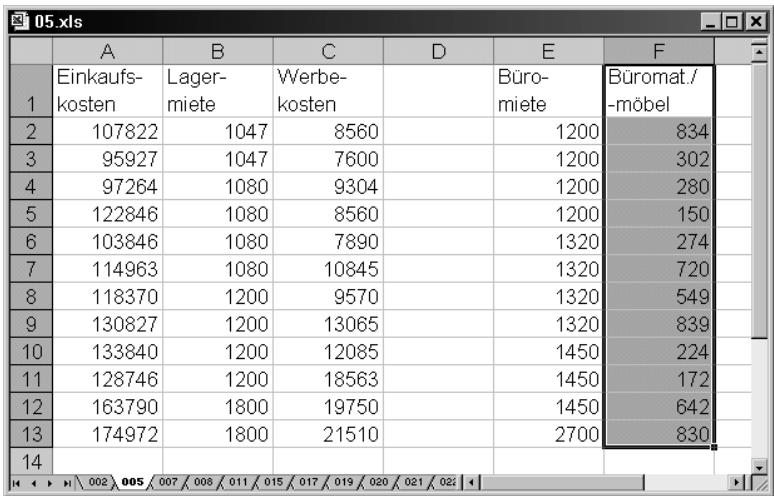


	A	B	C	D	E
1	Einkaufskosten	Lagermiete	Werbekosten	Büromat./-möbel	Büromiete
2	107822	1047	8560	834	1200
3	95927	1047	7600	302	1200
4	97264	1080	9304	280	1200
5	122846	1080	8560	150	1200
6	103846	1080	7890	274	1320
7	114963	1080	10845	720	1320
8	118370	1200	9570	549	1320
9	130827	1200	13065	839	1320
10	133840	1200	12085	224	1450
11	128746	1200	18563	172	1450
12	163790	1800	19750	642	1450
13	174972	1800	21510	830	2700
14					

TIPP

Mit der Tastatur selektieren Sie ebenfalls zunächst den zu verschiebenden Bereich A1:E12. Danach wählen Sie **BEARBEITEN|AUSSCHNEIDEN**. Der Bereich wird nicht sofort ausgeschnitten, sondern zunächst nur von einem Laufrahmen umgeben. Sie geben das Verschiebeziel an, indem Sie den Cursor dorthin steuern, wo sich nach dem Verschieben die linke obere Bereichsecke befinden soll, also zu A2.

Abbildung 5.6:
D1: D13 nach F1:
F13 verschieben

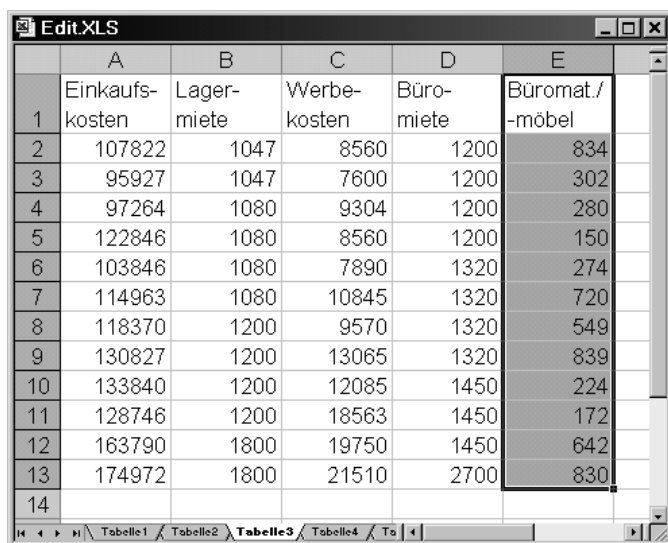


	A	B	C	D	E	F
1	Einkaufskosten	Lagermiete	Werbekosten		Büromiete	Büromat./-möbel
2	107822	1047	8560		1200	834
3	95927	1047	7600		1200	302
4	97264	1080	9304		1200	280
5	122846	1080	8560		1200	150
6	103846	1080	7890		1320	274
7	114963	1080	10845		1320	720
8	118370	1200	9570		1320	549
9	130827	1200	13065		1320	839
10	133840	1200	12085		1450	224
11	128746	1200	18563		1450	172
12	163790	1800	19750		1450	642
13	174972	1800	21510		2700	830
14						

Danach wählen Sie BEARBEITEN|EINFÜGEN oder drücken einfach . Der selektierte Bereich wird an der alten Position ausgeschnitten und an der neuen eingefügt.

Das Verschieben von Bereichen ermöglicht sogar komplexe Aktionen wie das Vertauschen von Bereichen. Wollen Sie die Kostenarten »Büromaterial« und »Büromiete« miteinander vertauschen, verschieben Sie zunächst den Bereich D1:D13 nach F1:F13 (Abbildung 5.6).

Danach verschieben Sie E1:E13 nach D1:D13 und im letzten Schritt F1:F13 nach E1:E13 (Abbildung 5.7).



	A	B	C	D	E
1	Einkaufskosten	Lagermiete	Werbekosten	Büromiete	Büromat./-möbel
2	107822	1047	8560	1200	834
3	95927	1047	7600	1200	302
4	97264	1080	9304	1200	280
5	122846	1080	8560	1200	150
6	103846	1080	7890	1320	274
7	114963	1080	10845	1320	720
8	118370	1200	9570	1320	549
9	130827	1200	13065	1320	839
10	133840	1200	12085	1450	224
11	128746	1200	18563	1450	172
12	163790	1800	19750	1450	642
13	174972	1800	21510	2700	830
14					

Abbildung 5.7:
Edit.xls, »Tabelle3«

5.6 Einfügen, Entfernen und Löschen

Benötigen Sie Platz für Überschriften oder Ähnliches, müssen Sie nicht unbedingt Bereiche verschieben; meist genügt es, an der betreffenden Stelle eine einzelne Zelle oder einen Bereich aus leeren Zellen einzufügen.

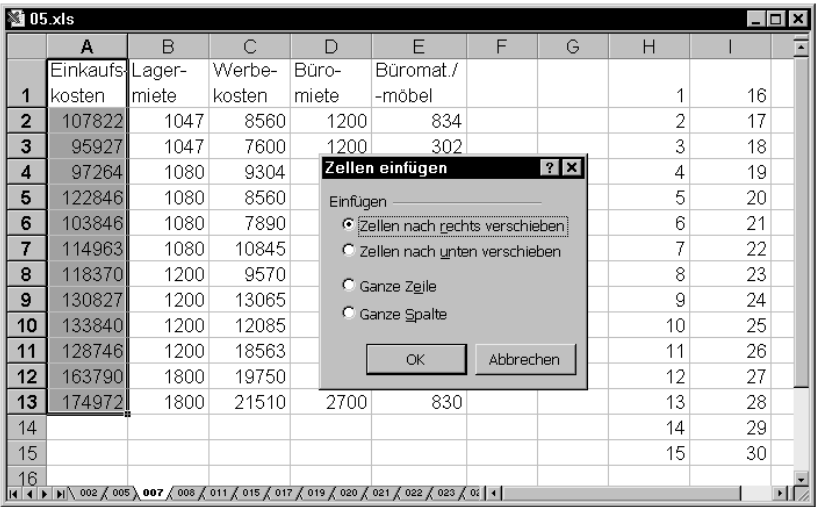
Einfügen

Angenommen, Sie wollen in Spalte A Monatsnamen wie »Jan« und »Feb« einfügen. Statt den Bereich A1:E13 nach B1:F13 zu verschieben, fügen Sie einfach mit EINFÜGEN|ZELLEN... bzw. dem abgebildeten Symbol einen Bereich aus Leerzellen ein, der groß genug ist, um die Zeilenüberschriften aufzunehmen (Abbildung 5.8).

Selektieren Sie entsprechend der Abbildung jenen Bereich, in den die Leerzellen eingefügt werden sollen (im Beispiel A1:A13), und klicken Sie das abgebildete Symbol an oder wählen Sie EINFÜGEN|ZELLEN... .



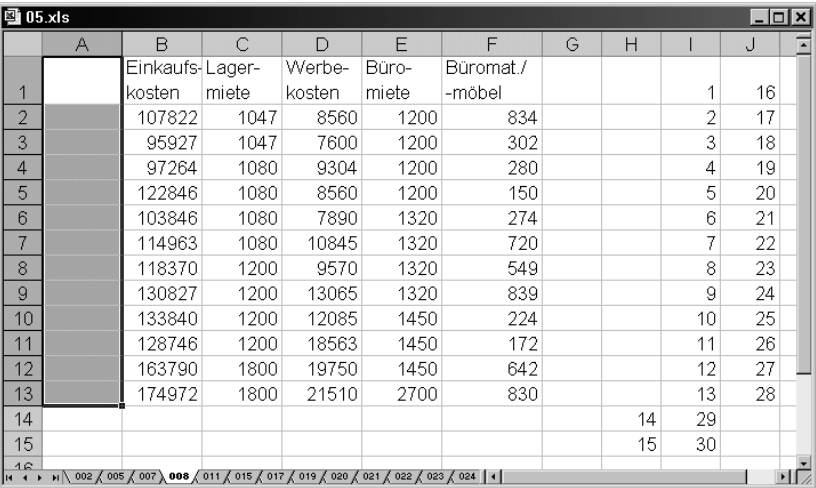
Abbildung 5.8:
Einfügen
vorbereiten



Dieser Befehl fügt einen Bereich aus Leerzellen ein, der in Form und Größe dem momentan selektierten Bereich entspricht und verschiebt diesen zuvor in die anzugebende Richtung, ebenso wie alle in dieser Richtung daran angrenzenden Zellen.

Die Verschieberichtung geben Sie im Dialogfeld an, wobei normalerweise bereits die passende Richtung vorgegeben ist. Bestätigen Sie die vorgegebene Option *Zellen nach rechts verschieben* (Abbildung 5.9).

Abbildung 5.9:
Nach dem Einfügen



Im Bereich A1:A13 wurden wie gewünscht Leerzellen eingefügt und dabei sowohl die zuvor in diesem Bereich enthaltenen als auch die rechts angrenzenden Zellen (siehe Spalten H und I) nach rechts verschoben.

Hätten Sie *Zellen nach unten verschieben* gewählt, wären die in A1:A13 enthaltenen Zellen und alle Zellen darunter entsprechend nach unten statt nach rechts verschoben worden.

Wie Sie sehen, besitzt das Einfügen von Zellen im Gegensatz zum Verschieben von Bereichen Auswirkungen auf die gesamte Tabelle, die anschließend sehr unschön aussehen kann, da alle in der Verschieberichtung angrenzenden Zellen verschoben werden.

Sie können den eingefügten Zellbereich jedoch problemlos wieder entfernen, indem Sie ihn selektieren, also A1:A13 und BEARBEITEN|ZELLEN LÖSCHEN... wählen oder auf das abgebildete Symbol klicken (Abbildung 5.10).

Entfernen

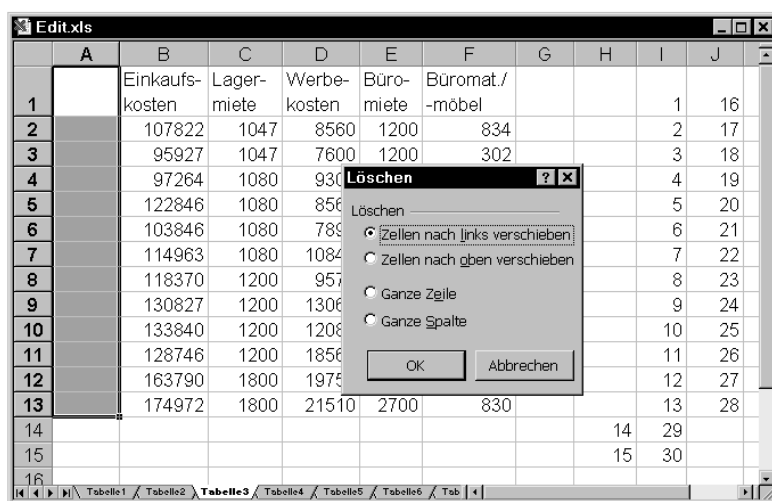


Abbildung 5.10:
Bereiche entfernen

Excel will wissen, in welche Richtung angrenzende Zellen nach dem Entfernen des Bereichs verschoben werden sollen.

Zellen nach links verschieben stellt den Zustand vor dem Einfügen des Leerbereichs wieder her, da der Bereich A1:A13 entfernt und alle Zellen rechts davon um eine Spalte nach links verschoben werden.

BEARBEITEN|ZELLEN LÖSCHEN... entfernt einen Zellbereich und beeinflusst dabei immer die gesamte Tabelle.

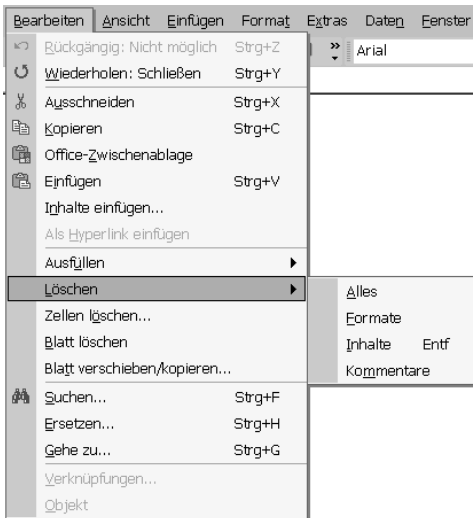
Inhalte löschen

Entfernen Sie beispielsweise den Bereich H1:J15, der nur benötigt wurde, um die Beeinflussung der restlichen Tabelle beim Einfügen und Löschen zu demonstrieren, und wählen Sie dabei als Verschieberichtung *Zellen nach links verschieben*, werden alle Zellen rechts vom zu entfernenden Bereich in diese Richtung verschoben.

Zugegeben, weiter rechts befindet sich nichts – allerdings nur in unserer Beispiel-Tabelle! In Ihrer Buchführungs- oder sonstigen Tabelle könnte das anders sein.

Wollen Sie sichergehen und jegliche ungewollte Beeinflussung der restlichen Tabelle vermeiden, entfernen Sie die Zellen nicht komplett, sondern selektieren den Bereich H1:J15 und wählen danach einen der Unterbefehle des Befehls BEARBEITEN|LÖSCHEN, um nur die *Inhalte* der betreffenden Zellen zu löschen (Abbildung 5.11).

Abbildung 5.11:
Ausgewählte Inhalte
löschen



ALLES löscht alles: den Inhalt der Zelle, eventuell eingestellte Formatierungen, zugehörige Kommentare und Hyperlinks etc.



FORMATE löscht nur Formatierungen.



INHALTE löscht den Inhalt (eine Zahl, einen Text oder eine Formel), belässt jedoch die Formatierung, sodass danach eingegebene Zahlen wieder genauso formatiert werden wie zuvor.

KOMMENTARE löscht nur Kommentare, die zur betreffenden Zelle gehören.

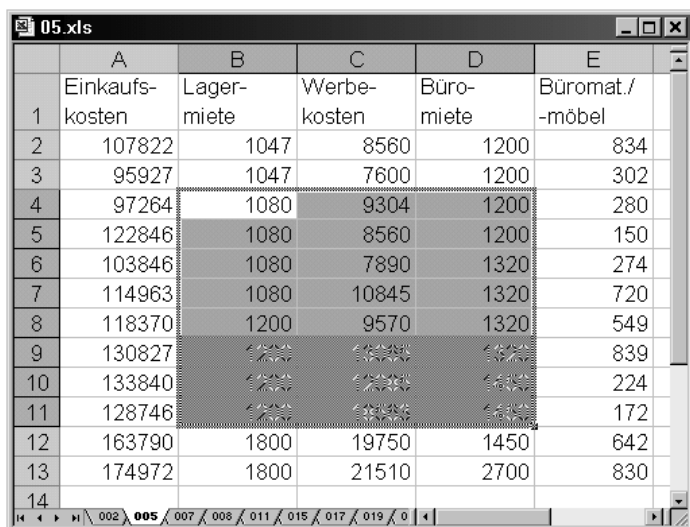
Wählen Sie nun bitte ALLES oder INHALTE, um die Zahlen zu löschen oder drücken Sie einfach **[Entf]**, was der Option INHALTE entspricht.



Mit der Maus können Sie Bereiche oder ihre Inhalte auch per Drag&Drop löschen bzw. einfügen: Zunächst selektieren Sie den betreffenden Bereich und lassen die Maustaste wieder los.

Danach ziehen Sie das kleine Selektionskästchen in der rechten unteren Bereichsecke (der Cursor wird an der korrekten Position zu einem Fadenkreuz) bei gedrückter Maustaste nach oben in den Bereich hinein.

Alle dabei überstrichenen Zellen werden während des Ziehens hellgrau dargestellt (Abbildung 5.12).

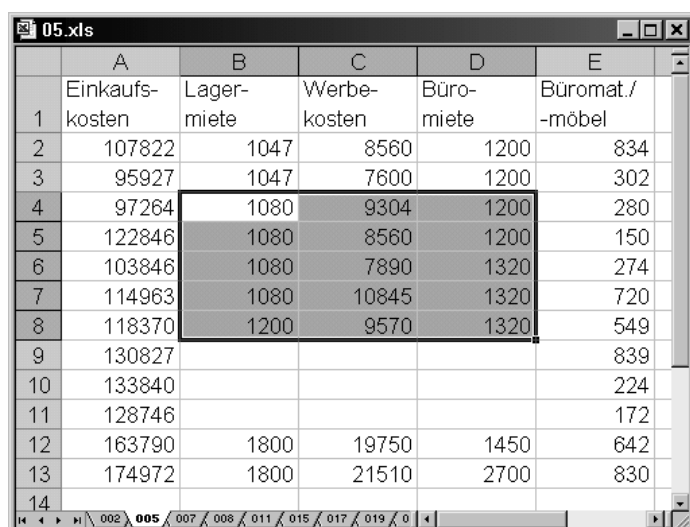


	A	B	C	D	E
1	Einkaufskosten	Lagermiete	Werbekosten	Büromiete	Büromat./-möbel
2	107822	1047	8560	1200	834
3	95927	1047	7600	1200	302
4	97264	1080	9304	1200	280
5	122846	1080	8560	1200	150
6	103846	1080	7890	1320	274
7	114963	1080	10845	1320	720
8	118370	1200	9570	1320	549
9	130827				839
10	133840				224
11	128746				172
12	163790	1800	19750	1450	642
13	174972	1800	21510	2700	830
14					

Abbildung 5.12:
Bereichsteilmenge
selektieren

Lassen Sie die Maustaste los, werden die Inhalte der hellgrau dargestellten Zellen gelöscht (Abbildung 5.13).

Drag&Clear



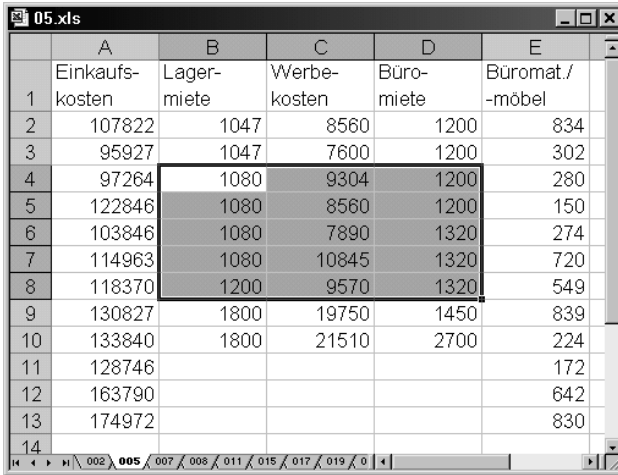
	A	B	C	D	E
1	Einkaufskosten	Lagermiete	Werbekosten	Büromiete	Büromat./-möbel
2	107822	1047	8560	1200	834
3	95927	1047	7600	1200	302
4	97264	1080	9304	1200	280
5	122846	1080	8560	1200	150
6	103846	1080	7890	1320	274
7	114963	1080	10845	1320	720
8	118370	1200	9570	1320	549
9	130827				839
10	133840				224
11	128746				172
12	163790	1800	19750	1450	642
13	174972	1800	21510	2700	830
14					

Abbildung 5.13:
Inhalt der Teilmenge löschen

Drag&Delete

Halten Sie beim Ziehen des Selektionskästchens die -Taste gedrückt (der Cursor ändert dabei seine Form) und lassen Sie die Maustaste los, werden stattdessen die grau dargestellten Zellen entfernt (Abbildung 5.14).

Abbildung 5.14:
Teilmenge
entfernen



	A	B	C	D	E
1	Einkaufskosten	Lagermiete	Werbekosten	Büromiete	Büromat./-möbel
2	107822	1047	8560	1200	834
3	95927	1047	7600	1200	302
4	97264	1080	9304	1200	280
5	122846	1080	8560	1200	150
6	103846	1080	7890	1320	274
7	114963	1080	10845	1320	720
8	118370	1200	9570	1320	549
9	130827	1800	19750	1450	839
10	133840	1800	21510	2700	224
11	128746				172
12	163790				642
13	174972				830
14					

Um den Bereich B4:D11 komplett zu entfernen oder seine Inhalte zu löschen, hätten Sie das Selektionskästchen nur weit genug nach oben ziehen müssen, bis der gesamte Bereich hellgrau dargestellt wird.

Drag&Insert

Um genau umgekehrt Zellen einzufügen, ziehen Sie das Selektionskästchen nicht in den selektierten Bereich hinein, sondern gerade umgekehrt heraus und lassen die Maustaste los, während Sie weiterhin die -Taste drücken (Abbildung 5.15).

Um eine einzelne Zelle zu entfernen oder ihren Inhalt zu löschen, klicken Sie sie an, lassen die Maustaste los, klicken das Kästchen an und ziehen es in die Zelle hinein, die daraufhin grau hervorgehoben und entfernt bzw. deren Inhalt gelöscht wird.

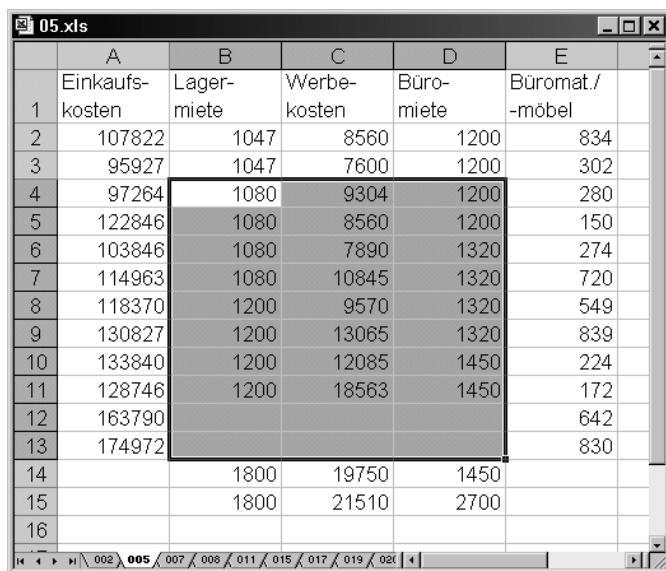
Analog dazu können Sie ausgehend von einer einzelnen Zelle das zugehörige Kästchen bei gedrückter -Taste aus der Zelle herausziehen, um unterhalb, oberhalb, rechts oder links von ihr Leerzellen einzufügen.

Drag&Copy

Halten Sie die -Taste nicht gedrückt, wird nach dem Loslassen der Maustaste der Inhalt der betreffenden Zelle in die nun markierten Zellen kopiert.



Nähere Informationen dazu – und zum AutoAusfüllen-Smart Tag, das nach dem Loslassen der Maustaste erscheint – finden Sie im Kapitel 5.10.1, »AutoAusfüllen«.



	A	B	C	D	E
1	Einkaufs-	Lager-	Werbe-	Büro-	Büromat./
2	kosten	miete	kosten	miete	-möbel
3	107822	1047	8560	1200	834
4	95927	1047	7600	1200	302
5	97264	1080	9304	1200	280
6	122846	1080	8560	1200	150
7	103846	1080	7890	1320	274
8	114963	1080	10845	1320	720
9	118370	1200	9570	1320	549
10	130827	1200	13065	1320	839
11	133840	1200	12085	1450	224
12	128746	1200	18563	1450	172
13	163790				642
14	174972				830
15		1800	19750	1450	
16		1800	21510	2700	

Abbildung 5.15:
Zellen einfügen

5.7 Spalten und Zeilen behandeln

Um das Problem der in Spalte A einzufügenden Zeilenüberschriften ästhetisch zu lösen (im Gegensatz zu Bild 5.9), können Sie entweder den gesamten in der Tabelle benutzten Bereich A1:I15 nach rechts verschieben oder aber einen von A1 bis A15 reichenden Leerbereich einfügen.

Alternativ dazu können Sie ganz links nicht nur einen Bereich, sondern eine komplette Spalte einfügen.

Selektieren Sie dazu die Spalte wie in Kapitel 4.3, »Spalten, Zeilen und das Arbeitsblatt« beschrieben und wählen Sie EINFÜGEN|SPALTEN bzw. klicken Sie auf das abgebildete Symbol (Abbildung 5.16).



Nun können Sie die gewünschten Zeilenüberschriften eingeben (Abbildung 5.17).

Sie können nicht nur eine, sondern mehrere Spalten selektieren und einen entsprechend großen Spaltenbereich und analog dazu mit EINFÜGEN|ZEILEN eine oder mehrere Zeilen auf einmal einfügen.

Die Angabe der Verschieberichtung erübrigt sich in beiden Fällen: Wird eine komplette Spalte eingefügt, ist für Excel klar, dass alle daran angrenzenden Spalten nach rechts verschoben werden müssen; wird eine Zeile eingefügt, müssen alle Zeilen darunter nach unten verschoben werden.

Abbildung 5.16:
Spalte einfügen

	A	B	C	D	E	F
1		Einkaufskosten	Lagermiete	Werbekosten	Büromiete	Büromat./-möbel
2		107822	1047	8560	1200	834
3		95927	1047	7600	1200	302
4		97264	1080	9304	1200	280
5		122846	1080	8560	1200	150
6		103846	1080	7890	1320	274
7		114963	1080	10845	1320	720
8		118370	1200	9570	1320	549
9		130827	1200	13065	1320	839
10		133840	1200	12085	1450	224
11		128746	1200	18563	1450	172
12		163790	1800	19750	1450	642
13		174972	1800	21510	2700	830
14						

Abbildung 5.17:
Edit.xls, »Tabelle4«

	A	B	C	D	E	F
1		Einkaufskosten	Lagermiete	Werbekosten	Büromiete	Büromat./-möbel
2	Jan	107822	1047	8560	1200	834
3	Feb	95927	1047	7600	1200	302
4	Mae	97264	1080	9304	1200	280
5	Apr	122846	1080	8560	1200	150
6	Mai	103846	1080	7890	1320	274
7	Jun	114963	1080	10845	1320	720
8	Jul	118370	1200	9570	1320	549
9	Aug	130827	1200	13065	1320	839
10	Sep	133840	1200	12085	1450	224
11	Okt	128746	1200	18563	1450	172
12	Nov	163790	1800	19750	1450	642
13	Dez	174972	1800	21510	2700	830
14						



Um beispielsweise zwischen dem ersten und dem zweiten Quartal eine Leerzeile einzufügen, selektieren Sie Zeile 5 und wählen EINFÜGEN|ZEILEN oder klicken auf das abgebildete Symbol.

Fügen Sie anschließend auch zwischen den restlichen Quartalen je eine Leerzeile ein, so ergibt sich folgendes Bild (Abbildung 5.18).



Sind komplette Spalten selektiert, entfernt BEARBEITEN|ZELLEN LÖSCHEN... diese Spalten (einfacher: Anklicken des abgebildeten Symbols).



Sind komplette Zeilen selektiert, entfernt BEARBEITEN|ZELLEN LÖSCHEN... diese Zeilen (einfacher: Anklicken des abgebildeten Symbols).

	A	B	C	D	E	F	G
1		Einkaufskosten	Lagermiete	Werbekosten	Büromiete	Büromat./-möbel	
2	Jan	107822	1047	8560	1200	834	
3	Feb	95927	1047	7600	1200	302	
4	Mae	97264	1080	9304	1200	280	
5							
6	Apr	122846	1080	8560	1200	150	
7	Mai	103846	1080	7890	1320	274	
8	Jun	114963	1080	10845	1320	720	
9							
10	Jul	118370	1200	9570	1320	549	
11	Aug	130827	1200	13065	1320	839	
12	Sep	133840	1200	12085	1450	224	
13							
14	Okt	128746	1200	18563	1450	172	
15	Nov	163790	1800	19750	1450	642	
16	Dez	174972	1800	21510	2700	830	
17							

Abbildung 5.18:
Edit.xls, »Tabelle5«

5.8 Kopieren

BEARBEITEN|KOPIEREN kopiert Zellinhalte und kann mit beliebigen rechteckigen Bereichen umgehen (einfacher: das abgebildete Symbol der Standardleiste anklicken oder einfach **[Strg] + [C]** drücken).



Um beispielsweise die Monatsnamen, die sich in Spalte A befinden, in Spalte G zu kopieren, selektieren Sie A1:A16 und wählen KOPIEREN, klicken danach auf G1, die linke obere Ecke des Zielbereichs, und wählen BEARBEITEN|EINFÜGEN oder drücken einfach **[↵]** (Alternativen: entsprechendes Symbol der Standardleiste oder Kurztaste **[Strg] + [V]**).



Die Monatsnamen befinden sich nun auf beiden Seiten der Zahlenkolonnen (Abbildung 5.19).

Nur wenn Sie die Monatsnamen mit BEARBEITEN|EINFÜGEN einfügen, erscheint danach das abgebildete Einfügen-Smart Tag. Es enthält einige beim Kopieren und Einfügen häufig verwendete Optionen, die großteils verschiedenen Optionen des Befehls BEARBEITEN|INHALTE EINFÜGEN... entsprechen (siehe Kapitel 5.9, »Inhalte einfügen«).



Zahlenkolonnen möchte man meist addieren. Also geben wir unter jeder Zahlenreihe eine Formel ein, die die Summe dieses Postens über alle zwölf Monate bildet.

Abbildung 5.19:
Edit.xls, »Tabelle6«

	A	B	C	D	E	F	G	H
1		Einkaufs-	Lager-	Werbe-	Büro-	Büromat./		
2	Jan	107822	1047	8560	1200	834	Jan	
3	Feb	95927	1047	7600	1200	302	Feb	
4	Mae	97264	1080	9304	1200	280	Mae	
5								
6	Apr	122846	1080	8560	1200	150	Apr	
7	Mai	103846	1080	7890	1320	274	Mai	
8	Jun	114963	1080	10845	1320	720	Jun	
9								
10	Jul	118370	1200	9570	1320	549	Jul	
11	Aug	130827	1200	13065	1320	839	Aug	
12	Sep	133840	1200	12085	1450	224	Sep	
13								
14	Okt	128746	1200	18563	1450	172	Okt	
15	Nov	163790	1800	19750	1450	642	Nov	
16	Dez	174972	1800	21510	2700	830	Dez	
17								
18								

☒ Alle Formate der Ursprungszellen beibehalten
☐ Formatierung der Zielzellen übernehmen
☐ Werte und Zahlformate
☐ Breite der Ursprungsspalte beibehalten
☐ Nur Fgrmatierung
☐ Zellen verknüpfen

Für die erste Zahlenreihe, die – mit Unterbrechungen – von B2 bis B16 reicht, lautet diese Formel:

=B2+B3+B4+B6+B7+B8+B10+B11+B12+B14+B15+B16

Da es keinerlei Auswirkungen hat, überflüssige Nullen zu addieren, können Sie auch die Formel

=SUMME(B2:B16)

benutzen.

Lassen Sie unterhalb des Zahlenbereichs eine Leerzeile frei und geben Sie diese Formel in B18 ein. Oder steuern Sie einfach den Cursor zu B18, klicken das Summierungs-Symbol an und korrigieren den folgenden Bereichsvorschlag durch Selektion von B2:B16.

**Bezugs-
anpassungen**

Die Formel benötigen wir auch in C18, D18, E18 und F18, allerdings nicht exakt die gleiche Formel, die B2 bis B16 addiert, sondern in C18 beispielsweise eine Formel, die stattdessen C2 bis C16 addiert.

Unsere Formel enthält ausschließlich relative Bezüge (Entfernungsangaben), die in etwa besagen: »addiere die Inhalte der zwölf Zellen oberhalb der Formelzelle«. Daher klappt die Anpassung beim Kopieren problemlos, weil die Situation ja in allen fünf Spalten identisch ist: es sollen jeweils die Inhalte der zwölf Zellen oberhalb der Formelzelle addiert werden.

Also können wir unsere Formel problemlos kopieren, statt sie jedesmal neu einzugeben – was bei Verwendung absoluter Bezüge ($=SUMME(\$B\$2:\$B\$16)$) nicht möglich wäre!

Kopieren wir sie nach C18, wird die Formel automatisch angepasst und zu

$=SUMME(C2:C16)$.

Kopieren Sie die Formel bitte nicht separat nach C18, D18, E18 und F18, sondern gehen Sie so vor: Selektieren Sie die Zelle B18, die die Originalformel enthält, und wählen Sie den Befehl KOPIEREN.

**Mehrfaches
Kopieren**

Markieren Sie nun als Ziel den Bereich C18:F18, also alle Zellen, in die diese Formel kopiert werden soll (Abbildung 5.20).

	A	B	C	D	E	F	G
1		Einkaufskosten	Lagermiete	Werbekosten	Büromiete	Büromat./-möbel	
2	Jan	107822	1047	8560	1200	834	Jan
3	Feb	95927	1047	7600	1200	302	Feb
4	Mae	97264	1080	9304	1200	280	Mae
5							
6	Apr	122846	1080	8560	1200	150	Apr
7	Mai	103846	1080	7890	1320	274	Mai
8	Jun	114963	1080	10845	1320	720	Jun
9							
10	Jul	118370	1200	9570	1320	549	Jul
11	Aug	130827	1200	13065	1320	839	Aug
12	Sep	133840	1200	12085	1450	224	Sep
13							
14	Okt	128746	1200	18563	1450	172	Okt
15	Nov	163790	1800	19750	1450	642	Nov
16	Dez	174972	1800	21510	2700	830	Dez
17							
18		1493213					
19							

Abbildung 5.20:
Mehrfaches
Kopieren
vorbereiten

Excel kopiert die Formel in *jede einzelne Zelle des selektierten Zielbereichs* (Abbildung 5.21).

Der Zielbereich darf sogar aus mehreren separaten Bereichen bestehen, die mit dem Hinzufügemodus selektiert werden. Dann wird die Formel in alle Zellen der selektierten Bereiche kopiert.



Sie können nicht nur einzelne Zellen auf diese Weise mehrfach kopieren, sondern beliebig große Ausschnitte selektieren und in größere Einfügebereiche hineinkopieren.

**Bereichsform
und -größe**

Abbildung 5.21:
Edit.xls, »Tabelle7«

	A	B	C	D	E	F	G
1		Einkaufs-	Lager-	Werbe-	Büro-	Büromat./	
2	Jan	107822	1047	8560	1200	834	Jan
3	Feb	95927	1047	7600	1200	302	Feb
4	Mae	97264	1080	9304	1200	280	Mae
5							
6	Apr	122846	1080	8560	1200	150	Apr
7	Mai	103846	1080	7890	1320	274	Mai
8	Jun	114963	1080	10845	1320	720	Jun
9							
10	Jul	118370	1200	9570	1320	549	Jul
11	Aug	130827	1200	13065	1320	839	Aug
12	Sep	133840	1200	12085	1450	224	Sep
13							
14	Okt	128746	1200	18563	1450	172	Okt
15	Nov	163790	1800	19750	1450	642	Nov
16	Dez	174972	1800	21510	2700	830	Dez
17							
18	Summen	1493213	14814	147302	17130	5816	
19							

Voraussetzung für diese Vervielfältigung ist jedoch immer, dass der Einfü-
gungsbereich aus Rechtecken der gleichen Größe und Form wie der
Ursprungsbereich besteht (Abbildung 5.22).

Abbildung 5.22:
Mehrfaches
Kopieren eines
Bereichs

	A	B	C	D	E
1	1	6		1	6
2	2	7		2	7
3	3	8		3	8
4	4	9		4	9
5	5	10		5	10
6				1	6
7				2	7
8				3	8
9				4	9
10				5	10
11					

Für diese Abbildung wurde A1:B5 selektiert, KOPIEREN gewählt, als Zielbe-
reich D1:E10 selektiert und EINFÜGEN gewählt. Excel kopierte die Inhalte
der zehn Zellen nach D1:E5 *und* nach D6:E10 und erstellte somit gleich
zwei Kopien des selektierten Bereichs.



Sie können einen selektierten Bereich nicht nur einmal, sondern beliebig oft nacheinander an unterschiedliche Positionen im Arbeitsblatt kopieren. Voraussetzung dazu ist allerdings, dass Sie den selektierten Bereich an der gewünschten Position nicht mit einfügen, sondern mit dem Befehl EINFÜGEN (Kurtzaste +) oder dem entsprechenden Symbol.

Im Gegensatz zu bleibt der Laufrahmen dabei erhalten, Sie können zur nächsten Einfügeposition gehen und erneut den Befehl EINFÜGEN wählen. (Alternative: Wenn Sie die Symbolleiste *Zwischenablage* einblenden, können Sie den selektierten Bereich ebenfalls beliebig oft durch Klicken auf das betreffende Symbol einfügen.)

Beim Kopieren eines Bereichs mit KOPIEREN und EINFÜGEN werden alle Zellen im Zielbereich überschrieben. Passt Ihnen das nicht, können Sie die betreffenden Zellen zuvor manuell verschieben oder aber statt BEARBEITEN|EINFÜGEN den Befehl EINFÜGEN|KOPIERTE ZELLEN... verwenden.

Kopie einfügen

Angenommen, Sie wollen die Monatsnamen zusätzlich in der Mitte der Tabelle sehen, in Spalte D. Zunächst selektieren Sie den zu kopierenden Bereich A2:A16, der die Namen enthält, wählen BEARBEITEN|KOPIEREN und klicken D2 an, die obere linke Ecke des gewünschten Zielbereichs.

Statt BEARBEITEN|EINFÜGEN wählen Sie nun jedoch den Befehl EINFÜGEN|KOPIERTE ZELLEN..., der EINFÜGEN|ZELLEN... ersetzt hat (Abbildung 5.23).

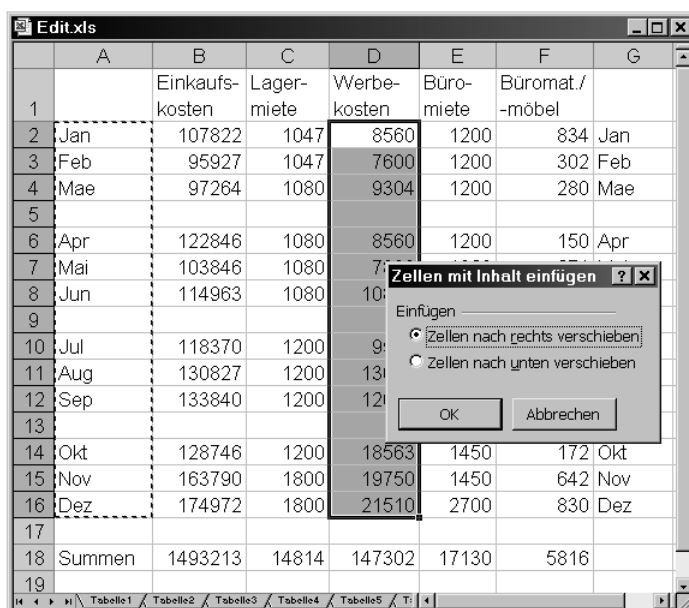


Abbildung 5.23:
Verschieberichtung
angeben

Analog zu diesem Befehl werden Sie nach der Verschieberichtung gefragt. Übernehmen Sie die Vorgabe *Zellen nach rechts verschieben*.

Excel wird zunächst den Bereich D2:D16 und alle rechts angrenzenden Zellen um eine Spalte nach rechts verschieben, um für den zu kopierenden Bereich A2:A16 Platz zu schaffen, und diesen Bereich anschließend einfügen (Abbildung 5.24).

Abbildung 5.24:
Einfügen und
Kopieren

	A	B	C	D	E	F	G	H
1		Einkaufskosten	Lagermiete	Werbekosten	Büromiete	Büromat./-möbel		
2	Jan	107822	1047	Jan	8560	1200	834	Jan
3	Feb	95927	1047	Feb	7600	1200	302	Feb
4	Mae	97264	1080	Mae	9304	1200	280	Mae
5								
6	Apr	122846	1080	Apr	8560	1200	150	Apr
7	Mai	103846	1080	Mai	7890	1320	274	Mai
8	Jun	114963	1080	Jun	10845	1320	720	Jun
9								
10	Jul	118370	1200	Jul	9570	1320	549	Jul
11	Aug	130827	1200	Aug	13065	1320	839	Aug
12	Sep	133840	1200	Sep	12085	1450	224	Sep
13								
14	Okt	128746	1200	Okt	18563	1450	172	Okt
15	Nov	163790	1800	Nov	19750	1450	642	Nov
16	Dez	174972	1800	Dez	21510	2700	830	Dez
17								
18	Summen	1493213	14814	147302	17130	5816		
19								

Wie Sie sehen, haben wir einen kleinen Fehler begangen: Weder die Texte »Werbekosten«, »Büromiete«, »Büromat./-möbel« in D1, E1 und F1 noch die zugehörigen Summenformeln in D18, E18 und F18 wurden verschoben.

Am einfachsten korrigieren Sie das, indem Sie die gesamte Aktion mit BEARBEITEN|RÜCKGÄNGIG:... ungeschehen machen und danach nicht nur A2:A16, sondern den kompletten Bereich A1:A18 kopieren und in D1:D18 einfügen.



Sie können auch komplette Zeilen/Spalten selektieren, KOPIEREN wählen und die Zeilen bzw. Spalten mit BEARBEITEN|EINFÜGEN kopieren oder mit EINFÜGEN|KOPIERTE ZELLEN... an der aktuellen Position einfügen. Allerdings müssen Sie dabei vor dem Einfügen die komplette Zielzeile bzw. Zielspalte markieren!

Kopieren Sie eine Spalte wie A zum Beispiel nach D, müssen Sie vor dem Einfügen die komplette Spalte D als Zielspalte selektieren. Analog dazu müssen Sie beim Kopieren von Zeile 3 nach Zeile 6 vor dem Einfügen die gesamte Zielzeile 6 selektieren.

5.9 Inhalte einfügen

BEARBEITEN|EINFÜGEN fügt an der aktuellen Position einen zuvor ausgeschnittenen oder kopierten Bereich ein.

In der Praxis ist es jedoch häufig notwendig, jene Zellen nicht mit »Allem Drum und Dran« einzufügen, inklusive Formatierungen, Kommentaren etc., sondern selektiv vorzugehen: beispielsweise nur die zuvor kopierten Formeln einzufügen, aber keine Zahlen; oder nur die Kommentare einzufügen, die die zuvor kopierten Zellen enthalten; oder aber die einzufügenden Zahlen gar zu den momentan im Zielbereich enthaltenen Zahlen beim Einfügen addieren zu lassen. Die beiden einfachsten Möglichkeiten sind folgende:

- ➔ Wie bereits erwähnt erscheint nach dem Einfügen mit BEARBEITEN|EINFÜGEN bzw. dem zugehörigen Symbol das Einfügen-Smart Tag, dessen Liste häufig verwendete Einfüge-Optionen enthält. Damit können Sie *nach* dem Einfügen entscheiden, was genau eingefügt werden soll.
- ➔ Sie können auch *vorher* entscheiden, indem Sie nicht wie gewohnt auf das *Einfügen*-Symbol selbst klicken, sondern auf den Listenpfeil am rechten Rand dieses Symbols. Die daraufhin geöffnete Liste enthält ebenfalls häufig verwendete Einfüge-Optionen – und zwar andere als das Einfügen-Smart Tag!

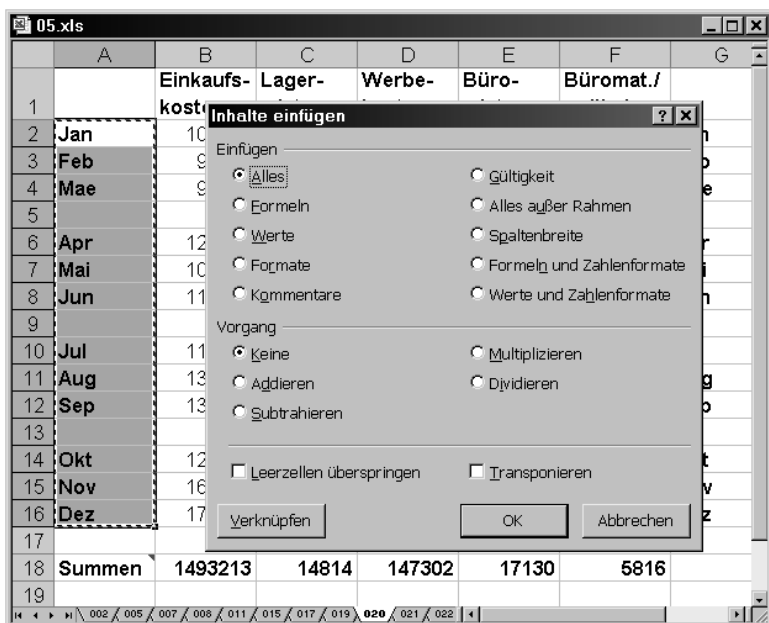
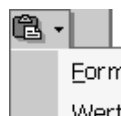
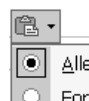


Abbildung 5.25:
Formatierungen und
Kommentare

Beide Techniken stellen Ihnen jedoch nur einen Bruchteil aller vorhandenen Einfüge-Optionen zur Verfügung, eben die am häufigsten verwendeten. Alle Optionen stellt nur der Befehl BEARBEITEN|INHALTE EINFÜGEN... zur Verfügung (Abbildung 5.25).

5.9.1 Inhalte auswählen

Die in Bild 5.25 gezeigte Tabelle entstand aus dem Blatt »Tabelle7« der Demoarbeitsmappe EDIT.XLS: Ich selektierte die Bereiche, die die Texte und Summenformeln enthalten, und klickte auf das Symbol »F« der Formatleiste, um der Zelle das Zellformat »Fett« zuzuweisen. Zusätzlich speicherte ich zu Zelle A18 einen Zellkommentar.

Enthält eine Tabelle wie diese eine bunte Mischung aus Werten, Formeln, Formatierungen und Kommentaren, so können Sie Teile daraus ausschneiden oder kopieren, die linke obere Ecke des Zielbereichs selektieren und mit BEARBEITEN|INHALTE EINFÜGEN... nur die gewünschten Zellbestandteile einfügen. Dabei öffnet sich das in der letzten Abbildung gezeigte Dialogfeld:

- ➔ *Alles* fügt alles ein und entspricht damit dem normalen Einfügen.
- ➔ *Formeln* fügt nur Formeln, Werte und Texte ein, aber weder Formate noch Kommentare. Im Beispiel werden die Formeln, Werte und Texte eingefügt, aber weder die Fettschrift, noch eine Formatierung oder Zellkommentar.

Ist der Zielbereich beispielsweise kursiv formatiert, bleibt diese Formatierung erhalten und wird nicht wie beim normalen Einfügen durch die fette Formatierung der zu kopierenden Zellen ersetzt.



- ➔ *Werte* kopiert nur die *angezeigten* Werte oder Texte, aber weder Formeln noch Kommentare oder Formate. Im Zielbereich werden daher statt der Formeln deren aktuelle Resultate eingefügt.

Enthält zum Beispiel B6 die Formel $=B3+B4+B5$ und ergibt diese Formel momentan das Ergebnis 90, wird in die zugehörige Zelle des Zielbereichs nur dieses Ergebnis eingefügt; also der Wert 90 und nicht die Formel selbst.

Alternativ dazu können Sie die Taste $\boxed{F9}$ verwenden. Ist der Bearbeitungsmodus aktiv ($\boxed{F2}$), ersetzt diese Taste den Inhalt der Bearbeitungsleiste (oder einen darin selektierten Ausdruck), also auch eine komplette Formel, durch ihren Wert.



- ➔ *Formate* kopiert nur die Formatierungen der einzelnen Zellen, aber weder deren Inhalte (beispielsweise Werte oder Formeln) noch zugehörige Kommentare.

- ➔ *Kommentare* kopiert nur Zellkommentare, aber weder Formeln noch Werte oder Formate.
- ➔ *Alles außer Rahmen* fügt außer Zellinhalten auch Formatierungen ein, wobei Zellrahmen jedoch nicht eingefügt werden.
- ➔ *Spaltenbreite* kopiert nur die Spaltenbreiten der kopierten Zellen, sonst nichts.
- ➔ *Formeln und Zahlenformate* fügt nur Formeln (und Werte und Texte) und das Zahlenformat des Quellbereichs ein.
- ➔ *Werte und Zahlenformate* fügt nur Zahlen und das Zahlenformat ein.
- ➔ *Gültigkeit* fügt nur die Gültigkeitsregeln ein (siehe Kapitel 38.4, »Gültigkeitsregeln benutzen«).

5.9.2 Leerzellen überspringen

Normalerweise überträgt Excel beim Einfügen auch leere Zellen: Selbst wenn eine einzufügende Zelle nichts enthält, wird die zugehörige Zelle des Zielbereichs überschrieben und enthält anschließend ebenfalls nichts mehr.

Genau das können Sie jedoch mit der Option *Leertzellen überspringen* verhindern: Sind Zellen des einzufügenden Bereichs leer, wird der Inhalt der zugehörigen Zielbereichszellen erhalten.

5.9.3 Transponieren

Bei der Einfüge-Option *Transponieren* verändert Excel die Anordnung der einzufügenden Daten, so als wären die beiden Achsen des Koordinatensystems miteinander vertauscht worden.

Die folgende Abbildung enthält einen Zahlenbereich B2:C6, der mit aktivierter Option *Transponieren* kopiert wurde, wobei ich als linke obere Ecke des Zielbereichs B9 wählte (Abbildung 5.26).

Zuvor untereinander angeordnete Werte befinden sich anschließend nebeneinander und umgekehrt: Die ursprünglich in der ersten Spalte des Ausgangsbereichs untereinander angeordneten Zahlen werden in der obersten Zeile des Zielbereichs nebeneinander angeordnet; die zweite Spalte von links wird zur zweiten Zeile von oben; die dritte Spalte von links wird zur dritten Zeile von oben und so weiter.

Transponieren können Sie verwenden, um aus unserem Demo-Arbeitsblatt auf einfachste Art und Weise eine Kopie zu erzeugen, in der nicht die Monate untereinander angeordnet sind, sondern die Kostenarten. Dafür befinden sich nun die Monate in dieser Kopie nebeneinander.

Abbildung 5.26:
Transponieren
einer Tabelle

	A	B	C	D	E	F
1						
2		1	6			
3		2	7			
4		3	8			
5		4	9			
6		5	10			
7						
8						
9		1	2	3	4	5
10		6	7	8	9	10
11						

5.9.4 Rechenoperationen

Unter *Vorgang* können Sie bestimmen, wie die im Quellbereich enthaltenen Werte beim Einfügen in den Zielbereich mit eventuell bereits darin enthaltenen Werten verknüpft werden sollen (Abbildung 5.27).

Abbildung 5.27:
Vor der arithmetischen
Verknüpfung

	A	B	C	D	E	F
1						
2		1	6		11	16
3		2	7		12	17
4		3	8		13	18
5		4	9		14	19
6		5	10		15	20
7						

Inhalte einfügen

Einfügen

- ☒ Alles
- ☐ Formeln
- ☐ Werte
- ☐ Formate
- ☐ Kommentare
- ☐ Gültigkeit
- ☐ Alles außer Rahmen
- ☐ Spaltenbreite
- ☐ Formeln und Zahlenformate
- ☐ Werte und Zahlenformate

Vorgang

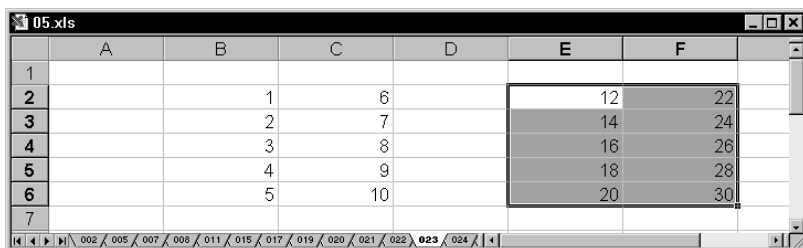
- ☐ Keine
- ☒ Addieren
- ☐ Subtrahieren
- ☐ Multiplizieren
- ☐ Dividieren

☐ Leerzellen überspringen ☐ Transponieren

Verknüpfen OK Abbrechen

In dieser Abbildung enthalten die Bereiche B2:C6 und E2:F6 jeweils zehn Zahlen. Selektieren Sie B2:C6 und wählen Sie KOPIEREN. Gehen Sie zu E2, wählen Sie INHALTE EINFÜGEN... und aktivieren Sie die Optionen *Alles* und *Addieren*.

Alle Zellattribute werden eingefügt und die Zellen des Ausgangsbereichs mit denen des Zielbereichs durch eine Addition verknüpft (Abbildung 5.28).



	A	B	C	D	E	F
1						
2		1	6		12	22
3		2	7		14	24
4		3	8		16	26
5		4	9		18	28
6		5	10		20	30
7						

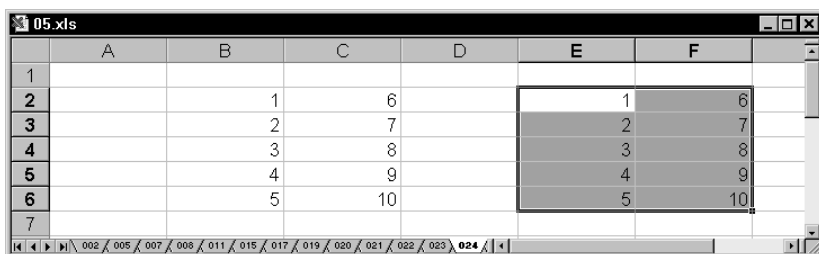
Abbildung 5.28:

Nach dem
Verknüpfen

Das Resultat: Die 1 in B2 und die 11 in E2 werden addiert und als neuer Inhalt von E2 ergibt sich 12. Entsprechend werden B3 und E3 addiert, sodass E3 danach den Wert 14 enthält etc.

5.9.5 Verknüpfen

Die Schaltfläche *Verknüpfen* ist nützlich, wenn Sie Zahlenkolonnen kopieren und daran interessiert sind, dass bei jeder Änderung der Originaldaten die Kopien dieser Daten aktualisiert werden (Abbildung 5.29).



	A	B	C	D	E	F
1						
2		1	6		1	6
3		2	7		2	7
4		3	8		3	8
5		4	9		4	9
6		5	10		5	10
7						

Abbildung 5.29:

Verknüpfte Bereiche

Der selektierte Bereich E2:F6 ist nicht durch einfaches Kopieren entstanden. Zwar wurde wie beim Kopieren B2:C6 selektiert, KOPIEREN gewählt und als linke obere Ecke des Zielbereichs E2 selektiert. Statt EINFÜGEN wählte ich jedoch INHALTE EINFÜGEN... und aktivierte *Verknüpfen*.

Das Resultat ist eine Kopie, die mit dem Ursprungsbereich verknüpft ist, da Excel in E2 die Formel $=B2$ speichert, die bewirkt, dass darin der gleiche Wert wie in B2 angezeigt wird.

In E3 trägt Excel analog dazu die Formel $=B3$ ein, in E4 die Formel $=B4$ und so weiter.

Ändern Sie einen Wert in diesem Bereich, d.h. geben Sie beispielsweise in B3 die Zahl 63 ein, wird die korrespondierende Zelle E3 des Zielbereichs sofort entsprechend aktualisiert (Abbildung 5.30).

Abbildung 5.30:
Aktualisierung

	A	B	C	D	E	F
1	1	1	6		1	6
2		63	7		63	7
3	3	3	8		3	8
4	4	4	9		4	9
5	5	5	10		5	10
6						
7						

5.10 Füllen

5.10.1 AutoAusfüllen

In der rechten unteren Ecke des Rahmens, der die aktive Zelle beziehungsweise einen selektierten Bereich umgibt, befindet sich das bereits erwähnte kleine Selektionskästchen. Mit ihm beziehungsweise mit dem Befehl BEARBEITEN|AUSFÜLLEN können Sie einen Bereich sehr einfach mit bestimmten Werten ausfüllen und dabei beispielsweise eine Zahlen- oder Datumsreihe erzeugen.

Nehmen wir an, Sie haben ein Arbeitsblatt mit genau einer Zahlenkolonne, die von A1 bis A10 reicht. Sie wollen diese Zahlenreihe in die benachbarten Spalten B, C und D kopieren.

Statt die Technik des Kopierens und Einfügens zu verwenden, selektieren Sie zunächst den zu kopierenden Bereich A1:A10. Nach dem Loslassen der Maustaste bewegen Sie den Mauscursor zu dem schwarzen Kästchen in der unteren rechten Ecke des Selektionsrahmens. Er ändert seine Form und wird zu einem Fadenkreuz.

Sie klicken dieses Kästchen nun mit der linken Maustaste an. Ziehen Sie daran (bei weiterhin gedrückter Maustaste), so erscheint ein schraffierter Selektionsrahmen, den Sie im Beispiel nach rechts ziehen zu Spalte D (Abbildung 5.31).

Nach dem Loslassen der Maustaste werden die zusätzlich selektierten Spalten B, C und D mit dem Inhalt von Spalte A ausgefüllt (Abbildung 5.32).



Zusätzlich erscheint das AutoAusfüllen-Smart Tag. Das zugehörige Kontextmenü zeigt durch die aktivierte Option aus, welche Aktion soeben ausgeführt wurde (Füllen), ermöglicht Ihnen jedoch durch Wählen einer der anderen Optionen, diese rückgängig zu machen und stattdessen eine der Varianten zum Berechnen von Reihen auszuführen, die ich im Kapitel 5.10.2, »Reihen berechnen«, erläutere.

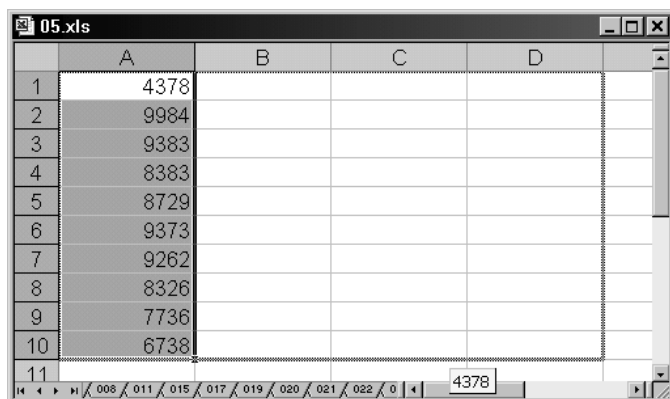


Abbildung 5.31:

Selektion des auszufüllenden Bereichs mit der Maus

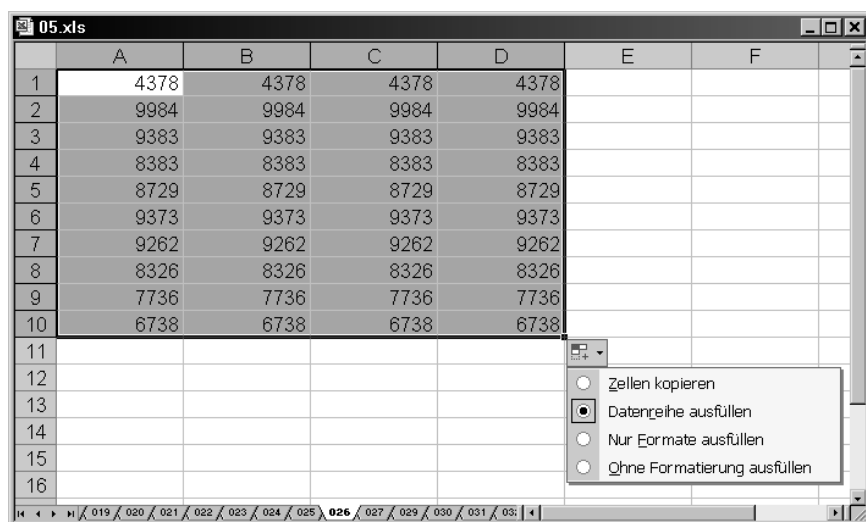


Abbildung 5.32:

Den Bereich rechts ausfüllen

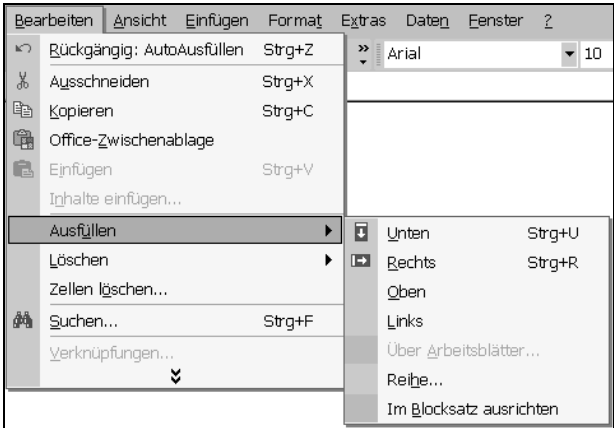
Übrigens hätten Sie den selektierten Bereich ebenso gut nach unten verlängern können, um die in A1:A10 enthaltenen Werte beispielsweise nach A11:A20 zu kopieren. Oder – bei Bereichen, die sich nicht wie dieser in der linken oberen Tabellenecke befinden – nach links oder oben.

Mit der Tastatur selektieren Sie einen Bereich, der sowohl Quellbereich als auch auszufüllenden Bereich umfasst, im Beispiel A1:D10, und wählen BEARBEITEN|AUSFÜLLEN, um folgendes Untermenü zu öffnen (Abbildung 5.33).



UNTEN kopiert den Inhalt der obersten Bereichszeile in die Zeilen darunter; RECHTS kopiert analog zur vorletzten Abbildung den Inhalt der linken Bereichsspalte in die Spalten rechts davon; OBEN kopiert den Inhalt der untersten Bereichszeile in die Zeilen darüber; LINKS kopiert den Inhalt der rechten Bereichsspalte in die Spalten links davon.

Abbildung 5.33:
Untermenü
von Bear-
beiten/Ausfüllen



Das Ausfüllen lässt sich auch auf mehrere im Hinzufügemodus selektierte Bereiche gleichzeitig anwenden. Dann wird beispielsweise die oberste Zeile *jedes* dieser Bereiche in die darunter liegende kopiert.

Zwei spezielle Symbole der Kategorie *Bearbeiten* des *Anpassen*-Dialogfelds erleichtern das Ausfüllen von Bereichen:



Kopiert die linke Spalte des selektierten Bereichs in die restlichen Bereichsspalten (BEARBEITEN|AUSFÜLLEN|RECHTS)

Halten Sie beim Anklicken die -Taste gedrückt, so wird genau umgekehrt die rechte Spalte in die restlichen Bereichsspalten kopiert.



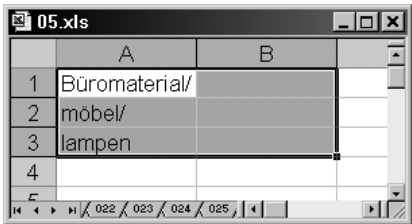
Kopiert die oberste Zeile des selektierten Bereichs in die restlichen Bereichs-zeilen (BEARBEITEN|AUSFÜLLEN|UNTEN)

Halten Sie beim Anklicken die -Taste gedrückt, so wird ebenfalls genau umgekehrt die unterste Zeile in die restlichen Bereichsspalten kopiert.

Aufteilen

Mit BEARBEITEN|AUSFÜLLEN|IM BLOCKSATZ AUSRICHTEN können Sie Texte, die sich in einer Spalte untereinander befinden, kompakter zusammenfassen (Abbildung 5.34).

Abbildung 5.34:
Verstreute Texte



Sie selektieren einen Bereich, der außer den Texten zusätzlich die leeren Zellen umfasst, die zur Zusammenfassung benutzt werden sollen.

Excel benutzt daraufhin zunächst die erste Zeile des Bereichs zur gemeinsamen Darstellung aller Texte und – falls diese nicht ausreichend breit ist – erst danach die folgende Zeile (Abbildung 5.35).

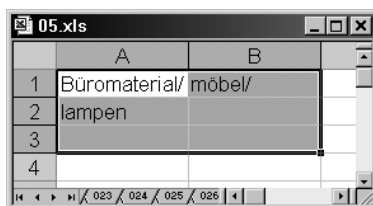


Abbildung 5.35:
Zusammengefasste Texte

Excel verhält sich beim Ausfüllen mit der Maus dank seiner AutoAusfüllen-Funktion sehr clever. Enthält der selektierte Bereich, den Sie durch Ziehen am Selektionskästchen verlängern, eine Wertereihe, wird diese Reihe in die gewünschte Richtung fortgesetzt (Abbildung 5.36).

Datenreihen

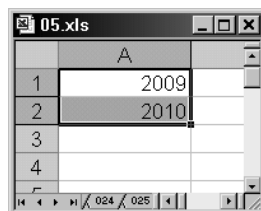


Abbildung 5.36:
Anfang einer Reihe

Die beiden in A1 und A2 enthaltenen Werte 2009 und 2010 interpretiert Excel als Beginn der Reihe 2009, 2010, 2011, 2012 etc.

Wird der Bereich selektiert und per Ziehen am Selektionskästchens nach unten ausgedehnt, ergänzt Excel die Datumsreihe (Abbildung 5.37).

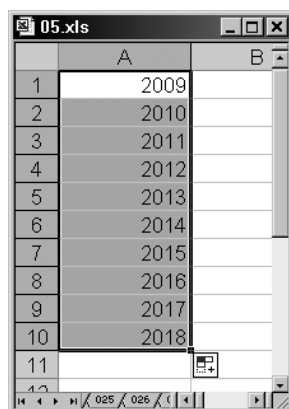


Abbildung 5.37:
Automatisches Ergänzen der Reihe

Das Ganze funktioniert mit beliebigen Reihen, die in der Art

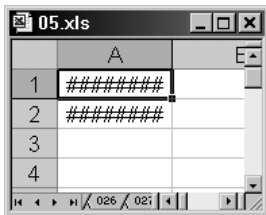
$$X_{n+1} = X + K$$

gebildet werden, bei denen sich also eine Zahl durch Addition einer Konstanten zum Vorgänger ergibt. Die Vorgabe der beiden Zahlen 1 und 3 würde Excel entsprechend zur Zahlenreihe 1, 3, 5, 7... ergänzen und die Vorgabe von 5 und 10 zur Reihe 5, 10, 15, 20

Excel kann echte Datumsreihen bilden, bei denen auch die Monats- und Jahresübergänge berücksichtigt werden: Geben Sie in A1 »27.1.09« ein und ziehen Sie danach am Selektionskästchen nach rechts oder links, so setzt Excel dieses Datum mit »28.1.09«, »29.1.09« etc. fort.

Interessanter ist es allerdings, gleich zwei nicht direkt aufeinander folgende Datumswerte einzugeben, beispielsweise in A1 »27.1.09« und in A2 »29.1.09« (Abbildung 5.38).

Abbildung 5.38:
Unzureichende
Spaltenbreite



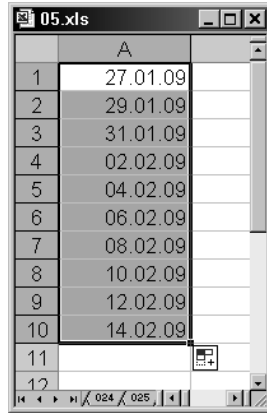
Möglicherweise erscheinen auch bei Ihnen statt der Datumswerte nur eine Reihe von Nummernzeichen »#«. Sie bedeuten, dass die Spaltenbreite zur Anzeige nicht ausreicht.

Klicken Sie in diesem Fall bitte die dünne Linie zwischen den Spaltenbuchstaben A und B an und ziehen Sie die Spalte A bei weiterhin gedrückter Maustaste nach rechts, um sie zu verbreitern: Ist die Spalte breit genug, verwendet Excel die Darstellung »27.1.09« und »29.1.09«, also zweistellige Jahreszahlen.

Excel erkennt, dass diese Eingaben keine Zahlen, sondern je ein Datum darstellen. Ein solches Datum kann Excel selbstständig zu einer fortlaufenden Datumsreihe ergänzen.

Dazu wenden Sie die gleiche Technik an wie zuvor: Sie selektieren die beiden Zellen und lassen die Maustaste los. Danach verlängern Sie die Selektion, indem Sie mit dem Fadenkreuz, zu dem der Cursor an der korrekten Position wird, das Selektionskästchen nach unten ziehen (Abbildung 5.39).

Excel behandelt den Monatsübergang bei der Ergänzung der Reihe absolut korrekt. Das gleiche gilt für Jahresübergänge, sollte diese Reihe lang genug sein, um bis ins folgende Jahr zu reichen.

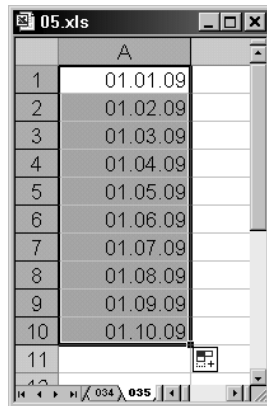


	A
1	27.01.09
2	29.01.09
3	31.01.09
4	02.02.09
5	04.02.09
6	06.02.09
7	08.02.09
8	10.02.09
9	12.02.09
10	14.02.09
11	
12	

Abbildung 5.39:
Tagesreihe erzeugen

Statt Tages- sind auch Monats- oder Jahresreihen möglich. Dass nicht die Tage, sondern eine andere Datumskomponente variiert werden soll, müssen Sie Excel durch die Vorgabe von zumindest zwei Datumswerten klarmachen, sonst erhalten Sie eine Tagesreihe (»27.1.09«, »28.1.09« etc.).

Geben Sie beispielsweise in A1 »1.1.09« ein und in A2 »1.2.09«, so ist klar, dass die Monate fortlaufend variiert und eine Monatsreihe erzeugt werden soll. Wenn Sie diese Reihe von Excel ergänzen lassen, wird sie folgendermaßen fortgesetzt (Abbildung 5.40).



	A
1	01.01.09
2	01.02.09
3	01.03.09
4	01.04.09
5	01.05.09
6	01.06.09
7	01.07.09
8	01.08.09
9	01.09.09
10	01.10.09
11	
12	

Abbildung 5.40:
Monatsreihe

Entsprechend können Sie mit den Vorgaben »1.1.09« und »1.1.10« eine Jahresreihe erzeugen lassen.

Monatsreihen lassen sich auch durch Eingabe eines Monatsnamens erzeugen. Die Vorgabe »Jan« führt zur Fortsetzung mit »Feb«, »Mrz«, »Apr« etc. Die Vorgaben »Jan« und »Mrz« zur Fortsetzung mit »Mai«, »Jul« etc.





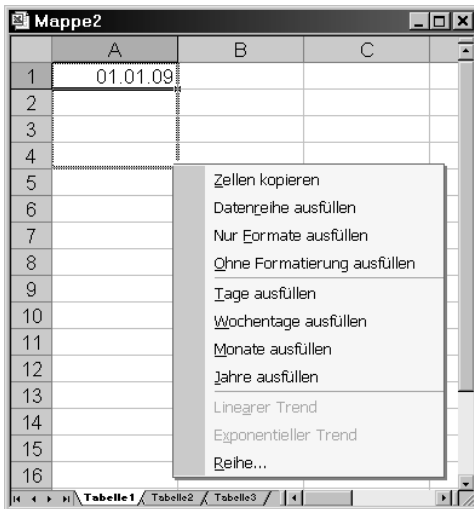
Statt mit vordefinierten Reihen können Sie einen Bereich auch mit eigendefinierten Reihen ausfüllen lassen. Beispielsweise können Sie nach Definition der Reihe »Äpfel«, »Birnen«, »Orangen« und »Kiwis« (siehe Anhang 7, »Excel individuell konfigurieren«, Abschnitt »Benutzerdefinierte Listen«) eine mit »Birnen« begonnene Liste mit »Orangen«, »Kiwis«, »Äpfel«, »Birnen«, »Orangen« ... fortsetzen lassen.

5.10.2 Reihen berechnen



Viele der im Folgenden erläuterten Ausfüll-Optionen stehen Ihnen sofort zur Verfügung, wenn Sie das AutoAusfüllen-Kästchen mit der rechten Maustaste über den auszufüllenden Bereich ziehen. Nach dem Loslassen der Maustaste öffnet sich nämlich folgendes Kontextmenü, das Ihnen direkten Zugriff auf die verschiedenen Ausfüll-Optionen für den selektierten Bereich bietet (Abbildung 5.41).

Abbildung 5.41:
Das AutoAusfüllen-
Kontextmenü



Alternativ dazu können Sie auch die linke Maustaste benutzen. Dann erscheint anschließend das AutoAusfüllen-Smart Tag, dessen Kontextmenü nun nahezu alle Optionen des abgebildeten »echten« Kontextmenüs bietet.

Diese Optionen sind praktisch eine Untermenge jener Optionen, die das Dialogfeld des Befehls BEARBEITEN|AUSFÜLLEN|REIHE... enthält (Abbildung 5.42).

In diesem Dialogfeld können Sie die gewünschte Reihenart näher definieren. Geben Sie dazu in irgendeiner Zelle einen Anfangswert ein, selektieren Sie diese und die benachbarten leeren Zellen, in denen die Reihe ergänzt werden soll und wählen Sie BEARBEITEN|AUSFÜLLEN|REIHE... .

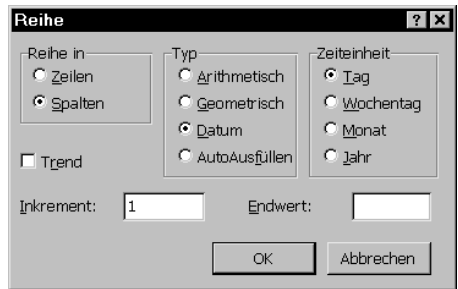


Abbildung 5.42:
Der Befehl
BEARBEITEN
AUSFÜLLEN|REIHE...

Excel verhält sich dabei recht clever und erkennt die grundlegende Reihenart und -position. Selektierten Sie einen spaltenweise angeordneten Bereich, ist unter *Reihe in* automatisch die korrekte Option *Spalten* vorgegeben.

Enthält die obere linke Bereichsecke ein Datum, ist unter *Typ* korrekt *Datum* vorselektiert, sonst *Arithmetisch*.

Die Option *AutoAusfüllen* kennen Sie bereits. Sie entspricht dem interaktiven Verlängern eines Bereichs mit der Maus und veranlasst Excel, die im ausgewählten Bereich enthaltenen Daten zu analysieren, um die gewünschte Reihe selbstständig zu erkennen.

Selektieren Sie beispielsweise A1:A10 und enthält A1 den Wert 1, so ergänzt Excel diese Reihe selbstständig zu 1, 2, 3, ..., 10. Enthält A1 stattdessen einen gültigen Wochentagstext wie »Montag«, ergänzt Excel in A2 »Dienstag«, in A3 »Mittwoch« etc.

Um die einzelnen Datumsreihen-Optionen zu verstehen, sollten Sie sich die folgende Abbildung ansehen, deren Spalten alle mit der Reihenberechnungsfunktion von Excel entstanden (Abbildung 5.43).

Datumsreihen

1	A	B	C	D	E
2	Tagesreihe	Wochentag	Datumsreihen	Jahresreihe	Stop-Reihe
3	01.01.2009	01.01.2009	01.01.2009	01.01.2009	01.01.2009
4	02.01.2009	02.01.2009	01.02.2009	01.01.2010	01.03.2009
5	03.01.2009	05.01.2009	01.03.2009	01.01.2011	01.05.2009
6	04.01.2009	06.01.2009	01.04.2009	01.01.2012	01.07.2009
7	05.01.2009	07.01.2009	01.05.2009	01.01.2013	01.09.2009
8	06.01.2009	08.01.2009	01.06.2009	01.01.2014	01.11.2009
9	07.01.2009	09.01.2009	01.07.2009	01.01.2015	01.01.2010
10	08.01.2009	12.01.2009	01.08.2009	01.01.2016	
11	09.01.2009	13.01.2009	01.09.2009	01.01.2017	
12	10.01.2009	14.01.2009	01.10.2009	01.01.2018	
13	11.01.2009	15.01.2009	01.11.2009	01.01.2019	
14	12.01.2009	16.01.2009	01.12.2009	01.01.2020	
15	13.01.2009	19.01.2009	01.01.2010	01.01.2021	
16	14.01.2009	20.01.2009	01.02.2010	01.01.2022	
17					

Abbildung 5.43:
Optionen zur
Berechnung von
Datumsreihen

In die dritte Zeile jeder Spalte wurde das Datum 1.1.09 eingetragen, anschließend der entsprechende Bereich selektiert, beispielsweise A3:A16 und danach mit unterschiedlichen Optionen eine Datumsreihe berechnet:

- ➔ Spalte A: Typ *Datum*, Zeiteinheit *Tag*, Schrittweite (*Inkrement*) 1; Excel erhöht das Datum entsprechend um jeweils einen Tag.
- ➔ Spalte B: Typ *Datum*, Zeiteinheit *Wochentag*, Schrittweite 1; Excel erhöht das Datum um jeweils einen Tag, berücksichtigt dabei aber nur Wochentage und übergeht Samstage und Sonntage.
- ➔ Spalte C: Typ *Datum*, Zeiteinheit *Monat*, Schrittweite 1; Excel erhöht das Datum um jeweils einen Monat (1.1.2009, 1.2.2009 etc.).
- ➔ Spalte D: Typ *Datum*, Zeiteinheit *Jahr*, Schrittweite 1; Excel erhöht das Datum um jeweils ein Jahr (1.1.2009, 1.1.2010 etc.).
- ➔ Spalte E: Typ *Datum*, Zeiteinheit *Monat*, Schrittweite 2, Endwert 1.1.2010; wegen der Schrittweite 2 erhöht Excel das Datum um jeweils zwei Monate. Da zusätzlich der Endwert 1.1.2010 angegeben wurde, füllt Excel den markierten Bereich nicht vollständig aus, sondern hört beim Erreichen des Enddatums auf.

Zahlenreihen Die Einheiten *Arithmetisch* und *Geometrisch* beziehen sich nicht auf Tages-, sondern auf Zahlenreihen. *Arithmetisch* meint Reihen, bei denen die jeweils nächste Zahl durch Addition der Schrittweite *Inkrement* zur vorhergehenden Zahl gebildet wird:

Arithmetische Folgen: $N+1 = N + \text{Schrittweite}$

Mit der Schrittweite 2 und dem Anfangswert 10 ergibt sich damit zum Beispiel die Folge

10, 12, 14, 16 ...

Bei geometrischen Reihen wird die Schrittweite dagegen nicht zur jeweils letzten Zahl addiert, sondern mit ihr multipliziert:

Geometrische Folgen: $N+1 = N * \text{Schrittweite}$

Mit der gleichen Schrittweite 2 und wiederum dem Anfangswert 10 ergibt sich diesmal die Folge

10, 20, 40, 80 ...

Einige Beispiele für beide Reihenarten (Abbildung 5.44):

	A	B	C	D	E	F	G	H	I
1	Arithmetische Reihen				Geometrische Reihen				
2	Schritt.1	Schritt.2	Schritt.1,5	Schritt.5		Schritt.1	Schritt.2	Schritt.3	Schritt.4
3	1	1	1	5		1	1	1	5
4	2	3	2,5	10		1	2	3	20
5	3	5	4	15		1	4	9	80
6	4	7	5,5	20		1	8	27	320
7	5	9	7	25		1	16	81	1280
8	6	11	8,5	30		1	32	243	5120
9	7	13	10	35		1	64	729	20480
10	8	15	11,5	40		1	128	2187	81920
11	9	17	13	45		1	256	6561	327680
12	10	19	14,5	50		1	512	19683	
13	11	21	16			1	1024	59049	
14	12	23	17,5			1	2048	177147	
15									

Abbildung 5.44:
Optionen zur
Berechnung von
Zahlenreihen

- ➔ Spalte A: Arithmetische Reihe mit Schrittweite 1; Excel erhöht die Zahl fortlaufend um 1.
- ➔ Spalte B: Dasselbe mit der Schrittweite 2
- ➔ Spalte C: Wie Sie sehen, können auch krumme Schrittweiten verwendet werden, keineswegs nur ganzzahlige. Die Schrittweite 1,5 mit der Basiszahl 1 führt entsprechend zur Folge »1«, »2,5«, »4«, »5,5« etc.
- ➔ Spalte D: Schrittweite 5 und Angabe von 50 als Endwert
- ➔ Spalte F: Geometrische Reihe mit dem Startwert 1 und der Schrittweite 1
- ➔ Spalte G: Dasselbe mit der Schrittweite 2
- ➔ Spalte H: Mit der Schrittweite 3
- ➔ Spalte I: Mit der Schrittweite 4, dem Startwert 5 und dem Endwert 1.000.000. Die unterste Zahl ist die letzte Zahl der Folge, die diesen Endwert noch nicht überschreitet. Die nächste Zahl wäre bereits erheblich größer, daher beendet Excel die Berechnung.

Die Option *Trend* veranlasst Excel zu einer Trendberechnung, zur Durchführung einer linearen oder exponentiellen Regression.

Trend

Angenommen, bei einem Experiment wird eine Variable in gleichförmigen Schritten verändert und es ergeben sich dabei für eine andere Variable folgende Messwerte:

0,9
2,1
3,67

Excel kann einen linearen oder exponentiellen Zusammenhang zwischen den Werten einer Reihe nach der Methode der kleinsten quadratischen Abweichung berechnen, um eine Gleichung zu finden, die an diese Werte möglichst gut angepasst ist, und die Reihe entsprechend fortzusetzen. Allerdings müssen Sie selbst angeben, ob eine lineare oder eine exponentielle Regression gewünscht wird.

Tragen Sie die drei Messwerte in eine Spalte ein und selektieren Sie diese Werte und die darunter liegenden Zellen, in denen die Reihe fortgesetzt werden soll.

Aktivieren Sie im Dialogfeld von BEARBEITEN|AUSFÜLLEN|REIHE... das Kontrollkästchen *Trend* und als Reihentyp *Arithmetisch* oder *Geometrisch*; in unserem Fall aktivieren Sie bitte entsprechend der folgenden Abbildung *Geometrisch* (Abbildung 5.45).

Abbildung 5.45:
Vorbereitung einer
Trendberechnung

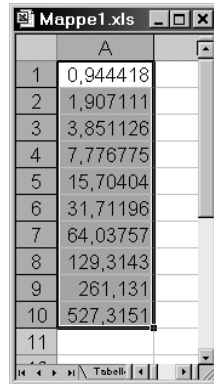


Excel stellt die Exponentialgleichung auf, die den vorliegenden Werten am besten angepasst ist.

Danach werden entsprechend dieser Gleichung die vorgegebenen Werte überschrieben und die Reihe fortgesetzt, der resultierende Trend also extrapoliert (Abbildung 5.46).



Für eher wirtschaftlich interessierte Leser: Die vorgegebenen Daten können ebenso gut die Umsätze einer Firma in den Jahren 2009, 2010 und 2011 sein. Dann würde Excel den zu Grunde liegenden Trend ermitteln und die Fortsetzung der Reihe entspräche einer Schätzung der zukünftigen Umsatzentwicklung.



	A
1	0,944418
2	1,907111
3	3,851126
4	7,776775
5	15,70404
6	31,71196
7	64,03757
8	129,3143
9	261,131
10	527,3151
11	

Abbildung 5.46:
Ergebnis der Trend-
berechnung

5.11 Suchen und Ersetzen

Mit BEARBEITEN|SUCHEN... können Sie in einer Excel-Tabelle eine Zeichenkette suchen und gegebenenfalls durch eine andere ersetzen. Das vollständige Dialogfeld sehen Sie jedoch nur, wenn Sie anschließend auf *Optionen* >> klicken (Abbildung 5.47).

Suchen

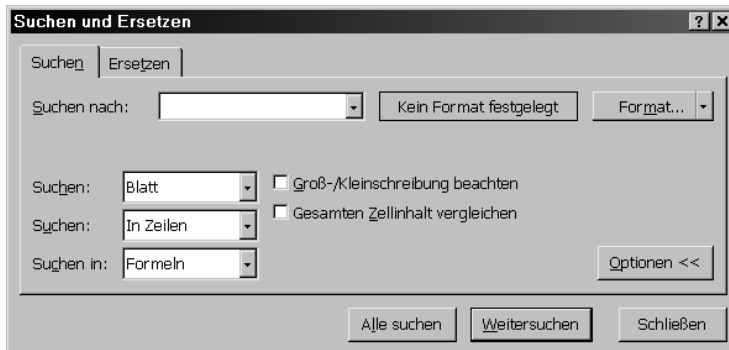


Abbildung 5.47:
Zeichenkette
suchen

Ist vor Anwahl dieses Befehls ein Bereich selektiert, wird nur dieser nach der unter *Suchen nach* eingegebenen Zeichenkette durchsucht. Ist kein Bereich selektiert, wird die gesamte Tabelle durchsucht.

Ob Sie bei der Eingabe des Suchbegriffs Groß- oder Kleinschreibung verwenden, spielt keine Rolle. Außer Sie aktivieren *Groß-/Kleinschreibung beachten*, da Excel nur dann zwischen Groß- und Kleinschreibung unterscheidet.

Ist *Gesamten Zelleninhalt vergleichen* deaktiviert, wird »Jan« auch in einer Zelle gefunden, die »Dies ist Jan 2009« enthält; sonst ausschließlich in Zellen, die nur den Suchbegriff »Jan« enthalten.

Die Optionen *Blatt* und *Arbeitsmappe* im Listenfeld *!!!Suchen!!!* legen fest, ob nur das aktuelle Blatt oder die gesamte Arbeitsmappe durchsucht wird.

Die Optionen *In Zeilen* und *In Spalten* im Listenfeld *Suchen* bestimmen, ob der ausgewählte Bereich Zeile für Zeile oder Spalte für Spalte durchsucht wird (was meist auf das Gleiche hinausläuft).

Selektieren Sie im Listenfeld *Suchen in* die Option *Formeln*, wird ein Suchbegriff wie »4« nur in Zellen gefunden, die in der Bearbeitungsleiste dieses Zeichen enthalten, unabhängig davon, was in der Zelle selbst angezeigt wird.

Der Suchbegriff »4« würde beispielsweise in der Formel $=3*4$ ebenso gefunden wie in einer Zelle, die einfach die Zahl 4 enthält; aber nicht in einer Zelle, die die Formel $=2*2$ enthält, da dann die Zahl 4 zwar als Wert der Zelle angezeigt wird, diese Zahl aber nicht in der Formel selbst enthalten ist!

Mit der Option *Werte* wird »4« genau umgekehrt in allen Formeln gefunden, die den Wert 4 besitzen; also in einer Zelle, die diese Zahl enthält ebenso wie in einer Zelle mit der Formel $=2*2$, die diesen Wert besitzt.

Kommentare durchsucht nur Zellkommentare nach dem Suchtext.

Das Listenfeld *Format...* ermöglicht Ihnen, nur Zellen zu durchsuchen, die eine bestimmte Formatierung aufweisen:

- ➔ Wählen Sie den gleichnamigen Unterbefehl *FORMAT...*, so können Sie anschließend im Dialogfeld zum Formatieren von Zellen die gewünschte Zellformatierung definieren (siehe Kapitel 6, »Formatieren«).
- ➔ Wählen Sie *FORMAT VON ZELLE WÄHLEN...*, so ändert der Cursor seine Form: Klicken Sie nun einfach auf die Zelle, die als Formatvorlage dienen soll, um nur gleichartig formatierte Zellen zu durchsuchen.

Das ermöglicht Ihnen, nur den rot formatierten Text »Brutto« durch den blau und fett formatierten Text »Bruttobetrag« zu ersetzen.

Excel zeigt eine Vorschau der auf diese Weise definierten Zellformate an und die Formateinschränkungen bei der Suche bleiben auch bei weiteren Suchaktionen erhalten, bis Sie den Unterbefehl *LÖSCHEN: FORMAT SUCHEN* wählen.

Mit *Weitersuchen* starten Sie die Suche. Excel setzt den Zellcursor auf die erste Zelle, in der der Suchbegriff entdeckt wird. Klicken Sie erneut auf *Weitersuchen*, wird die Suche fortgesetzt.

Zwischendurch können Sie die Sucheinstellungen jederzeit ändern und das Dialogfeld per Packen und Ziehen an der Titelleiste verschieben, wenn die gefundene Zelle davon verdeckt wird.

Ist der Suchbegriff gefunden und Sie stellen fest, dass Sie ihn an der betreffenden Stelle durch eine andere Zeichenkette ersetzen wollen, aktivieren Sie das Register *Ersetzen...*, das auch durch den Befehl BEARBEITEN|ERSETZEN aufgeschlagen wird (Abbildung 5.48).

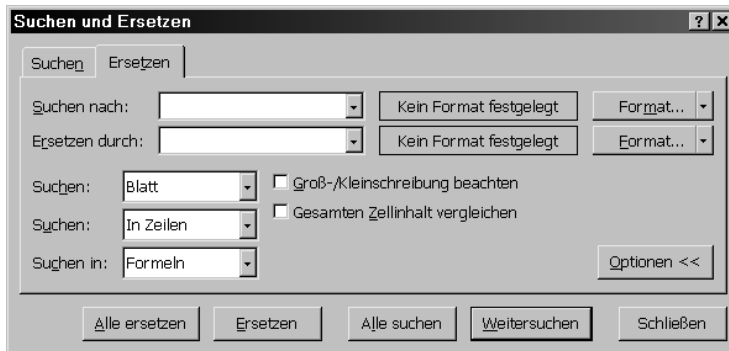
Ersetzen

Abbildung 5.48:
Zeichenkette
ersetzen

Unter *Ersetzen durch* geben Sie die Zeichenkette ein, durch die der Suchbegriff ersetzt werden soll.

Alle ersetzen ersetzt den Suchbegriff in der gesamten Tabelle durch diesen zweiten Begriff.

Ersetzen führt dazu, dass er in der zuletzt gefundenen Zelle ersetzt wird und Excel Ihnen die nächste ihn enthaltende Zelle zeigt. Wählen Sie erneut *Ersetzen*, wird er auch in dieser ersetzt und so weiter.

Weitersuchen ersetzt den Suchbegriff nicht, sondern übergeht die Zelle ohne Aktion und zeigt die nächste ihn enthaltende Zelle an.

Wählen Sie *Weitersuchen* und drücken Sie dabei die -Taste, so wird der Suchbegriff in der aktuellen Zelle ebenfalls nicht ersetzt. Anschließend zeigt Ihnen Excel jedoch nicht die nächste, sondern genau umgekehrt die *vorhergehende* Zelle des Bereichs, die den Suchbegriff enthält.

Mit *Ersetzen* und *Weitersuchen* können Sie den ausgewählten Bereich somit in Richtung zum Bereichsende oder Bereichsanfang nach dem Suchbegriff absuchen und ihn nur in ganz bestimmten Zellen ersetzen.

Statt Texte und Zahlen können Sie auch reine Formatierungen in Ihren Tabellen ersetzen. Dazu lassen Sie beide Eingabefelder leer und geben unter Suchen nach nur die zu suchende Formatierung an, beispielsweise Kursiv, und unter Ersetzen durch die gewünschte neue Formatierung, beispielsweise Rot und Fett.



Für eher wirtschaftlich interessierte Leser: Die vorgegebenen Daten können ebenso gut die Umsätze einer Firma in den Jahren 2009, 2010 und 2011 sein. Dann würde Excel den zu Grunde liegenden Trend ermitteln und die Fortsetzung der Reihe entspräche einer Schätzung der zukünftigen Umsatzentwicklung.

Nehmen Sie als praktisches Beispiel eine Tabelle, die mehrfach Formeln wie $=C8/116*16$ enthält, die die Mehrwertsteuer errechnen, die in einem Betrag enthalten ist, der sich in einer anderen Zelle befindet.

Erhöht sich der Mehrwertsteuersatz von 16% auf 20%, können Sie alle Formeln der Tabelle, die ihn verwenden, per Hand ändern, oder aber das Register *Ersetzen* aktivieren und folgende Einstellungen verwenden (Abbildung 5.49).

Abbildung 5.49:
Praktisches Beispiel



Sie dürfen keinesfalls *Alle ersetzen* verwenden! Befindet sich irgendwo im Zahlenbereich die Zahl 16, wird auch diese durch 20 ersetzt. Stattdessen müssen Sie sich mit *Weitersuchen* Schritt für Schritt vorantasten und prüfen, ob die Bearbeitungsleiste eine der Formeln enthält, in der 16 durch 20 ersetzt werden soll. Nur wenn das der Fall ist, aktivieren Sie *Ersetzen*.

5.12 Gehe zu



Die einfachste Möglichkeit, direkt zu einer bestimmten Zelle zu gelangen oder einen Bereich zu selektieren: Sie klicken auf das geschlossene Listenfeld ganz links in der Bearbeitungsleiste, in dem ständig Ihre aktuelle Position angezeigt wird. In diesem Feld tippen Sie nun den gewünschten Zell- oder Bereichsbezug ein und gelangen nach  direkt dorthin.

Erheblich weitergehende Möglichkeiten bietet der Befehl **BEARBEITEN|GEHE ZU...** (.

Er öffnet ein Dialogfeld, in dem Sie eine Zellreferenz wie C5 eingeben können, die danach direkt angesprungen wird. Geben Sie stattdessen einen Bereichsbezug wie E4:E6 ein, wird dieser Bereich selektiert (Abbildung 5.50).

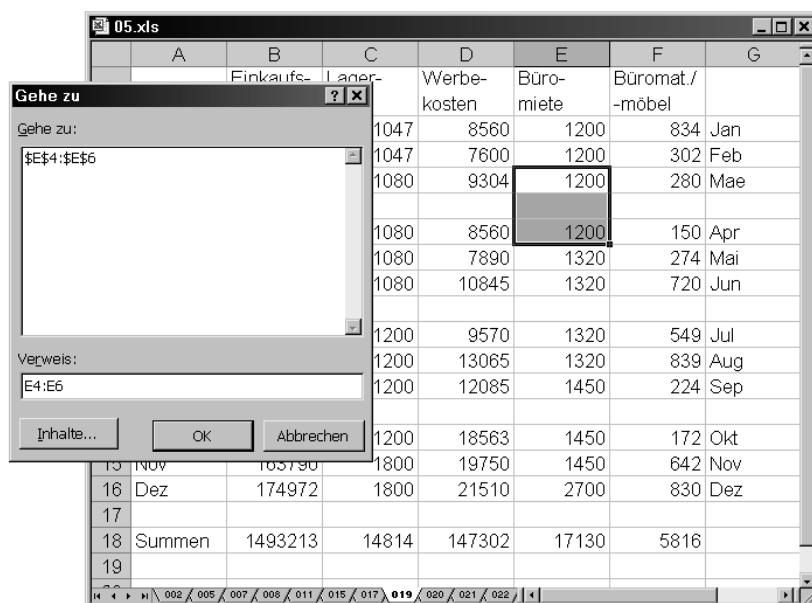


Abbildung 5.50:
»Gehe zu« auf
Bereich anwenden

Befindet sich vor dem Bezug ein Tabellenname, gefolgt von einem Ausrufezeichen (»Jan!C5«), wird die Tabelle aktiviert und die darin angegebene Zelle selektiert.

Mit einem dem Bezug vorangestellten und in eckige Klammern eingeschlossenen Arbeitsmappenamen wie »[BUCHFRG.XLS]Jan!C5« wird die angegebene Mappe und darin wiederum die betreffende Tabelle aktiviert und in dieser Tabelle die angegebene Zelle selektiert.

Statt Zellreferenzen können Sie auch entsprechend definierte Namen verwenden und beispielsweise mit »MWSt« zu jener Zelle springen, die den Namen »MWSt« besitzt.

Im Listenfeld *Gehe zu* werden jene Referenzen angezeigt, zu denen Sie zuletzt gesprungen sind. Selektieren einer Zellreferenz und OK (oder Doppelklick auf die Zellreferenz) genügt, um sofort wieder dorthin zu gelangen.

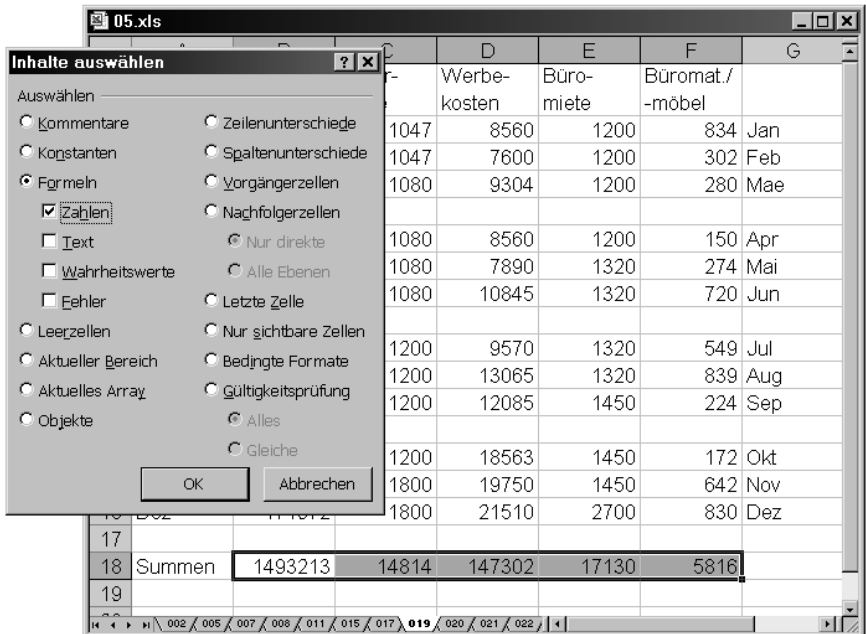
Übrigens befindet sich der letzte Sprung immer am Listenanfang und ist unter *Verweis* vorselektiert. OK genügt daher, um jeweils zur letzten Position zurückzukehren. Drücken Sie immer wieder **F5** und **↩**, so springen Sie entsprechend immer wieder von A nach B und umgekehrt.

5.12.1 Bestimmte Datenarten suchen

Die Schaltfläche *Inhalte...* öffnet ein Dialogfenster, in dem Sie gezielt bestimmte Datenarten auswählen können. Selektieren Sie zuvor einen Bereich, so wird nur dieser nach der gewünschten Datenart durchsucht. Ist dagegen kein Bereich selektiert und nur eine Zelle aktiv, wird die gesamte Tabelle durchsucht.

- ➔ Mit der Option *Formeln* und dem zugehörigen Kontrollkästchen *Zahlen* wird die Tabelle nach Zellen durchsucht, die als Resultate Zahlen liefern (eine Formel kann auch andere Resultate liefern, zum Beispiel einen Text) (Abbildung 5.51).

Abbildung 5.51:
Datenart auswählen



Stattdessen können Sie mit der Option *Fehler* alle Formeln suchen, die Fehlerwerte wie #WERT? oder #NAME? liefern, also aus irgendeinem Grund fehlerhaft sind. Mit *Text* können Sie Formeln suchen, die Texte liefern, und mit *Wahrheitswerte* Formeln, die Wahrheitswerte übergeben. Sind mehrere Kontrollkästchen aktiv, werden alle Formeln selektiert, die eine der zugehörigen Datenarten enthalten.

- ➔ *Kommentare* sucht Zellen, an die ein Zellkommentar gebunden ist.
- ➔ *Konstanten* sucht Zellen, die ausschließlich Zahlen, Zeichenketten oder ein Datum enthalten.
- ➔ *Leerzellen* sucht alle Zellen, die überhaupt nichts enthalten.

- ➔ *Aktueller Bereich* selektiert ebenso wie das zugehörige Symbol (Kategorie *Bearbeiten* des *Anpassen*-Dialogfelds) alle Zellen, die um die aktive Zelle herum einen rechteckigen Bereich bilden. Wenn vor der Befehlswahl die Zelle C7 die aktive Zelle der zuletzt abgebildeten Tabelle ist, selektiert *Aktueller Bereich* den Bereich A6:G8, in dem sich diese Zelle befindet und der ringsherum von leeren Zellen umgeben ist (Abbildung 5.52).

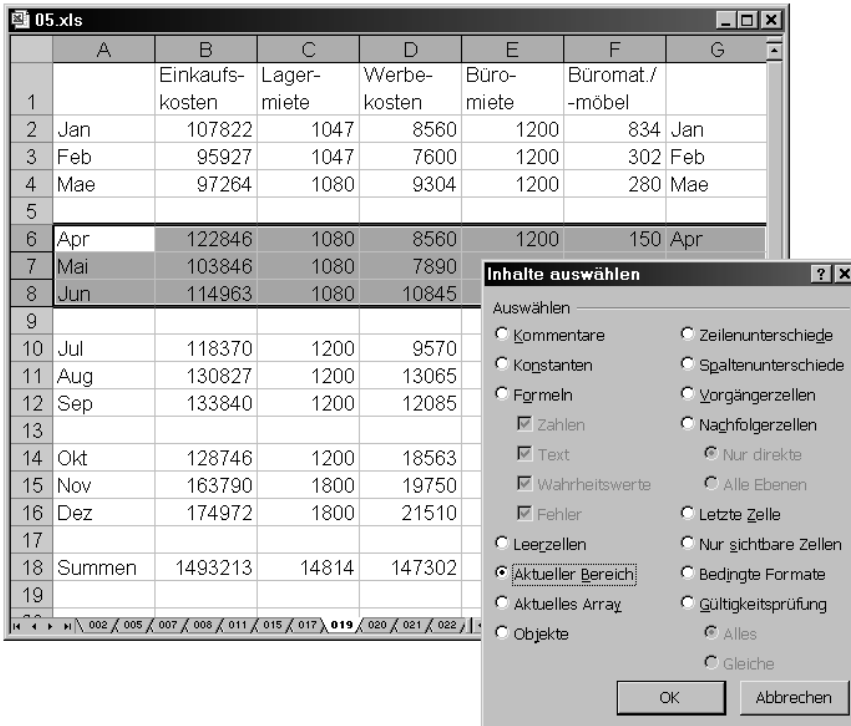


Abbildung 5.52:
Aktuellen Bereich
selektieren

- ➔ *Aktuelles Array* selektiert das gesamte Array, zu dem die aktive Zelle gehört, analog zur Tastenkombination **Strg** + **⇧** + **/**.
- ➔ *Letzte Zelle* selektiert die letzte Zelle (rechts unten) der Tabelle, die Daten oder Formate enthält.
- ➔ *Nur sichtbare Zellen* selektiert ebenso wie das zugehörige Symbol (Kategorie *Bearbeiten* des *Anpassen*-Dialogfelds) im ausgewählten Bereich oder (wenn nichts selektiert ist) der gesamten Tabelle nur sichtbare Zellen. Per Gliederungsfunktion ausgeblendete Spalten oder Zeilen werden dagegen nicht selektiert, sodass diese durch nachfolgende Editierungen des nun selektierten Bereichs auch nicht beeinflusst werden.
- ➔ *Objekte* selektiert alle Grafikobjekte.





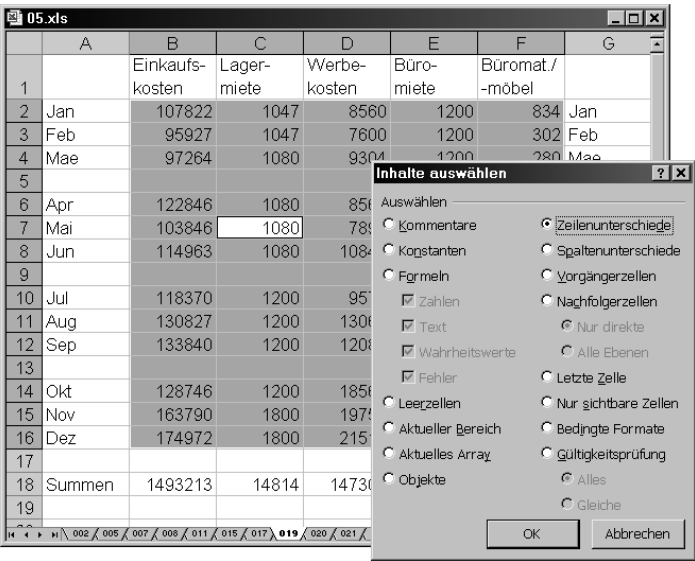
Bedingte Formate wählt jene Zellen aus, denen so genannte bedingte Formate zugewiesen wurden (siehe Kapitel 6.12, »Bedingte Formatierung«). Ist dabei statt Alles das Kontrollkästchen Gleiche aktiviert, werden nur Zellen mit den gleichen bedingten Formaten wie die aktuelle Zelle ausgewählt.

Gültigkeitsprüfung wählt jene Zellen aus, denen Gültigkeitsregeln zugewiesen wurden (siehe Kapitel 38.4, »Gültigkeitsregeln benutzen«). Ist zusätzlich statt Alles das Kontrollkästchen Gleiche aktiviert, werden nur Zellen mit den gleichen Gültigkeitsregeln wie die aktuelle Zelle ausgewählt.

5.12.2 Zeilen- und Spaltenunterschiede suchen

Vor der Benutzung der Optionen *Zeilenunterschiede* und *Spaltenunterschiede* müssen Sie zunächst den zu bearbeitenden Bereich auswählen, auch dann, wenn das die gesamte Tabelle ist (Abbildung 5.53).

Abbildung 5.53:
Vorbereitung einer
Auswahl nach
Zeilen- oder
Spalten-
unterschieden



Hier selektierte ich den Bereich B2:F16 und steuerte den Zellcursor darin zu C7 – ohne dabei die Auswahl zu löschen, also nicht mit den Cursortasten, sondern mit und !

Wählen Sie anschließend BEARBEITEN|GEHE ZU..., aktivieren die Schaltfläche *Inhalte...* und wählen im zugehörigen Dialogfeld die Option *Zeilenunterschiede*, so erhalten Sie folgendes Resultat (Abbildung 5.54).

Vergleichsspalte

Excel benutzt die Spalte C, in der sich die aktive Zelle befand, als Vergleichsspalte: Jede Zeile des selektierten Bereichs wird nach Werten abgesucht, die sich von dem Wert unterscheiden, der sich in Spalte C der betreffenden Zeile befindet.

	A	B	C	D	E	F	G
1		Einkaufs-	Lager-	Werbe-	Büro-	Büromat./	
2	Jan	107822	1047	8560	1200	834	Jan
3	Feb	95927	1047	7600	1200	302	Feb
4	Mae	97264	1080	9304	1200	280	Mae
5							
6	Apr	122846	1080	8560	1200	150	Apr
7	Mai	103846	1080	7890	1320	274	Mai
8	Jun	114963	1080	10845	1320	720	Jun
9							
10	Jul	118370	1200	9570	1320	549	Jul
11	Aug	130827	1200	13065	1320	839	Aug
12	Sep	133840	1200	12085	1450	224	Sep
13							
14	Okt	128746	1200	18563	1450	172	Okt
15	Nov	163790	1800	19750	1450	642	Nov
16	Dez	174972	1800	21510	2700	830	Dez
17							
18	Summen	1493213	14814	147302	17130	5816	
19							

Abbildung 5.54:

Auswahl nach
Zeilenunterschieden

Zunächst wird die oberste Zeile des selektierten Bereichs untersucht, also Zeile 2: Der in der Vergleichsspalte C dieser Zeile enthaltene Wert 1047 wird mit den restlichen Werten in dieser Zeile verglichen und jede Zelle selektiert, die einen anderen Wert enthält, also alle Zellen dieser Zeile, B2, D2, E2 und F2.

Danach werden auf die gleiche Weise in der folgenden Zeile 3 alle Werte selektiert, die sich vom in C4 enthaltenen Wert unterscheiden und so weiter.

Als Resultate sind in allen Zeilen – abgesehen natürlich von der Vergleichsspalte selbst – alle Werte selektiert, mit Ausnahme der Zeilen 5, 9 und 13.

Warum in diesen Zeilen kein einziger Wert selektiert ist, ist leicht zu erklären. Nehmen wir als Beispiel Zeile 5: Der Vergleichswert in Spalte C dieser Zeile ist eine leere Zelle. Da alle anderen Zellen dieser Zeile ebenfalls leer sind, gibt es keinen Unterschied zwischen ihnen und dem Vergleichsfeld und sie werden entsprechend auch nicht selektiert. Das gleiche gilt für die Zeilen 9 und 13.

Interessanter wird es, wenn wir erneut B2:F16 selektieren, wieder C7 als aktive Zelle wählen und den Bereich diesmal nach Spaltenunterschieden durchsuchen lassen (Abbildung 5.55).

Mit dieser Option vergleicht Excel jede Zelle der Zeile 7, in der sich die aktive Zelle befindet, mit den Werten, die sich in der gleichen Spalte, aber in anderen Zeilen befinden.

Vergleichszeile

Abbildung 5.55:
Auswahl nach Spaltenunterschieden

	A	B	C	D	E	F	G
1		Einkaufskosten	Lagermiete	Werbekosten	Büromiete	Büromat./-möbel	
2	Jan	107822	1047	8560	1200	834	Jan
3	Feb	95927	1047	7600	1200	302	Feb
4	Mär	97264	1080	9304	1200	280	Mär
5							
6	Apr	122846	1080	8560	1200	150	Apr
7	Mai	103846	1080	7890	1320	274	Mai
8	Jun	114963	1080	10845	1320	720	Jun
9							
10	Jul	118370	1200	9570	1320	549	Jul
11	Aug	130827	1200	13065	1320	839	Aug
12	Sep	133840	1200	12085	1450	224	Sep
13							
14	Okt	128746	1200	18563	1450	172	Okt
15	Nov	163790	1800	19750	1450	642	Nov
16	Dez	174972	1800	21510	2700	830	Dez
17							
18	Summen	1493213	14814	147302	17130	5816	
19							

In Spalte B sind außer der Vergleichszelle B7 alle Zellen selektiert, da alle einen anderen Inhalt als den Vergleichswert 103846 enthalten. In Spalte C sind dagegen mehrere Zellen nicht selektiert, da sie ebenso wie die Vergleichszelle C7 den Wert 1080 enthalten und so weiter.



Die aktive Zelle, die die Vergleichsspalte bzw. Vergleichszeile definiert, muss sich keineswegs im selektierten und zu durchsuchenden Bereich befinden! Es kann ebenso gut irgendeine Zelle außerhalb des Bereichs sein, auch wenn sich die Vergleichszeile/-spalte dann möglicherweise ebenfalls außerhalb dieses Bereichs befindet.

Außerdem können Sie nicht nur Bereiche durchsuchen, die Konstanten enthalten. Ebenso gut können die Bereiche Formeln oder Kommentare enthalten oder eine Mischung all dieser Datenarten.

Bei Formeln ist wichtig, dass Excel zwei Formeln nicht im Bezugsformat A1 miteinander vergleicht, sondern sie vor dem Vergleich intern (also ohne die eingetragene Formel tatsächlich zu verändern) in das Format Z1S1 umwandelt.

Angenommen, A3 enthält die Formel

=2*B3

und eine Zeile darunter enthält A4 die Formel

=2*B4,

dann wird Excel beide Formeln als identisch ansehen. Die Bezüge B3 und B4 sind zwar scheinbar nicht identisch. Excel vergleicht aber intern im Z1S1-Format und da sich beide Bezüge auf die jeweils rechts benachbarte Zelle beziehen, lauten beide in diesem Format ZS(1) und sind somit identisch. Das gleiche gilt für die vollständigen Formeln, die daher für Excel

=2*ZS(1)

und

=2*ZS(1)

lauten.

5.12.3 Vorgänger- und Nachfolgerzellen suchen

Mit *Vorgängerzellen* durchsucht Excel die gesamte Tabelle nach Zellen, auf die sich die Bezüge in jenen Formeln beziehen, die der zuvor selektierte Bereich enthält.

Angewandt auf die vorhergehende Abbildung bedeutet das: Die in Zeile 18 enthaltenen Formeln addieren die Inhalte der Zellen darüber. B18 enthält beispielsweise die Formel =SUMME(B2:B16). Selektieren Sie zuvor B18:C18, so selektiert *Vorgängerzellen* daher die beiden Bereiche B2:B16 und C2:C16, da sich die Formeln in den beiden selektierten Zellen auf die in diesen Bereichen enthaltenen Zellen beziehen.

Entsprechend selektiert *Nachfolgerzellen* genau umgekehrt alle Zellen einer Tabelle, die von den ausgewählten Zellen abhängig sind. Würden Sie in der abgebildeten Tabelle B7 selektieren, selektiert *Nachfolgerzellen* daher die Zelle B18, da sich die darin enthaltene Formel unter anderem auf B7 bezieht und daher vom Inhalt dieser Zelle abhängig ist.

Mit den Optionen *Nur direkte* und *Alle Ebenen* entscheiden Sie in beiden Fällen, ob Excel beim Selektieren nur direkte oder auch indirekte Abhängigkeiten berücksichtigt.

Befindet sich beispielsweise in A1 der Wert 10 und enthält zusätzlich A2 die Formel

=2*A1

und A3 die Formel

=A2+100,

dann ist A2 direkt und A3 indirekt von A1 abhängig. Selektieren Sie A1 und wählen *Nachfolgerzellen* mit der Option *Nur direkte*, wird Excel nur A2 auswählen; mit *Alle Ebenen* dagegen auch die indirekt (über A2) von A1 abhängige Zelle A3.

Selektieren Sie in diesem Beispiel A3 und wollen Sie mit *Vorgängerzellen* wissen, auf welche Zellen sich die darin enthaltene Formel bezieht, so wird auf alle Fälle A2 selektiert, da A1 einen direkten Bezug auf diese Zelle enthält.

Nur mit der Option *Alle Ebenen* wird jedoch zusätzlich A1 selektiert werden, eine Zelle, auf die A3 keinen indirekten Bezug enthält, sondern nur einen indirekten Bezug über Zelle A2, die wiederum einen direkten Bezug auf A1 enthält.

6 Formatieren

Beim Formatieren geht es um das Layout einer Tabelle, also um die rein optische Seite. Durch geschicktes Formatieren einzelner Zellen oder ganzer Bereiche gewinnt eine komplexe Tabelle erheblich an Übersichtlichkeit.

Ich zeige Ihnen anhand des Schritt für Schritt weiterentwickelten Demos »Buchführung«, wie Sie Zellen Formate zuweisen und damit Zahlen individuell darstellen, beispielsweise mit zwei Tausenderpunkten und zwei Nachkommastellen, mit einem zusätzlichen Währungssymbol »DM« oder »\$«, in roter Farbe, falls die Zahl negativ ist und so weiter.

Ähnliches ist mit Datums- und Uhrzeitangaben möglich und darüber hinaus können Sie natürlich sowohl Zahlen als auch Texte in nahezu beliebigen Schriftgrößen und Schriftarten darstellen, unterstreichen, fett hervorheben etc.

Per Drag&Drop können Sie die Breite von Spalten und die Höhe von Zeilen verändern. Sie können Zellen mit Rahmen in beliebigen Farben versehen und damit oder durch einen andersfarbigen Zellhintergrund hervorheben. Sie können sogar den Inhalt einer beliebigen Grafikdatei als Hintergrund Ihrer Tabelle verwenden.

Formatierungsprofis sehen anschließend, wie Sie beliebige Formatdefinitionen zu einer Formatvorlage zusammenfassen, die Sie jederzeit per Mausklick einer Zelle oder einem Bereich zuweisen können, sodass der betreffenden Zelle alle darin definierten Formatierungsmerkmale gleichzeitig zugewiesen werden.

Sind Sie bequem, so verwenden Sie zum Formatieren einer Tabelle einfach die AutoFormatieren-Funktion, bei der Excel einem Bereich oder der gesamten Tabelle auf Knopfdruck eines von vielen vorgegebenen Formatierungsschemata zuweist.

Dazu analysiert Excel den Bereich und versucht herauszufinden, welche Teile Daten enthalten und wo sich zugehörige Spalten-/Zeilenüberschriften und Summen befinden – und hebt letztere deutlich hervor.

Darüber hinaus erläutere ich die bedingte Formatierung, die es ermöglicht, in einem zuvor selektierten Bereich nur jene Zellen zu formatieren, auf die

bestimmte Bedingungen zutreffen. Eine sehr elegante Technik, um in einem Bereich beispielsweise jene Zahlen hervorzuheben, die einen bestimmten Grenzwert über- oder unterschreiten.



Merken Sie sich bitte: Formatierungen sind nicht an den Inhalt einer Zelle gebunden, sondern ein Bestandteil der Zelle selbst, egal was diese gerade enthält!

Wenn sich in Zelle A1 der Text »Ein Test« befindet und Sie diesen Text fett formatieren, erscheint zwar der Text fett. Sie haben das Format »Fett« jedoch nicht diesem speziellen Text zugewiesen, sondern *der Zelle selbst!*

Das Format »Fett« ist nun eine Eigenschaft der Zelle. Geben Sie einen anderen Text in A1 ein, wird daher auch dieser fett dargestellt. Das gleiche gilt für jede andere Formatierung!



Wie ich in Anhang 7, »Excel individuell konfigurieren«, erläutere, formatiert Excel Zellen immer dann selbstständig, wenn die benachbarten Zellen eine einheitliche Formatierung aufweisen. Geben Sie beispielsweise in A5 eine Zahl ein und sind alle darüber liegenden Zellen A1 bis A4 fett formatiert, formatiert Excel A5 unmittelbar nach Ihrer Eingabe ebenfalls fett. Passt Ihnen diese Automatik nicht, können Sie sie wie im genannten Kapitel erläutert deaktivieren.

6.1 Die Format-Symboleiste

Kaum eine Symbolleiste ist so nützlich wie die Format-Symboleiste. Das Symbol mit dem fettgedruckten »F« formatiert beispielsweise Zahlen und Text in der aktiven Zelle bzw. dem selektierten Bereich fett, wenn Sie es anklicken. Nach dem Anklicken ist das Symbol hervorgehoben. Erneutes Anklicken hebt die Formatierung wieder auf.

Der Zustand dieser Symbole informiert Sie beim Wandern durch eine Tabelle über die Formatierung einer Zelle oder eines Bereichs. Ist er hervorgehoben, gilt diese Formatierung für die gerade aktive Zelle bzw. mindestens einen Teil des gerade selektierten Bereichs.

Die über die Format-Symboleiste erreichbaren Formatierungsmöglichkeiten reichen Ihnen möglicherweise bereits vollständig, sodass der Rest des Kapitels in diesem Fall uninteressant für Sie wäre (Abbildung 6.1).

Die Spalten D und E enthalten den ursprünglichen Zustand von Texten und Zahlen. Die Spalten G und H enthalten den Zustand nach der Selektion der betreffenden Zelle und dem Anklicken des in Spalte A per Stichwort beschriebenen Leisten-Symbols.

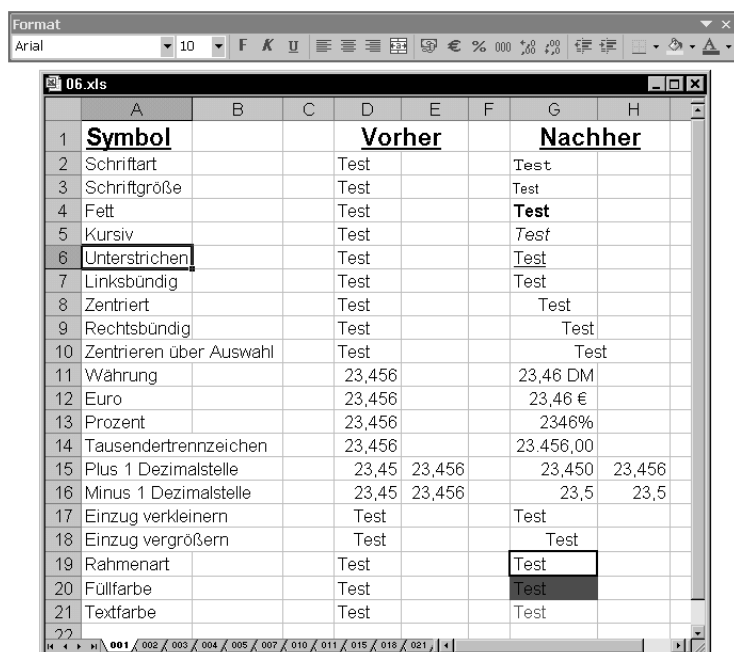


Abbildung 6.1:
Symbolleisten-
Formatierungen

Die Symbole der Formateiste wurden der Reihe nach von links nach rechts geordnet angewendet: Zeile 2 beschreibt die Funktion des Symbols ganz links, des Listenfelds zur Schriftartwahl, und Zeile 20 die Funktion des Symbols ganz rechts, mit dem die Textfarbe verändert wird.

Die Auswirkungen der meisten Symbole sind anhand dieser Tabelle verständlich. Die meisten Symbole werden auf den nachfolgenden Seiten detailliert erläutert. Zu einigen will ich Ihnen dennoch bereits jetzt ein paar Erläuterungen geben:

Das Währungs- und das Euro-Symbol hängen »DM« bzw. »€« an, weisen zwei Nachkommastellen zu und fügen bei entsprechend hohen Werten Tausenderpunkte ein.

*Währungs-
Symbole*

Die beiden Symbole mit den beiden kleinen nach rechts/links weisenden Pfeilchen erhöhen bzw. vermindern die Nachkommastellenanzahl bei jedem Anklicken schrittweise um 1: Aus 23,456 würde 23,4560 beziehungsweise 23,46, je nachdem, welches der beiden Symbole Sie anklicken.

Dezimalstellen

Vor Anwahl dieser beiden Symbole wurden ausnahmsweise jeweils zwei Zellen selektiert, D14 und E14 beziehungsweise D15 und E15. Sind vor dem Erhöhen oder Vermindern wie in diesem Fall mehrere Zellen selektiert, orientiert sich Excel an der Zelle mit den meisten Nachkommastellen und weist die daraus resultierende Nachkommastellenanzahl allen Zellen des selektierten Bereichs zu.

Im Beispiel besitzen E14 und E15 mit jeweils drei die höchste Nachkommastellenanzahl. Daher werden beim Erhöhen des Bereichs D14:E14 beiden Zellen vier Nachkommastellen zugewiesen und beim Vermindern beiden Zellen zwei Nachkommastellen.

Beide Symbole erlauben somit eine sehr einfache *einheitliche* Nachkommastellen-Formatierung auch umfangreicher Zahlenbereiche.



Weitere Formatierungs-Symbole finden sich unter den Kategorien *Format im Anpassen-Dialogfeld*.

6.2 Formate zuweisen und kopieren

Nach Selektion der zu formatierenden Zelle oder eines Bereichs wählen Sie **FORMAT|ZELLEN...** (oder klicken mit der rechten Maustaste auf den selektierten Bereich, wählen im Kontextmenü **ZELLEN FORMATIEREN...**) und aktivieren das Register mit der gewünschten Formatierungskategorie.

In den Listefeldern wählen Sie die gewünschten Formatierungsvarianten aus und aktivieren/deaktivieren gegebenenfalls noch ein paar zusätzliche Optionen.

In jedem Fall zeigt Excel anhand eines Beispiels, wie sich die betreffende Formatierung auswirken würde. Sind Sie damit einverstanden, müssen Sie nur noch auf **OK** klicken, um der aktiven Zelle bzw. dem selektierten Bereich die Formatierung zuzuweisen.

Formatanzeige

Die verschiedenen Listfelder und Kontrollkästchen besitzen jedoch noch eine zweite Aufgabe: Ihnen zu zeigen, welches Format eine bestimmte Zelle oder ein Bereich momentan besitzt (Abbildung 6.2).

Vor Anwahl von **FORMAT|ZELLEN...** selektierte ich den Bereich B2:B3. Im Dialogfeld aktivierte ich das Register *Schrift*. Darin sind nun alle Einstellungen vorselektiert, die auf den aktuellen Bereich zutreffen.

Offenbar gelten für diesen Bereich unter anderem die Formate *Schriftschnitt* »Standard« und *Schriftgrad* 10.

Wähle ich nun stattdessen den Schriftgrad 12 und den Stil »Fett«, schließe das Dialogfeld mit **OK**, selektiere B2:B4 und öffne das Dialogfeld erneut, so erhalte ich folgende Darstellung (Abbildung 6.3).

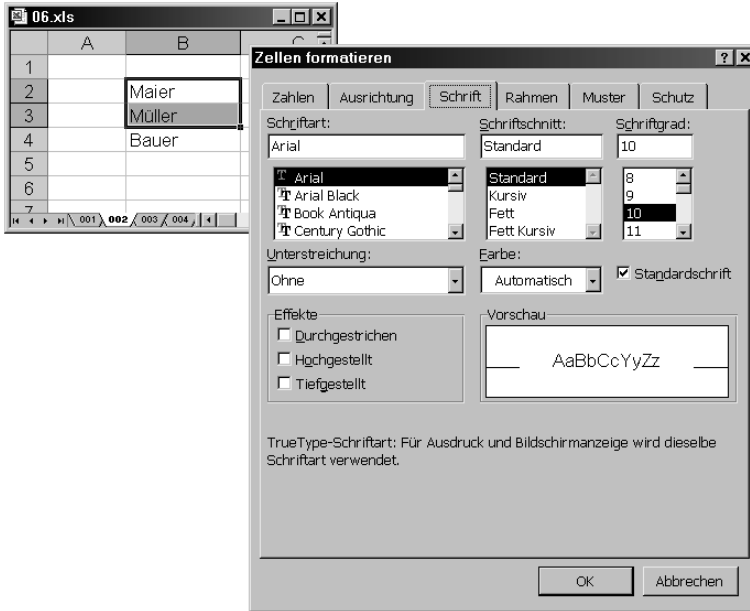


Abbildung 6.2:
Anzeige der
gültigen
Formatierungen

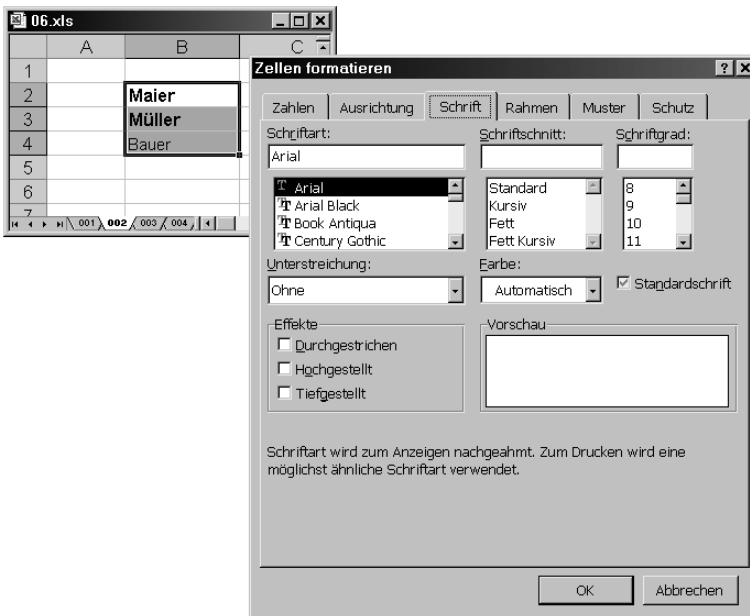


Abbildung 6.3:
Uneinheitliche
Schriftformatierung
des selektierten
Bereichs

Im Listenfeld *Schriftschnitt* ist diesmal kein Element vorselektiert. Zwei der drei selektierten Zellen, B2 und B3, wurden soeben fett und in dem Schriftgrad 12 formatiert, die Zelle B4 besitzt diese Formatierungen jedoch nicht. Daher gibt es keinen Schriftschnitt und keinen Schriftgrad, die einheitlich für den gesamten selektierten Bereich gelten.

Formate löschen Zugewiesene Formate können Sie jederzeit überschreiben, indem Sie der Zelle oder dem Bereich ein anderes Format zuweisen oder mit dem Befehl BEARBEITEN|LÖSCHEN|FORMATE entfernen. Er löscht nur Formatierungen; Formeln, Texte oder Zahlen bleiben erhalten. Danach gelten wieder die Standard-Formateinstellungen von Excel (Texte linksbündig ausrichten, Zahlen rechtsbündig, Schriftgrad 10 etc.).

Formate kopieren Sie können die Formate einer Zelle oder eines selektierten Bereichs kopieren, indem Sie zunächst wie gewohnt KOPIEREN, dann jedoch statt BEARBEITEN|EINFÜGEN den Befehl BEARBEITEN|INHALTE EINFÜGEN... wählen und im zugehörigen Dialogfeld *Formate* selektieren.



Oder Sie können einfach nach Selektion der Zelle, die die zu kopierenden Formate enthält, dieses Symbol anklicken: Der Mauscursor ändert seine Form. Klicken Sie anschließend eine andere Zelle an, dann werden alle Formate der zuvor selektierten Zelle auf diese übertragen.

Wollen Sie Formate in mehrere Zellen/Bereiche kopieren, doppelklicken Sie auf das Symbol. Es bleibt dann solange aktiv, bis Sie es erneut anklicken.

6.3 Demo »Buchführung«

Die Formatierungsmöglichkeiten von Excel führe ich anhand der Buchführung eines Selbstständigen oder Freiberuflers vor, die dieser für seine Einnahme-/Überschussrechnung und die monatlichen Vorsteuerzahlungen an das Finanzamt benötigt.

Ausgangspunkt ist folgende Tabelle (Abbildung 6.4):

Abbildung 6.4:
Format.xls,
»Tabelle1«

B9 =B8/107*7

	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230	3859,5	2	19,8	284	184	473	1728
4		1637,58		4	53,2	76,5	396		
5				3,7		73			
6						68,9			
7									
8	Brutto	1867,58	3859,5	9,7	73	502,4	580	473	1728
9	MWSt	122,178131	532,344828	0	10,0689655	69,2965517	80	65,2413793	238,344828
10	Netto	1745,40187	3327,15517	9,7	62,9310345	433,103448	500	407,758621	1489,65517
11									
12					Netto	MWSt			
13	- Steuerpflichtige Umsätze mit 16%				3327,15517	532,344828			
14	- Steuerpflichtige Umsätze mit 7%				1745,40187	122,178131			
15	Gesamt				5072,55704	654,522958			
16	Vorsteuerbeträge					462,951724			
17	Umsatzsteuervorauszahlung					191,571234			
18									

In den Spalten B und C sollen die Einnahmen eingetragen werden, unterteilt in die Sparten »Bücher« und »Programme«. Beides muss getrennt erfasst werden, da für die Umsatzsteuervorauszahlungen die jeweils erhaltene Umsatzsteuer wichtig ist: bei Programmen wird der volle Umsatzsteuersatz von 16% berechnet, bei Büchern jedoch nur der ermäßigte Satz von 7%.

Auch die Ausgaben werden entsprechend in verschiedene Kostenarten unterteilt.

Die Werte selbst sollen in den Zeilen 3 bis 7 in die betreffenden Spalten eingetragen werden. In der Praxis werden Sie für die Einnahmen/Ausgaben eines Monats natürlich wesentlich mehr Zeilen vorsehen, mindestens 20 oder 30.

In Zeile 8 werden die Bruttobeträge ermittelt: Die Zellen B8 bis I8 enthalten die Formeln $=\text{SUMME}(B3:B7)$, $=\text{SUMME}(C3:C7)$, $=\text{SUMME}(D3:D7)$ etc., addieren also jeweils die Werte der darüber liegenden Zellen.

In Zeile 9 wird die in diesen Bruttobeträgen enthaltene Umsatzsteuer ermittelt: B9 enthält die Formel $=B8/107*7$ (siehe Bearbeitungsleiste der Abbildung), errechnet also die im Bruttobetrag aller Einnahmen aus Buchverkäufen (Zelle B8) bereits enthaltenen 7% Umsatzsteuer.

Analog dazu ermittelt die in C9 enthaltene Formel $=C8/116*16$ die im Bruttobetrag aller Einnahmen aus Programmen (Zelle C8) enthaltenen 16% USt.

Die gleiche Formel befindet sich im Bereich D9:I9, abgesehen von Zelle D9, die einfach den Wert 0 enthält, da weder bei Briefmarken noch bei Telefonrechnungen Umsatzsteuer anfällt.

In Zeile 10 werden die Nettobeträge durch Subtraktion der Umsatzsteuer von den Bruttobeträgen ermittelt. Entsprechend enthält B10 die Formel $=B8-B9$, C10 die Formel $=C8-C9$ und so weiter.

Für die monatlichen Umsatzsteuervorauszahlungen wird zusätzlich eine Gesamtübersicht der USt-Einnahmen und -Ausgaben benötigt, die darunter enthalten ist.

Die einzelnen Zellen und die darin enthaltenen Formeln:

- ➔ E13: $=C10$ (Nettobetrag der Einnahmen aus Programmen)
- ➔ F13: $=C9$ (zugehörige USt)
- ➔ E14: $=B10$ (Nettobetrag der Einnahmen aus Büchern)
- ➔ F14: $=B9$ (zugehörige USt)

In E15 und F15 werden diese Brutto- und USt-Beträge addiert. E15 enthält die Formel $=E13+E14$ und F15 die Formel $=F13+F14$. E15 enthält somit die Summe aller Nettoeinnahmen, F15 die Summe aller USt-Einnahmen.

In F16 wird mit $=SUMME(D9:I9)$ die Summe aller USt-Ausgaben ermittelt und in F17 dieser Betrag mit $=F15-F16$ von den USt-Einnahmen subtrahiert.

Das Ergebnis, der Überschuss der USt-Einnahmen gegenüber den USt-Ausgaben, ist jener Betrag, der als Umsatzsteuervorauszahlung für den betreffenden Monat an die Finanzkasse zu überweisen ist, im Beispiel rund 188 DM.

Diese Mini-Buchführung ist primitiv, beispielsweise für mich selbst jedoch völlig ausreichend – zumindest in einer leicht erweiterten und durch verschiedene Formatierungen übersichtlich gestalteter Form.

Unsere erste Tat sollte darin bestehen, die Zahlen etwas einheitlicher zu formatieren und vor allem die katastrophale Darstellung von völlig überflüssigen drei oder gar vier Nachkommastellen zu unterbinden.

6.4 Zahlen-, Datums- und Zeitformate

Im Register *Zahlen* des Befehls **FORMAT|ZELLEN...** können Sie das gewünschte Format zur Darstellung einer Zahl, eines Datums oder einer Uhrzeit auswählen. Zuvor müssen Sie die zu formatierende Zelle oder den Bereich selektieren (Abbildung 6.5).

Abbildung 6.5:
Verfügbare Formate



In diesem Dialogfeld ist zunächst immer das Format vorgegeben, das die gerade aktive Zelle bzw. der selektierte Bereich besitzt, normalerweise »Standard«.

Allgemeinformat

In diesem Format werden Zahlen rechts- und Texte linksbündig ausgerichtet. Dabei wird jede Zahl genauso dargestellt, wie Sie sie eingaben – mit Ausnahme zu langer Ziffernfolgen, bei denen Excel das Exponentialformat wählt.

Stattdessen können Sie unter *Kategorie* jedoch eine beliebige andere Formatierungskategorie wählen, beispielsweise »Zahl«, wenn Sie eine allgemeine Zahl wie 23,4 formatieren wollen, »Währung«, wenn die Zahl mit einem Währungszusatz wie »DM« versehen werden soll, oder »Datum«, wenn ein Datum wie »21.12.2009« formatiert werden soll.

Formatkategorien

Je nach Kategorie erscheinen daraufhin im Dialogfeld zusätzliche Auswahlfelder und Kontrollkästchen, mit denen Sie das betreffende Format genauer definieren können.

Zum Beispiel können Sie bei der Kategorie »Zahl« die gewünschte Dezimalstellenanzahl angeben und wählen, ob Tausenderpunkte angezeigt werden sollen oder nicht.

Wenn Sie »Datum« wählen, können Sie stattdessen in einer Liste wählen, ob das betreffende Datum als »21.12.2009«, als »21.Dezember 2009« oder in einer anderen Datumsform angezeigt wird.

In jedem Fall wird im Feld *Beispiel* angezeigt, wie sich die betreffenden Formatierungseinstellungen auf die momentan aktive Zelle auswirken würde – vorausgesetzt, diese Zelle enthält überhaupt eine Zahl bzw. ein Datum!

Haben Sie einer Zelle ein etwas verunglücktes Format zugewiesen, weisen Sie ihr einfach das Ausgangsformat Standard zu, um diese Formatierung rückgängig zu machen.



Vernünftige Zahlenformatierungen verbessern die Übersichtlichkeit erheblich. Angewandt auf die kleine Buchführung könnten Sie beispielsweise die Bereiche B3:I10, E13:F15 und F16:F17 selektieren und formatieren.

Entweder formatieren Sie jeden Bereich separat oder aber Sie selektieren gleich alle Bereiche (beim Selektieren des jeweils nächsten Bereichs die Strg-Taste gedrückt halten).

Wählen Sie danach FORMAT|ZELLEN... und in der Kategorie »Zahl« zwei Nachkommastellen, um alle in diesen Bereichen enthaltenen Zahlen mit zwei Nachkommastellen darzustellen (Abbildung 6.6).

Abbildung 6.6:
Format.xls,
»Tabelle2«

	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12					Netto	MWSt			
13	- Steuerpflichtige Umsätze mit 16%				3327,16	532,34			
14	- Steuerpflichtige Umsätze mit 7%				1745,40	122,18			
15	Gesamt				5072,56	654,52			
16	Vorsteuerbeträge					462,95			
17	Umsatzsteuervorauszahlung					191,57			
18									

6.4.1 Zahlenformate

Allgemeine Zahlen Bei allgemeinen Zahlen können Sie unter *Dezimalstellen* festlegen, wie viele Dezimalstellen angezeigt werden sollen, wobei gegebenenfalls gerundet wird (Abbildung 6.7).

Abbildung 6.7:
Zahlenformate

Zellen formatieren

Zahlen Ausrichtung Schrift Rahmen Muster Schutz

Kategorie: Standard Zahl Währung Buchhaltung Datum Uhrzeit Prozent Bruch Wissenschaft Text Sonderformat Benutzerdefiniert

Beispiel: 1637,58

Dezimalstellen: 2

☐ Mit 1000er-Trennzeichen (.)

Negative Zahlen: -1234,10 1234,10 -1234,10 -1234,10

Zahl wird für die allgemeine Anzeige von Zahlen verwendet. Währung und Buchhaltung bieten spezielle Formate für monetäre Werte.

OK Abbrechen

»2« würde die Zahl 123,456 beispielsweise als 123,46 anzeigen.

Aktivieren Sie *Mit 1000er-Trennzeichen (.)*, so werden bei Zahlen mit mehr als drei Vorkommastellen Tausenderpunkte angezeigt. Beispiel: 12345,678 wird zu 12.345,678.

Negative Zahlen legt fest, wie negative Zahlen dargestellt werden: ob ein Minuszeichen verwendet wird, ob die betreffende Zahl stattdessen rot angezeigt wird oder das Minuszeichen und die Farbe Rot miteinander kombiniert werden.

Bei den Währungsformaten können Sie im Listenfeld *Symbol* einen zusätzlichen internationalen Währungstext wie »EUR«, »DEM« oder »USD« auswählen (oder mit »Keine« auf einen solchen Zusatztext verzichten). Darüber hinaus sind in diesem Listenfeld auch alle Währungssymbole wie »\$« oder das Euro-Symbol »€« verfügbar (Abbildung 6.8).

Währungsformate

Abbildung 6.8:
Währungsformate

Da bei Währungsformaten automatisch Tausenderpunkte benutzt werden, erscheint eine Zahl wie 12345,678 (mit dem Symbol »EUR« und Wahl von zwei Dezimalstellen) in der Form 12.345,68 EUR.

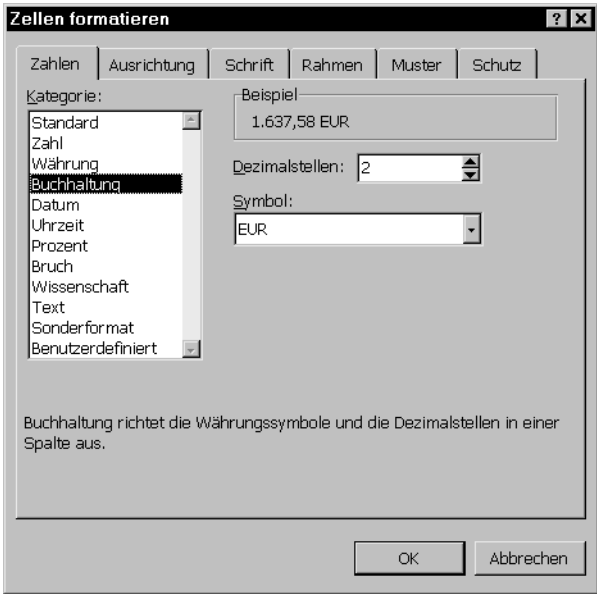
Die Kategorie »Buchhaltung« ermöglicht die perfekte Ausrichtung von Zahlen, die sich untereinander befinden (Abbildung 6.9).

Buchhaltungs-
formate

Sie können die Dezimalstellenanzahl und einen Währungszusatz wie »EUR«, »DM«, »\$« etc. festlegen.

Entscheidend ist, dass sich das Minuszeichen vor einer negativen Zahl wie -123,56 nicht mehr wie üblich unmittelbar vor der Zahl befindet, sondern *am linken Zellrand*.

Abbildung 6.9:
Buchhaltungs-
formate



Zusätzlich wird ohne Währungssymbol (Wahl von »Keine«) am rechten Zellrand Platz für ein entsprechendes Symbol gelassen.

Angewandt auf die beiden Zahlen 123,456 und -123,456 (Abbildung 6.10).

Abbildung 6.10:
Anwendung

Alle Zellen formatierte ich im Buchhaltungsformat. Nur bei der Formatierung der Zellen in den beiden unteren Zeilen wurde der Zusatz »EUR« aktiviert.

Obwohl in den beiden oberen Zellen dieser Zusatz nicht verwendet wird, bleibt am rechten Zellrand dennoch der dafür benötigte Platz erhalten, sodass entsprechend der Abbildung alle Zahlen perfekt untereinander ausgerichtet sind, sowohl positive als auch negative Zahlen und es ist egal, ob dabei der Zusatz »EUR« verwendet oder auf ihn verzichtet wird.

Prozentformate

Im Prozentformat wird das %-Zeichen angehängt und die betreffende Zahl mit 100 multipliziert (Abbildung 6.11).



Abbildung 6.11:
Prozentformate

Mit *Null Nachkommastellen* würde 12,567 entsprechend als 1257% angezeigt.

Im Bruchformat können Sie zunächst mit den drei ersten Varianten wählen, ob die Zahl als ein-, zwei- oder dreistelliger Bruch angezeigt wird (Abbildung 6.12).

Brüche

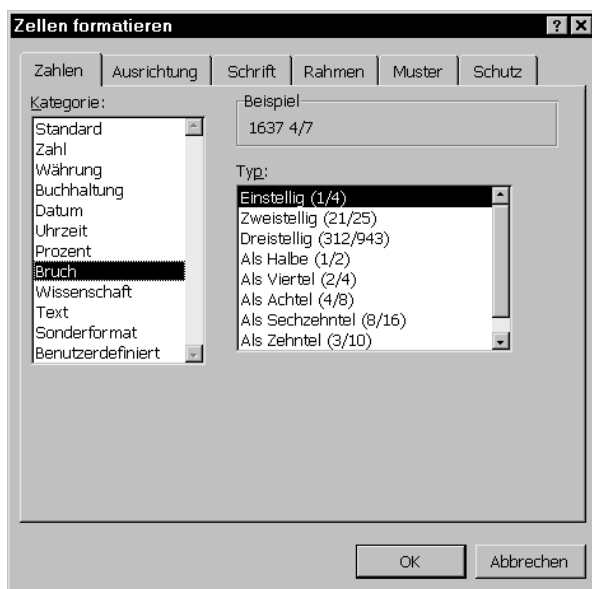


Abbildung 6.12:
Bruchformate

Die erste Variante (einstelliger Bruch) würde beispielsweise 1,2345 als $1 \frac{1}{4}$ anzeigen, die zweite (zweistelliger Bruch) als $1 \frac{19}{81}$ und die dritte (dreistelliger Bruch) als $1 \frac{87}{371}$. Diese drei Varianten unterscheiden sich somit durch die Stellenanzahl des Nenners.

Die restlichen Varianten zeigen in Bruchform an, wie viele Achtel, Sechzehntel etc. in den Nachkommastellen enthalten sind. Beispiel: »(8/16)« würde 123,45 als $123 \frac{7}{16}$ anzeigen, da in 0,45 sieben Sechzehntel enthalten sind.

Wissenschaftliches Format

Im wissenschaftlichen Format werden Zahlen als Exponentialzahlen angezeigt (Abbildung 6.13).

Abbildung 6.13:
Wissenschaftliches
Format



Beispiel: 12345,678 wird zu $1,23E+04$ (1,23 mal 10 hoch 4).

Textformat

Das Textformat ist nützlich, wenn Zellen Zahlen enthalten, diese aber als Texte behandelt werden sollen (Abbildung 6.14).

Optionen und Auswahlmöglichkeiten gibt es zwar keine, das Resultat ist dennoch befriedigend: Die Jahreszahlen 2009, 2010 etc. werden von Excel normalerweise als Zahlen interpretiert und entsprechend rechtsbündig ausgerichtet (siehe linke Spalte).

Wie die rechte Spalte zeigt, fasst Excel diese Zahlen jedoch nach Zuweisung des Textformats als Texte auf und richtet sie entsprechend linksbündig aus.

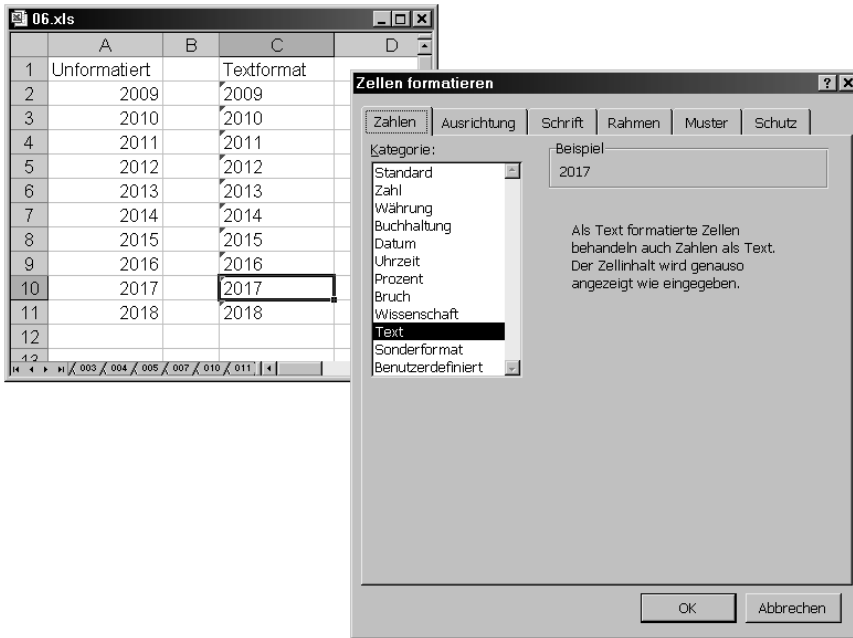


Abbildung 6.14:
Textformat

Die kleinen Dreiecke in den Zellecken sind »Indikatoren«. Klicken Sie auf eine Zelle mit einem solchen Indikator, erscheint – wie immer, wenn Excel meint, einen möglichen Fehler entdeckt zu haben – das Fehler-Smart Tag, dessen Liste entsprechende Fehlerbehebungs-Optionen anbietet. Die Indikatoren verschwinden, wenn Sie im Register Fehlerüberprüfung des Befehls EXTRAS|OPTIONEN... das Kontrollkästchen Als Text gespeicherte Zahlen deaktivieren, da Excel Ihnen dann »glaubt«, dass es sich nicht um ein Versehen handelt, sondern Sie ganz bewusst Zahlen als Texte formatieren.



Einige Formen von Texten interpretiert Excel als fehlerhafte Formeleingaben. Geben Sie beispielsweise »- Abschreibung« oder »+ Sonstiges« in eine Zelle ein, nimmt Excel an, Sie wollten eine Formel eingeben und ergänzt automatisch das einleitende »=«; es ergänzt Ihre Eingaben also zu »=- Abschreibung« bzw. »=+ Sonstiges«. Da diese Formeln jedoch fehlerhaft sind, erscheint in der Zelle anschließend der Fehlerwert #NAME?.

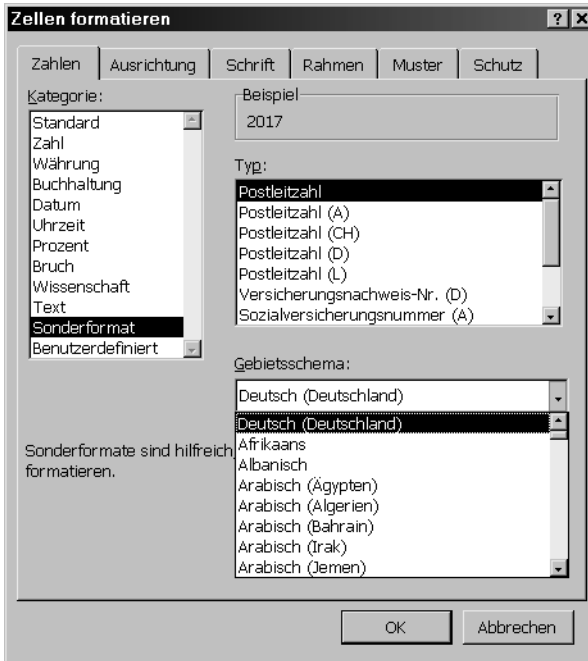


Um diese Fehlinterpretation zu vermeiden, formatieren Sie die betreffenden Zellen erst im Textformat und geben danach »- Abschreibung« oder »+ Sonstiges« ein. Dann unterlässt Excel den Versuch, Ihre Texte als Formeln zu interpretieren.

Wie der Name bereits sagt, enthält die Kategorie »Sonderformat« einige spezielle Formatierungen (Abbildung 6.15).

Sonderformate

Abbildung 6.15:
Sonderformate



Das erste Format würde die Zahl 123 entsprechend in Postleitzahlenform darstellen, als 00123. Das zweite Format würde ein »D-« davor setzen (»D-00123«) und so weiter.



Im Listenfeld Gebietsschema können Sie sich ein anderes Land als die Vorgabe »Deutschland« aussuchen, woraufhin in der Liste darüber die Sonderformate des betreffenden Landes angezeigt werden, bei den Vereinigten Staaten unter anderem das Format der berühmten »Sozialversicherungsnummer«.

Dieses Listenfeld mit landesspezifischen Formaten ist auch bei den nachfolgend beschriebenen Datums- und Zeitformaten verfügbar, die in unterschiedlichen Ländern teilweise sehr unterschiedlich gehandhabt werden.

6.4.2 Datums- und Zeitformate

Excel verfügt über eine Unmenge vorgegebener Datumsformate (Abbildung 6.16).

Datumseingabe

Welches Sie wählen, ist prinzipiell egal. Wichtiger ist die Art und Weise, in der Sie ein Datum in eine Zelle eingeben, da ein Datum wie der 12.2.2009 in einer Form eingegeben werden muss, durch die es für Excel eindeutig als Datum identifizierbar ist!

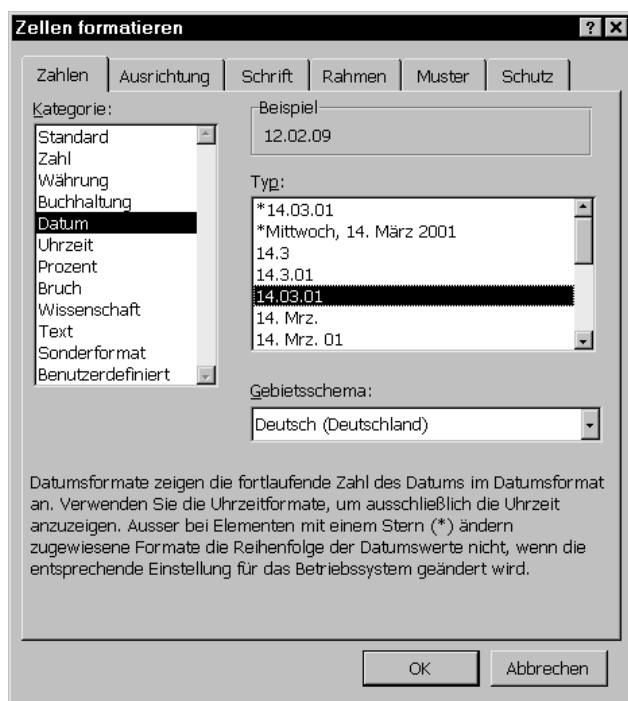


Abbildung 6.16:
Datumsformate

Zunächst einmal müssen alle drei Datumskomponenten durch Punkte (12.2.2009), Bindestriche (12-2-09) oder Bruchstriche (12/2/09) oder Kombinationen dieser Zeichen (12.2-09 oder 12/2-09) voneinander getrennt werden.

Die Tageskomponente muss eine ein- oder zweistellige Zahl sein (»31«, »07« oder »7«).

Die Monatskomponente muss ebenfalls eine ein- oder zweistellige Zahl (»02« oder »2« für den Februar) sein oder aber ein ausgeschriebener oder mit den ersten drei Buchstaben abgekürzter Monatsname (»Februar« oder »Feb«).

Das Jahr ist eine ein-, zwei oder vierstellige Zahl (für 1999 wahlweise »1999« oder »99«, für 2009 genügt »9«, da Excel bei einstelligen Zahlen ab dem Jahr 2000 beginnt).

Geben Sie einen Monatsnamen ein, wird kein Trennzeichen verwendet. Das Jahr folgt direkt danach oder durch Leerzeichen vom Monatsnamen getrennt (»12.Feb09« oder »12.Feb 09« oder »12. Feb 09« oder »12. Feb 09«). Groß-/Kleinschreibung wird ignoriert (»12.Feb 09« oder »12.FEB 09«).



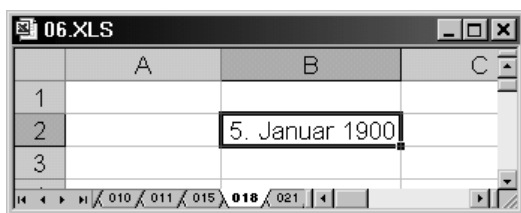
Die Anzeige eines formatierten Datums sieht möglicherweise zunächst merkwürdig aus. Eventuell erscheint in der betreffenden Zelle nur eine Reihe von Nummernzeichen »#####«. Sie zeigen an, dass die Spaltenbreite nicht ausreicht, um dieses Datum anzuzeigen.

Verbreitern Sie die Spalte, indem Sie auf die dünne Linie zwischen den Spaltenbuchstaben A und B klicken und sie bei gedrückter Maustaste nach rechts ziehen (Tastatur: FORMAT|SPALTE|BREITE... wählen und im zugehörigen Dialogfeld einen größeren als den vorgegebenen Wert eingeben).

Serielle Zahlen

Geben Sie ein für Excel identifizierbares Datum ein, so weist Excel der betreffenden Zelle automatisch das zugehörige Datumsformat zu. Geben Sie jedoch anschließend in der gleichen Zelle eine Zahl wie 5 ein, passiert Merkwürdiges (Abbildung 6.17).

Abbildung 6.17:
Zahleneingabe in
Zelle mit Datums-
format



Statt der Zahl 5 wird das Datum »5. Januar 1900« angezeigt! Die Erklärung: Ein Datum speichert Excel in Wahrheit nicht als Zeichenfolge, sondern wandelt es in eine serielle Zahl um, gemäß folgender Regel: Excels Zeitrechnung beginnt beim 1.1.1900. Diesem Datum ist die Zahl 1 zugeordnet. Dem 2.1.1900 ist die Zahl 2 zugeordnet, dem 3.1.1900 die Zahl 3 und so weiter.

Ist einer Zelle ein Datumsformat zugewiesen, zeigt Excel darin *immer* ein Datum an, auch wenn Sie eine Zahl wie 5 eingeben. Die 5 wird einfach als serielle Zahl interpretiert und das zugehörige Datum angezeigt, im Beispiel das Datum 5.1.1900, das der seriellen Zahl 5 zugeordnet ist.

Soll dagegen die *Zahl 5* angezeigt werden, müssen Sie der betreffenden Zelle das Format »Standard« oder eines der Zahlenformate zuweisen.

Umgekehrt gilt: Weisen Sie einer Zelle eines der Zahlenformate zu, so zeigt Excel darin *immer* eine Zahl an. Geben Sie ein Datum wie den 12.2.2009 ein, ermittelt Excel daher die zugeordnete serielle Zahl 39856 und zeigt sie an.

Soll jedoch wie üblich nicht diese serielle Zahl, sondern das zugehörige Datum angezeigt werden, müssen Sie der Zelle das Format »Allgemein« oder eines der Datumsformate zuweisen.

Um in einer Tabelle das aktuelle Datum und die Uhrzeit anzuzeigen, verwenden Sie folgende Formel:

=JETZT()

Sie bewirkt die Anzeige des aktuellen Datums – und zwar ständig aktualisiert! Heute wird zum Beispiel »12.2.09 10:09« angezeigt.

Öffnen Sie die Tabelle morgen um 15 Uhr erneut, lautet die Anzeige: »13.2.09 15:00«.

Analog dazu zeigt

=HEUTE()

nur das Datum an, ohne Uhrzeit.

Beachten Sie unbedingt, dass die Zellen, in der sich diese Formeln befinden, das Allgemein- oder eines der Datumsformate besitzen müssen. Sonst wird statt des Datums und/oder der Uhrzeit die zugehörige serielle Zahl angezeigt!

Excel verfügt über folgende vordefinierte Zeitformate (Abbildung 6.18):

Zeitformate

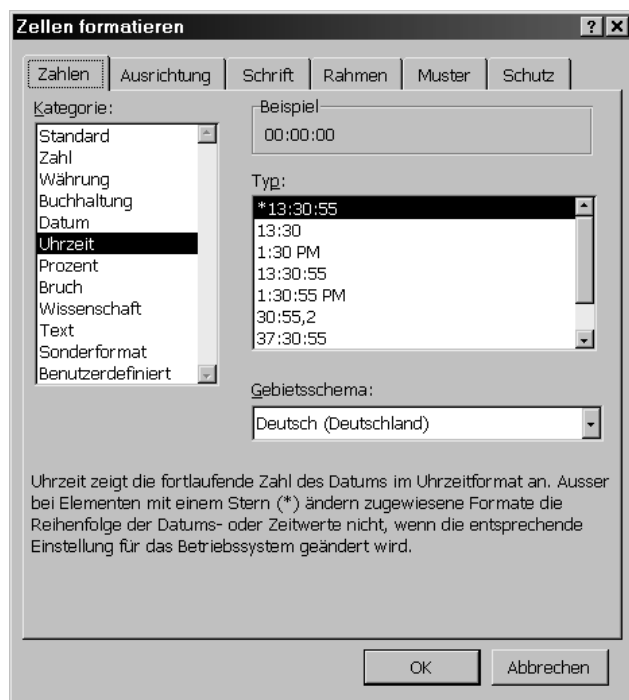


Abbildung 6.18:
Zeitformate

»PM« kennzeichnet dabei die Zwölf-Stunden-Einteilung, bei der auf 12 Uhr Vormittag nicht wie bei der 24-Stunden-Einteilung 13 Uhr, sondern 1 Uhr nachmittags folgt. Der Zusatz »AM« bzw. »PM« zeigt dabei an, ob Vormittag (AM) oder Nachmittag (PM) gemeint ist.

Sie können eine Uhrzeit in einem beliebigen dieser Formate eingeben, wobei zur Trennung der Zeitkomponenten jeweils ein Doppelpunkt verwendet wird. Die Zeit »14 Uhr, 50 Minuten, 15 Sekunden« geben Sie beispielsweise als »14:50:15« ein.

Sie können (siehe die beiden letzten Formate) auch Mischungen aus einem Datum und einer Uhrzeit eingeben. Dazu geben Sie das Datum in irgendeinem der gültigen Datumsformate ein und dahinter oder davor – aber durch ein Leerzeichen getrennt – die Zeit in einem der vordefinierten Zeitformate.

Zwei Beispiele für gültige gemischte Eingaben sehen Sie hier: »12-2-94 14:50:15« oder »Feb 94 2:50:15 PM«.



Auch einer Uhrzeit weist Excel intern eine serielle Zahl zu. Beachten Sie daher die entsprechenden Erläuterungen im Abschnitt über Datumsformate!

6.4.3 Selbstdefinierte Formate

Die Kategorie »Benutzerdefiniert« ermöglicht Ihnen ebenfalls die Auswahl unter mehreren vorgegebenen Formaten, aber vor allem die Definition eigener Formate (Abbildung 6.19).

Um eigene Formate zu definieren, müssen Sie verstehen, wie Formatbeschreibungen wie die hier vorgegebenen aufgebaut sind.

Das Prinzip ist immer das gleiche: Sie definieren eine Art Schablone, die aus wenigstens einem und maximal drei Abschnitten besteht, getrennt durch je ein Semikolon.

Formatabschnitte

Der erste Abschnitt definiert die Darstellung positiver, der zweite die Anzeige negativer Zahlen und der dritte die Anzeige der Zahl 0.

Wird anstelle eines der drei Abschnitte ersatzweise nur ein Semikolon angegeben, zeigt Excel eine dem betreffenden Abschnitt zugeordnete Zahl überhaupt nicht an. Entsprechend verbirgt das Format »;;;« die in einer Zelle enthaltene Zahl vollständig. Es wird nichts angezeigt, egal, ob die Zahl positiv, negativ oder 0 ist.

Formatdefinition

Um das auszuprobieren, definieren Sie einfach ein erstes eigenes Format. Dazu suchen Sie sich jene Vorgabe aus, die dem gewünschten Format am nächsten kommt und editieren diese Vorlage.



Abbildung 6.19:
Kategorie
»Benutzerdefiniert«

Klicken Sie beispielsweise auf den Eintrag »0,00«, erscheint er in der Eingabeleiste. Ersetzen Sie diesen Eintrag durch »;;;« und wählen Sie »OK«.

Der aktuellen Zelle wird dieses Format angezeigt und die darin enthaltene Zahl (bzw. der Text, das Datum oder die Uhrzeit) wird nicht mehr angezeigt.

Öffnen Sie das Dialogfeld anschließend erneut, ist das editierte Format »0,00« weiterhin vorhanden. Statt es zu überschreiben, wurde Ihr selbstdefiniertes Format »;;;« am Ende der Liste zusätzlich aufgenommen.

Mit der Schaltfläche *Löschen* können Sie ein solches selbstdefiniertes Format jederzeit wieder aus der Liste entfernen – sinnvollerweise jedoch keines der Standardformate.

Selbstdefinierte Formate sind immer untrennbar mit einer Arbeitsmappe verbunden: Speichern Sie eine Mappe, so speichert Excel auch Ihre zusätzlichen Formatdefinitionen, sodass sie nach dem späteren Öffnen der Tabelle automatisch wieder zur Verfügung stehen.



Ist dagegen eine andere Arbeitsmappe aktiv, sind nur die Standardformate verfügbar (und die in jener Mappe von Ihnen definierten Formate).

Um wirklich sinnvolle Formate definieren zu können, müssen Sie die verschiedenen Format-Symbole kennen.

- o Vor dem Komma eingesetzt, besitzt dieses Symbol nahezu keine Auswirkungen und legt nur fest, dass *mindestens* so viele Vorkommastellen angezeigt werden.

Das Format »0« zeigt daher 35 immer noch als 35 an. Das Format »000« würde jedoch stattdessen 035 anzeigen.

Wesentlich interessanter als vor dem Komma ist dieses Symbol jedoch dahinter, da es dann die *maximale* Nachkommastellenzahl festlegt.

Das Format »0«, in dem dieses Symbol also überhaupt nicht hinter dem Komma auftaucht, legt daher genau null Nachkommastellen fest: 123,45 würde entsprechend als 123 dargestellt.

Entsprechend zeigt das in der Liste enthaltene Format »0,00« ebenfalls alle Vor- und genau zwei Nachkommastellen an. 2135,1 wird als 2135,10 angezeigt und 2135,789 als 2135,79.

- . Enthält ein Format an der entsprechenden Stelle einen Tausenderpunkt ».,«, erscheint dieser später auch in der Zahlenanzeige. Zur Anzeige von zwei Nachkommastellen und zusätzlichen Tausenderpunkten benötigen Sie daher das nicht im Listenfeld enthaltene selbstdefinierte Format

0.000,00.

Angewandt auf die Zahl 23, führt es zur Anzeige von 0.023,00.

- # Allerdings wollen Sie wahrscheinlich bei den Vorkommastellen auf unnötige Nullen und Tausenderpunkte verzichten. Da der Platzhalter »0« vor dem Komma jedoch bedeutet, dass *mindestens* so viele Vorkommastellen angezeigt werden, auch wenn die Zahl selbst weniger enthält, müssen Sie dafür ein anderes Symbol verwenden und zwar »#«.

Im Gegensatz zu »0« bewirkt dieses Symbol kein Auffüllen mit Nullen, wenn die Zahl weniger Ziffern enthält.

Zur Festlegung der Nachkommastellen ist es somit ungeeignet, da mit dem Format »#,##« eine 12,6 immer noch mit nur einer Nachkommastelle angezeigt wird. Die einzige Wirkung dieses Symbols nach dem Komma: bei mehr Nachkommastellen wird wie bei »0« gerundet.

Da keine Nullen aufgefüllt werden, eignet sich dieses Symbol jedoch hervorragend zur optionalen Festlegung der Position von Tausenderpunkten. »Optional« heißt, dass diese Festlegung nur dann zur Geltung kommt, wenn die Zahl groß genug ist.

Beispielsweise wird im Format »###0,00« die Zahl 12345,678 als 12.345,68 angezeigt und die Zahl 21,4 als 21,40.

Mehrere Formate enthalten zusätzlich je ein Minuszeichen oder Texte wie »[Rot]« und zwar jeweils im zweiten Formatabschnitt, der die Anzeige negativer Zahlen festlegt.

Das Minuszeichen im negativen Abschnitt führt dazu, dass negativen Zahlen wie üblich ein »-« vorangestellt wird.

Der Text »[Rot]« bewirkt, dass die betreffende Zahl rot angezeigt wird. Allgemein: »[Schwarz]«, »[Weiß]«, »[Rot]«, »[Grün]«, »[Blau]«, »[Gelb]«, »[Magenta]«, »[Zyan]« führt jeweils zur Darstellung des Zellinhalts in der betreffenden Farbe.

[Farbe]

Beispielsweise würde das Format »[Gelb]0,00;[Blau]-0,00« positive Zahlen mit Gelb und mit zwei Nachkommastellen anzeigen, negative Zahlen dagegen in Blau und ebenfalls mit zwei Nachkommastellen.

Ein weiteres Beispiel ist das im Listenfeld enthaltene Format »###0,00 DM;[Rot]-###0,00 DM«: Der erste Abschnitt »###0,00 DM« entspricht bis auf den Zusatz » DM« dem Format »###0,00«, in dem Zahlen mit zwei Nachkommastellen und eventuell mit einem Tausenderpunkt dargestellt werden.

Zusätzlich folgt jeder Zahl der Text »DM«. Die Zahl 12345,678 wird demnach in der Form 12.345,68 DM erscheinen.

Die restlichen Symbole sind wesentlich spezieller als die bisher vorgestellten und vor allem in Spezialfällen nützlich:

Um eines der Zeichen »:«, »-«, »+«, »(«, »)«, das Leerzeichen und die Zeichenkette »DM« anzuzeigen, tippen Sie es einfach an der gewünschten Stelle der Formatbeschreibung ein.

: - + () Leer DM

Das Zeichen »\« dient dagegen als Einleitung für ein direkt folgendes Zeichen, das angezeigt werden soll (»\>« würde »>« anzeigen).

»*« führt dazu, dass das nächste Zeichen bis zum Erreichen des rechten Zellrands wiederholt wird (mit »0,00 *X« wird 123,456 als »123,46 XXXXX« angezeigt).

*

Ein in Anführungszeichen eingeschlossener Text wird unverändert ausgegeben.

Text

Das Symbol »@« ist ein Textplatzhalter: Enthält die formatierte Zelle einen Text, wird er im Format dort eingesetzt, wo sich dieses Zeichen befindet.

@

Enthält eine Zelle den Text »Hallo«, so zeigt das Format »! @ !« daher »! Hallo !« an.

[Farben] Mit einem Ausdruck »[Farbe n]« können Sie eine Farbe anhand Ihres Farbwerts definieren. n muss dabei ein Farbwert zwischen 0 und 56 sein.

[Bedingung Wert] Mit den Vergleichsoperatoren »<« (kleiner), »>« (größer), »=« (gleich), »<=« (kleiner oder gleich), »>=« (größer oder gleich), »<>« (ungleich) können Sie mehrere durch Semikola getrennte bedingte Formatabschnitte definieren.

Beispielsweise vergleicht das Format »[>50]0;[<=50]0,00« den Zelleninhalt mit der Zahl 50. Ist die Bedingung >50 erfüllt, enthält die Zelle also einen Wert, der größer als 50 ist, ist das Format »0« gültig. Ist dagegen die Bedingung <=50 erfüllt, wird das Format »0,00« angewendet.

6.5 Schriftformate

6.5.1 Schriftarten und -farben

Für Schriftarten und -größen, Schriftattribute wie »Fett« oder »Kursiv«, die Textfarbe und ähnliches ist das Register *Schrift* des Befehls **FORMAT|ZELLEN...** zuständig.

Selektieren Sie eine Zelle oder einen Bereich, wählen Sie diesen Befehl und aktivieren Sie dieses Register (Abbildung 6.20).

In diesem Dialogfeld sind zunächst jene Einstellungen selektiert, die momentan auf die betreffende Zelle bzw. den Bereich zutreffen.

Ist ein Bereich selektiert, dieser jedoch uneinheitlich formatiert, erhalten Sie verständlicherweise keine solche Vorgabe.

Egal, welche Einstellungen Sie nun selektieren, im Kasten *Vorschau* zeigt Ihnen Excel in jedem Fall, wie ein damit formatierter Text dargestellt werden kann.

In den drei oberen Listefeldern können Sie vollkommen unabhängig voneinander die Schriftart, den Schriftschnitt und den Schriftgrad festlegen.

Mit den Einträgen im Listefeld *Unterstreichung* können Sie Zelleninhalte einfach oder doppelt unterstreichen.



Abbildung 6.20:
Das Register
»Schrift«

Mit den Kontrollkästchen der Rubrik *Effekte* können Sie einen Strich mit- oder ohne den Text legen beziehungsweise ihn hoch- oder tiefgestellt anzeigen lassen.

Im Listenfeld *Farbe* können Sie die Textfarbe wählen (Abbildung 6.21).

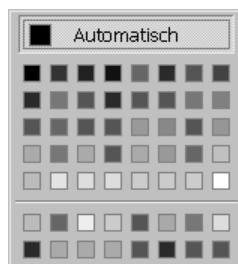


Abbildung 6.21:
Farbwahl

Mit der Voreinstellung »Automatisch« verwendet Excel jene Farbe, die in der Windows-Systemsteuerung im Register *Darstellung* für das Bildelement »Fenster« unter *Schriftfarbe* eingestellt ist (normalerweise Schwarz).

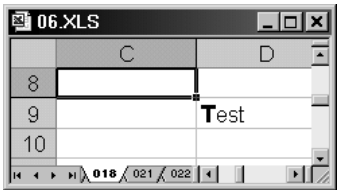
Standardfarbe

Ändern Sie diese Einstellung in Grün, werden sofort die Inhalte aller Zellen, die das Format »Automatisch« besitzen, grün statt schwarz angezeigt – üblicherweise also alle Zellen, da jede Zelle zunächst das Farbattribut »Automatisch« besitzt, solange ihr keine andere Farbe zugewiesen wird.



Außer kompletten Zellen können Sie auch einzelne Buchstaben oder Wörter formatieren. Dazu selektieren Sie beispielsweise in einer Zelle, die den Text »Test« enthält, das Zeichen »T«, wählen **FORMAT|ZELLEN...** und stellen die für diesen Buchstaben gewünschten Formatierungen ein (Abbildung 6.22).

Abbildung 6.22:
Zellteil formatieren



Nun ein praktisches Beispiel. Unsere kleine Buchführung ist höchst unübersichtlich, was jedoch sehr leicht durch geeignete Schriftformatierungen geändert werden kann.

Um beispielsweise die obersten Rubrikenüberschriften »Einnahmen« und »Ausgaben« deutlich hervorzuheben, sollten Sie sie fett formatieren und eine größere Schrift verwenden.

Dazu selektieren Sie B1 und D1 und wählen im *Schriftart*-Dialogfeld unter *Schriftschnitt* die Einstellung »Fett« und unter *Schriftgrad* die Größe 12 (Abbildung 6.23).

Abbildung 6.23:
Anpassung der
Zeilenhöhe

	A	B	C	D	E	F	G	
1		Einnahmen		Ausgaben				
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	Gv
3		230,00	3859,50	2,00	19,80	284,00	184,00	
4		1637,58		4,00	53,20	76,50	396,00	
5				3,70		73,00		
6						68,90		
7								
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	
11								
12					Netto	MWSt		
13		- Steuerpflichtige Umsätze mit 16%			3327,16	532,34		
14		- Steuerpflichtige Umsätze mit 70%			1745,40	122,18		

Schauen Sie sich die Abbildung genau an, dann werden Sie feststellen, dass Excel die Höhe von Zeile 1 automatisch an die nun größer – und damit sowohl höher als auch breiter – gewordene Schrift anpasste: Zeile 1 ist jetzt minimal höher als die restlichen Zeilen.

Allerdings betrifft diese Anpassung nur die Zeilenhöhe, nicht die Breite der Spalten!

Und da größere Schriften auch in der Breite mehr Platz benötigen, kann es passieren, dass Zellinhalte, die zuvor problemlos in eine Zelle passten, nun nicht mehr vollständig sichtbar sind. Wie dieses Problem zu lösen ist, erfahren Sie im Kapitel 6.6, »Spaltenbreite und Zeilenhöhe«.



Sie können Schriftarten oder Attribute für einen selektierten Bereich problemlos *teilweise* ändern.

Teilzuweisungen

Angenommen, Sie wollen alle in den ersten beiden Zeilen enthaltenen Überschriften grün darstellen. Dann selektieren Sie den Bereich A1:I2 und aktivieren wieder das *Schrift*-Register des *Zellen*-Dialogfelds (Abbildung 6.24).

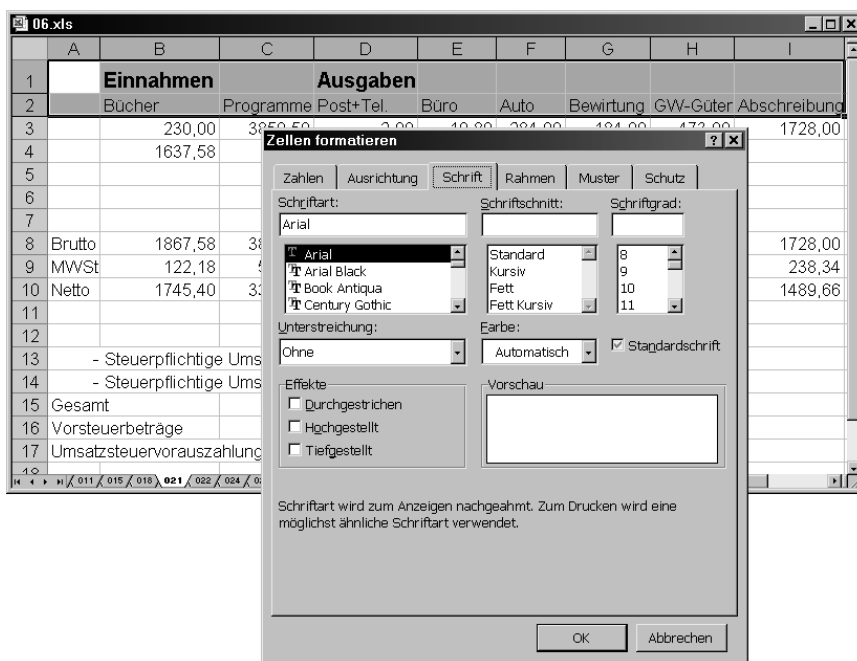


Abbildung 6.24:
Uneinheitliche
Bereichsformatie-
rung

Unter *Schriftgrad* ist keine Einstellung vorselektiert, da es keinen für den gesamten Bereich einheitlichen Schriftgrad gibt. B1 und D1 verwenden den Schriftgrad 12, alle anderen Zellen des Bereichs den Grad 10.

Sie können nun beliebige andere Einstellungen treffen, beispielsweise die Textfarbe Grün selektieren und das Dialogfeld mit OK schließen. Die gewählte Textfarbe wird wie gewünscht allen Zellen des Bereichs zugewiesen.

Beachten Sie, dass der uneinheitliche Schriftgrad dabei unverändert erhalten bleibt, solange Sie keine Einstellung im zugehörigen Listenfeld selektieren!

Um unsere Buchführungs-Tabelle noch ein wenig übersichtlicher zu gestalten, sollten wir darin alle Bereiche, die Überschriften enthalten, fett formatieren (Abbildung 6.25).

Abbildung 6.25:
Format.xls,
»Tabelle3«

	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8		Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00
9		MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24
10		Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76
11									
12					Netto	MWSt			
13						3327,16	532,34		
14						1745,40	122,18		
15		Gesamt				5072,56	654,52		
16		Vorsteuerbeträge					462,95		
17		Umsatzsteuervorauszahlung					191,57		
18									



Es genügt, jene Zelle zu formatieren, in der sich der betreffende Text oder die Zahl befindet. Bei überlängten Texten müssen Sie Zellen, die zur Anzeige mitbenutzt werden, nicht mitformatieren!

Daher genügt es, die Zelle A17 fett zu formatieren, um den gesamten darin enthaltenen Text »Umsatzsteuervorauszahlung« fett darzustellen, obwohl für die Anzeige dieses Textes die beiden benachbarten Zellen B17 und C17 mitbenutzt werden.

**Standardschrift
zuweisen**

Wenn Sie das Kontrollkästchen *Standardschrift* aktivieren, werden alle Änderungen gleichzeitig rückgängig gemacht. Der aktiven Zelle bzw. dem selektierten Bereich wird wieder die *Standardschrift* im *Standardschriftstil* in der *Standardgröße* mit den *Standardattributen* zugewiesen.

**Standard
definieren**

All diese Standards können Sie selbst definieren!



Beispielsweise können Sie den Standard für eine einzige Tabelle definieren, indem Sie das Standard-Druckformat dieser Tabelle ändern (siehe Kapitel 6.10, »Formatvorlagen«), oder für alle neu anzulegenden Arbeitsmappen, indem Sie eine neue Tabelle erzeugen, deren Standard-Druckformat nach Ihren Wünschen verändern und die Tabelle als Mustervorlage unter dem Namen TABELLE.XLT im Verzeichnis XLSTART speichern (siehe Kapitel 8.8, »Arbeitsmappen als Vorlagen verwenden und automatisch öffnen«).

Das Standard-Druckformat dieser Tabelle ist ab jetzt die Vorlage, die Excel für die Standardschrift in *allen neu erzeugten Tabellen* verwendet!

Die folgenden Symbole der Formatleiste und zusätzlich einige Symbole der Kategorie *Format* des *Anpassen*-Dialogfelds vereinfachen die Zuweisung von Schriftformaten:

Weist allen Texten im selektierten Bereich die im zugehörigen Listenfeld ausgewählte Punktgröße zu



Weist den selektierten Zellen den nächstgrößeren Schriftgrad zu. Enthält der Bereich Zellen mit unterschiedlichem Schriftgrad, wird allen Zellen die gegenüber der größten verwendeten Schrift nächstgrößere Punktgröße zugewiesen.



Weist den selektierten Zellen den nächstkleineren Schriftgrad zu. Enthält der Bereich Zellen mit unterschiedlichem Schriftgrad, wird allen Zellen die gegenüber der größten verwendeten Schrift nächstkleinere Punktgröße zugewiesen.



Weist allen Texten im selektierten Bereich die im zugehörigen Listenfeld ausgewählte Schriftart zu



Formatiert die selektierten Zellen fett bzw. hebt diese Formatierung wieder auf



Formatiert die selektierten Zellen kursiv bzw. hebt diese Formatierung wieder auf



Unterstreicht die selektierten Zellen bzw. hebt diese Formatierung wieder auf



Unterstreicht die selektierten Zellen mit einem Doppelstrich bzw. hebt diese Formatierung wieder auf



Streicht die selektierten Zellen durch (Strich auf mittlerer Schrifthöhe) bzw. hebt diese Formatierung wieder auf



Weist die im zugehörigen Listenfeld gewählte Schriftfarbe den selektierten Zellen zu



Ändert bei jedem Anklicken die Schriftfarbe der selektierten Zellen



6.5.2 Druckerschriftarten und TrueType-Schriften

Jeder Drucker besitzt eine oder mehrere eingebaute Schriftarten, die er meist in mehreren Varianten wie »Fett« oder »Kursiv« drucken kann. Wenn Sie Zellen nicht mit einer dieser Druckerschriftarten formatieren, sondern mit einer Schrift, die Ihr Drucker nicht beherrscht, dann verwendet Excel ein-

Druckerschriften

fach eine Druckerschriftart, die der Bildschirmschrift möglichst nahe kommt. Das bedeutet, dass sich der Ausdruck ganz erheblich von der Bildschirmdarstellung unterscheiden kann!

Daher ist es sinnvoll, nur die bereits *in Ihrem Drucker eingebauten Schriftarten* zu verwenden – oder die weiter unten erläuterten TrueType-Schriften.

TrueType-Schriften Auf TrueType-Schriften wird im Listenfeld durch entsprechende Hinweise zur selektierten Schrift am unteren Rand der Dialogbox hingewiesen.

Das Besondere an TrueType-Schriften ist, dass sie skalierbar sind und Bildschirmdarstellung und Ausdruck weitestgehend übereinstimmen (das berühmte WYSIWYG-Schlagwort: »what you see is what you get«, was im Falle der TrueType-Schriften auch einigermaßen zutrifft).

Skalierbare Schriften Skalierbar bedeutet, dass diese Schriften nicht wie andere als Bitmuster gespeichert sind und Windows für jeden Schriftgrad ein eigenes Muster besitzen muss, sondern ein Schriftschema genügt, das Regeln zum Aufbau der Schrift enthält, die Windows benutzt, um die betreffende Schrift immer gleich aufzubauen – unabhängig von der Größe!

Daraus resultiert, dass *alle* TrueType-Schriften zwischen acht und 72 Punkten variierbar sind, im Gegensatz zu den restlichen Schriften, bei denen Ihr Spielraum meist erheblich kleiner ist.

Zusätzlich werden TrueType-Schriften in extremen Fällen auf dem Bildschirm feiner dargestellt, zum Beispiel bei sehr großen Schriften.

Geschwindigkeit TrueType-Schriften besitzen einen Nachteil: Die Skalierbarkeit führt dazu, dass diese Schriften beim Bildschirmaufbau in der jeweils benötigten Größe berechnet werden, im Gegensatz zu normalen Schriften, wo ein fest gespeichertes Bitmuster ausgegeben wird.

Daraus resultiert, dass der Aufbau von TrueType-Schriften und damit das Scrollen des Bildschirms vor allem bei langsameren Rechnern deutlich langsamer als bei nicht skalierbaren Schriften erfolgt!

Fassen wir zusammen:

➡ Bei normalen Schriften – ohne zugehöriges Symbol im *Schriftart*-Listenfeld – ist der Ausdruck ein Glücksspiel; eventuell wird langsam gedruckt und die Druckqualität ist schlecht; möglicherweise verwendet Excel sogar statt der gewählten Schrift eine Druckerschriftart, die mit der verwendeten Bildschirmschrift nicht so ganz übereinstimmt.

- ➔ Letzteres gilt auch für Druckerschriftarten, dafür ist jedoch gewährleistet, dass der Ausdruck optimale Schriftqualität besitzt und zügig verläuft.
- ➔ Nur bei TrueType-Schriften können Sie sicher sein, dass der Ausdruck weitestgehend der Bildschirmdarstellung entspricht.

Zudem ist nur bei diesen frei skalierbaren Schriftarten gewährleistet, dass eine extrem große Punktgröße wie 72 Punkt auch tatsächlich verfügbar ist, wenn Sie sie benötigen, und Sie nicht gezwungenermaßen für eine Überschrift auf eine andere Schriftart umsteigen müssen als jene, die im Rest der Tabelle verwendet wird.

Dafür erfolgt der Bildschirmaufbau langsamer, da gegenüber anderen Schriften zusätzliche Berechnungen erforderlich sind.

6.6 Spaltenbreite und Zeilenhöhe

Excel passt die Zeilenhöhe automatisch an die Höhe der in der betreffenden Zeile verwendeten Schrift an. Allerdings wird die Spaltenbreite nicht an die Breite dieser Schrift angepasst!

Dadurch werden Texte bei Verwendung breiter Schriften anschließend oft nicht mehr vollständig dargestellt (außer, die rechts benachbarten Zellen sind leer und werden daher zur Anzeige mitverwendet).

Glücklicherweise können Sie die Breite und Höhe der einzelnen Zeilen und Spalten beliebig verändern.

6.6.1 Mit der Maus

Bewegen Sie den Mauszeiger zu den Zeilennummern am linken Fensterrand, genauer: zu der dünnen Begrenzungslinie zwischen zwei Zeilennummern, zum Beispiel zwischen den Zeilennummern 10 und 11, dann verändert sich der Mauscursor und wird zu einer Art Doppelpfeil. Drücken Sie nun die linke Maustaste, so wird die aktuelle Höhe der oberen Zeile angezeigt. Ziehen Sie die Zeilenbegrenzung nach oben, wird die Zeilenhöhe verringert bzw. beim Ziehen nach unten erhöht, wobei Excel die neue Zeilenhöhe ständig anzeigt.

Sobald Sie die Maustaste loslassen, ist die neue Zeilenhöhe eingestellt. Und zwar für Zeile 10, da sich das Ziehen an einer solchen Linie immer auf jene Zeile auswirkt, deren *untere* Begrenzung die Linie darstellt.

Analog dazu verändern Sie die Spaltenbreite, indem Sie die Begrenzungslinie zwischen zwei Spaltenbuchstaben nach links oder rechts ziehen, genauer zwischen dem Spaltenbuchstaben dieser und der vorhergehenden Spalte.

Optimale Höhe Mit einem Doppelklick auf eine Begrenzungslinie zwischen Zeilennummern lösen Sie für die betreffende Zeile die Funktion »Optimale Höhe« aus.

Sie verringert die Höhe der betreffenden Zeile so weit, dass der höchste Zellinhalt der betreffenden Zeile gerade noch vollständig sichtbar ist, ohne dass die Schrift am oberen Rand abgeschnitten wird.

Mit einem Doppelklick auf eine Begrenzungslinie zwischen zwei Spaltenbuchstaben lösen Sie die gleiche Funktion für die Spaltenbreite der linken Spalte aus.

**Mehrere Zeilen/
Spalten** Um mehrere Zeilen oder Spalten gleichzeitig zu manipulieren, selektieren Sie sie zunächst komplett und verändern danach die Höhe *irgendeiner* Zeile bzw. die Breite *irgendeiner* Spalte des selektierten Bereichs, entweder durch Ziehen an der Trennlinie oder (optimale Höhe/Breite) per Doppelklick darauf. Die Änderung wirkt sich automatisch auf alle Zeilen/Spalten des Bereichs aus.



Die Funktion »Optimale Breite« ist mit Vorsicht zu genießen. Wenden wir sie auf die Spalten A bis I unserer Buchführung an, indem wir diese Spalten selektieren und auf irgendeine Begrenzungslinie zwischen zwei Spaltenbuchstaben doppelklicken, erhalten wir folgendes Resultat (Abbildung 6.26):

Abbildung 6.26:
Nicht gerade optimale Breite

	A	B	C	D	E
1		Einnahmen		Ausgaben	
2		Bücher	Programme	Post+Tel.	Büro
3		230,00	3859,50	2,00	19,80
4		1637,58		4,00	53,20
5				3,70	
6					
7					
8	Brutto	1867,58	3859,50	9,70	73,00
9	MWSt	122,18	532,34	0,00	10,07
10	Netto	1745,40	3327,16	9,70	62,93
11					
12					Netto
13	- Steuerpflichtige Umsätze mit 16%				3327,16
14	- Steuerpflichtige Umsätze mit 7%				1745,40

Excel orientiert sich am breitesten in Spalte A enthaltenen Text, ignoriert dabei jedoch eventuell rechts benachbarte leere Zellen, die zur Textdarstellung mitverwendet werden könnten.

Entsprechend ist Spalte A nun leicht überdimensioniert und erfordert eine manuelle Nachbearbeitung, zum Beispiel mit dem gleich erläuterten Menübefehl **FORMAT|SPALTE|OPTIMALE BREITE BESTIMMEN...**

6.6.2 Menübefehle

Die Unterbefehle der Befehle **FORMAT|ZEILE** und **FORMAT|SPALTE** wirken sich immer auf die aktuelle Zeile/Spalte aus, also auf jene, in der sich momentan der Zellcursor befindet.

Um mehrere Zeilen/Spalten gleichzeitig mit diesen Befehlen zu manipulieren, können Sie sie zuvor komplett selektieren. Es genügt jedoch, stattdessen einen Zellbereich zu selektieren, der sich über die gewünschten Spalten/Zeilen erstreckt, beispielsweise den Bereich A1:B1, wenn Sie die Spalten A und B manipulieren wollen.

FORMAT|SPALTE|BREITE... öffnet ein Dialogfeld, in dem die aktuelle Spaltenbreite in Punkten vorgegeben wird und Sie eine beliebige andere Breite eingeben können.

Entsprechend gibt **FORMAT|ZEILE|HÖHE...** den Wert für die aktuelle Zeilenhöhe vor und ermöglicht Ihnen die Eingabe eines anderen Wertes.

FORMAT|SPALTE|STANDARDBREITE... öffnet ein Dialogfeld, in dem die aktuelle Spalten-Standardbreite vorgegeben wird, beispielsweise 10,78. Geben Sie einen anderen Wert wie 15 ein, wirkt sich das auf alle Spalten aus, die die zuvor angezeigte Standardbreite von 10,78 besitzen. All diesen Spalten wird dadurch die neue Standardbreite 15 zugewiesen.

FORMAT|SPALTE|OPTIMALE BREITE BESTIMMEN... stellt die optimale Breite einer Spalte ein und zwar im Gegensatz zu der entsprechenden Mausaktion (Doppelklick auf eine Spaltentrennlinie) *ausgehend vom Inhalt der momentan in dieser Spalte selektierten Zellen!*

Entsprechend lässt sich die in der letzten Abbildung gezeigte überdimensionierte Spalte A optimal anpassen, indem Sie die drei Zellen A8:A10 selektieren und diesen Befehl wählen (Abbildung 6.27).

	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8		Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00
9		MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24
10		Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76
11									
12					Netto	MWSt			
13		- Steuerpflichtige Umsätze mit 16%			3327,16	532,34			
14		- Steuerpflichtige Umsätze mit 7%			1745,40	122,18			
15		Gesamt			5072,56	654,52			
16		Vorsteuerbeträge				462,95			
17		Umsatzsteuervorauszahlung				191,57			

Abbildung 6.27:
Format.xls,
»Tabelle4«

*Zeilen und
Spalten aus-/
einblenden*

Die Unterbefehle AUSBLENDEN und EINBLENDEN der Befehle FORMAT|ZEILE bzw. FORMAT|SPALTE ändern die Zeilenhöhe/Spaltenbreite in einer sehr speziellen Art und Weise: AUSBLENDEN stellt die Zeilenhöhe/Spaltenbreite 0 ein – die Zeile/Spalte verschwindet!

Selektieren Sie beispielsweise den Bereich B1:C1 oder irgendeinen anderen Bereich, der Zellen von Spalte B und C enthält, und wählen Sie FORMAT|SPALTE|AUSBLENDEN, so werden diese beiden Spalten unsichtbar (Abbildung 6.28).

Abbildung 6.28:
Spalten ausblenden

	A	D	E	F	G	H	I
1		Ausgaben					
2		Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		2,00	19,80	284,00	184,00	473,00	1728,00
4		4,00	53,20	76,50	396,00		
5		3,70		73,00			
6				68,90			
7							
8	Brutto	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	9,70	62,93	433,10	500,00	407,76	1489,66
11							
12			Netto	MWSt			
13		- Steuerpflichtig	3356,09	503,41			
14		- Steuerpflichtig	1745,40	122,18			
15	Gesamt		5101,49	625,59			
16	Vorsteuerbeträge			462,95			
17	Umsatzsteuervorauszahlung			162,64			
18							

Dieses AUSBLENDEN kann sehr nützlich sein, wenn Sie ein Blatt ausdrucken und jemandem zeigen wollen, darin jedoch eine Zeile/Spalte vorkommt, die Auskunft über Ihr Einkommen oder den Gewinn Ihrer Firma enthält und nicht ausgedruckt werden soll: Sie verbergen die Zeile/Spalte, drucken das Blatt aus und machen sie anschließend wieder sichtbar.

Mit EINBLENDEN ist das allerdings nur möglich, wenn Sie den Cursor inzwischen nicht in eine andere Zeile/Spalte bewegt haben!

Schauen Sie sich noch einmal die Abbildung an. Die Spalten B und C sind verschwunden. Bewegen Sie den Cursor von links nach rechts, gelangen Sie von Spalte A direkt zu Spalte D. Es ist nicht mehr möglich, in Spalte B oder C zu gelangen, was jedoch Voraussetzung ist, um die Höhe dieser Spalten zu ändern.

Die einfachste Lösung: Sie selektieren einen Bereich, der *unter anderem* auch die unsichtbaren Spalten B und C enthält, zum Beispiel A1 bis D1. Anschließend wählen Sie FORMAT|SPALTE|EINBLENDEN.



Eine Alternative dazu bietet der Befehl BEARBEITEN|GEHE ZU... (Kurz Taste **F5**), der ein Dialogfeld öffnet, in dem Sie eine Zellreferenz angeben können. Geben Sie darin B1:C1 ein, springt der Zellcursor zur betreffenden Position und selektiert den angegebenen Bereich, was jedoch nur an einem waagerechten Strich erkennbar ist, da die beiden Spalten ja noch ausgeblendet sind.

Da Sie sich nun jedoch tatsächlich in diesen unsichtbaren Spalten befinden, können Sie sie mit dem Befehl EINBLENDEN problemlos wieder sichtbar machen.

6.7 Text- und Zahlenausrichtung

Der Befehl Format FORMAT|ZELLEN... öffnet im Register *Ausrichtung* folgendes Dialogfeld (Abbildung 6.29).

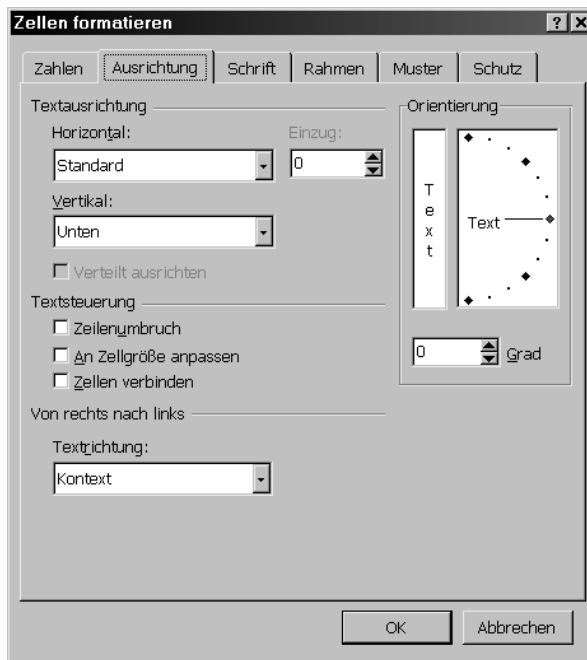


Abbildung 6.29:
Ausrichtungs-
varianten

Der im Listenfeld *Horizontal* gewählte Eintrag legt die horizontale Textausrichtung fest, bei der sich die in der Zelle enthaltenen Zeichen wie gewohnt nebeneinander befinden.

»Standard« ist die Standardausrichtung von Excel, bei der Text linksbündig ausgerichtet wird (also am linken Zellrand beginnt) und Zahlen rechtsbündig.

All diese Zellformate werden wieder den momentan selektierten Zellen zugewiesen und umgekehrt: Bei Anwahl des Befehls ist jene Option vorselektiert, die auf die momentan aktive Zelle bzw. den selektierten Bereich zutrifft. Die folgende Abbildung demonstriert die Auswirkungen der verschiedenen horizontalen bzw. vertikalen Varianten (Abbildung 6.30).

Abbildung 6.30:
Auswirkungen

	A	B	C	D	E	F	G
1	Option, Horizontal		Vorher			Nachher	
2	Standard		Test			Test	
3	Linksbündig		Test			Test	
4	Zentriert		Test			Test	
5	Rechtsbündig		Test			Test	
6	Ausfüllen		Test			TestTestTest	
7	Blocksatz		Dies ist ein Test des Blocksatzes			Dies ist ein Test des Blocksatzes	
8	Über Auswahl zentrieren		Test			Test	
9							
10	Option, Vertikal		Vorher			Nachher	
11	Oben		Test			Test	
12	Mitte		Test			Test	
13	Unten		Test			Test	
14	Aufteilen		Test			Test	
15							
16	Sonstiges		Vorher			Nachher	
17	Zeilenumbruch		Dies ist ein Test des Zeilenumbruchs			Dies ist ein Test des Zeilenumbruchs	
18	An Zellgröße anpassen		Anpassen an Zellgröße			Anpassen an Zellgröße	
19	Zellen verbinden		Zellen verbinden			Zellen verbinden	
20							

Horizontale Ausrichtung

Die horizontalen Varianten wie »Zentriert« oder »Rechts« sind weitgehend selbsterklärend und legen fest, ob die Ausrichtung am rechten Zellrand erfolgt, am linken oder in der Mitte der Zelle.

Die erklärungsbedürftigen Ausnahmen:

- ➔ Bei linksbündiger Ausrichtung können Sie mit dem Drehfeld *Einzug* den Abstand zum linken Zellrand festlegen (je höher der Wert, desto größer ist der Abstand). Durch Drehen an diesem Wert wird übrigens automatisch die linksbündige Ausrichtung selektiert, wenn das noch nicht der Fall ist.
- ➔ *Ausfüllen* wiederholt einen Zelleintrag bis zum Rand der Zelle.
- ➔ *Über Auswahl zentrieren* zentriert Text in der Mitte des momentan selektierten Bereichs. Vor der Wahl dieser Ausrichtung wurde F8:G8 selektiert, sodass sich der zuvor in F8 enthaltene Text nun in der Mitte der beiden Zellen befindet.

- ➔ Beachten Sie den Unterschied zwischen *Blocksatz* und dem Kontrollkästchen *Zeilenumbruch*: *Blocksatz* stellt Zellinhalte im selektierten Bereich (das können auch mehrere Zellen sein) im Blocksatz dar und fügt dazu Leerzeichen zwischen Wörtern ein. Zusätzlich findet automatisch ein Zeilenumbruch statt, wenn der Text breiter als die aktuelle Selektion ist. Das Kontrollkästchen *Zeilenumbruch* kann dagegen unabhängig von den restlichen Optionen aktiviert oder deaktiviert werden und ist nützlich, wenn für einen Text nicht ausreichend Platz zur Verfügung steht, da gegebenenfalls einfach die Spaltenbreite und der Text danach untereinander dargestellt wird. Im Gegensatz zum Blocksatz werden dabei jedoch keine Leerzeichen eingefügt.
- ➔ Um die einzelnen Zeichen einer Zelle unter- statt nebeneinander anzuordnen (siehe Bild 6.30, unterer Teil), klicken Sie unter *Orientierung* auf das etwas groß geratene Kontrollkästchen mit dem vertikal ausgerichteten Text *Text*.

*Vertikale
Ausrichtung*

Statt dieser 90-Grad-Verdrehung können Sie durch Ziehen am rechts daneben abgebildeten Skalenzeiger beliebige andere Winkel einstellen. Tun Sie das, werden übrigens nicht nur die Zellinhalte um den betreffenden Winkel rotiert, sondern auch die im folgenden Kapitel erläuterten Zellrahmen (Abbildung 6.31).

Zellen rotieren

Buchfgr.xls										
	A	B	C	D	E	F	G	H	I	J
1		Einnahmen		Ausgaben						
		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung	
2										
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00	
4		1637,58		4,00	53,20	76,50	396,00			
5				3,70		73,00				
6						68,90				
7										
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00	
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34	
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66	
11										

Abbildung 6.31:
Zellinhalte rotieren

Egal, ob der Zellinhalt mehr oder weniger vertikal ausgerichtet ist, anschließend können Sie in jedem Fall mit den Optionen des Listfelds *Vertikal* festlegen, ob die nun vertikale Ausrichtung am oberen Zellrand, am unteren oder dazwischen erfolgen soll (siehe Abbildung 6.30, unterer Teil).

Sie können diese vertikale Ausrichtung übrigens mit den horizontalen Optionen *Standard*, *Links*, *Zentriert* und *Rechts* kombinieren, um auch bei vertikaler Ausrichtung den Text an den rechten oder linken Zellrand oder in die Mitte zu schieben.

Wollen Sie wieder die normale horizontale Ausrichtung aktivieren, deaktivieren Sie das Kontrollkästchen *Text* einfach durch erneutes Anklicken.

An Zellgröße anpassen

Das Kontrollkästchen *An Zellgröße anpassen* verkleinert den Zellinhalt, damit er in die aktuelle Spalte passt – und zwar ohne die zugewiesene Schriftgröße zu verändern, sodass diese nach Deaktivierung des Kontrollkästchens sofort wiederhergestellt wird!

Zellen verbinden

Zellen verbinden verbindet die zuvor selektierten Zellen (F19 und F20) zu einer einzigen überbreiten Zelle (F19).

Von rechts nach links orientierte Sprachen

Das Listenfeld *Textausrichtung* ermöglicht es zusammen mit dem Register *International* des Befehls EXTRAS|OPTIONEN..., ein Excel-Blatt an Sprachen anzupassen, deren Schreibweise von rechts nach links orientiert ist.

Aktivieren Sie *Von rechts nach links*, wird scheinbar nur die Optik neu angelegter(!) Blätter beeinflusst, da alle möglichen Bedienungselemente plötzlich gespiegelt sind: Zeilennummern und Blattnamen befinden sich anschließend nicht mehr links, sondern rechts (erneutes Anklicken stellt den ursprünglichen Zustand wieder her), Kommentar-Dreiecke erscheinen in der linken oberen Zellecke statt in der rechten etc.

Das Listenfeld *Textausrichtung* beeinflusst im Gegensatz dazu nicht das gesamte Blatt, sondern nur die zuvor selektierten Zellen.



Unschöne Zeilenumbrüche können Sie nachträglich durch das Einfügen von Bindestrichen beeinflussen. Den höchst unschönen Umbruch »Zeilenumbr« und »uchs« in Zelle F17 korrigieren Sie beispielsweise, indem Sie das Wort »Zeilenumbruchs« in »Zeilen-umbruchs« ändern. Excel korrigiert den Umbruch daraufhin automatisch in »Zeilen-« und »umbruchs«.

Nun ein Beispiel für die praktische Anwendung: Es mag Geschmackssache sein, aber ich persönlich würde es vorziehen, wenn die in Zeile 2 unserer Buchführung enthaltenen Texte ebenso wie die Zahlen darunter rechtsbündig ausgerichtet wären.

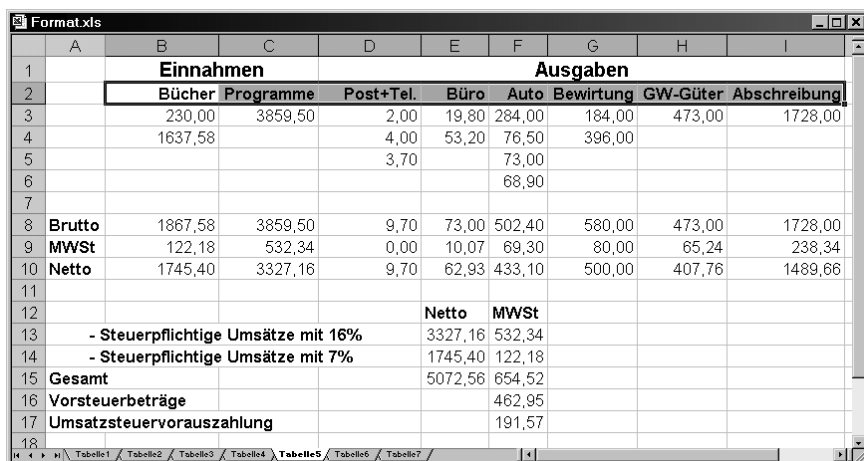
Dazu selektieren Sie den Bereich B2:I2, öffnen das *Ausrichtungs*-Dialogfeld und aktivieren unter *Horizontal* die Option *Rechts* (Abbildung 6.32).

Zusätzlich wurde die Option *Über Auswahl zentrieren* verwendet, um die in B1 und D1 enthaltenen Texte in der Mitte der zuvor selektierten Bereiche B1:C1 bzw. D1:I1 zu zentrieren.

Verschiedene Symbole der Formatleiste und des gleichnamigen Befehlskategorie im *Anpassen*-Dialogfeld vereinfachen die Ausrichtung.

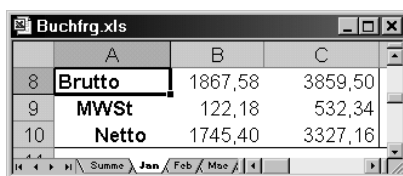


Beispielsweise rückt das abgebildete Symbol den Inhalt der zuvor markierten Zellen leicht ein. Jedesmal, wenn Sie darauf klicken, wird die Einrückung vergrößert (Abbildung 6.33).



	A	B	C	D	E	F	G	H	I
1		Einnahmen			Ausgaben				
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70			73,00		
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12					Netto	MWSt			
13		- Steuerpflichtige Umsätze mit 16%			3327,16	532,34			
14		- Steuerpflichtige Umsätze mit 7%			1745,40	122,18			
15	Gesamt				5072,56	654,52			
16	Vorsteuerbeträge					462,95			
17	Umsatzsteuervorauszahlung					191,57			
18									

Abbildung 6.32:

Format.xls,
»Tabelle5«


	A	B	C
8	Brutto	1867,58	3859,50
9	MWSt	122,18	532,34
10	Netto	1745,40	3327,16

Abbildung 6.33:

Zellinhalte
einrücken

Dieses Symbol hebt die Einrückung genau umgekehrt schrittweise wieder auf.



Richtet Text linksbündig aus bzw. hebt diese Formatierung wieder auf



Zentriert Text in der Zellmitte bzw. hebt diese Formatierung wieder auf



Richtet Text rechtsbündig aus bzw. hebt diese Formatierung wieder auf



Erzeugt Blocksatz – gleichzeitige Ausrichtung am rechten *und* linken Zellrand – durch das Einfügen zusätzlicher Leerzeichen zwischen den einzelnen Wörtern bzw. hebt diese Formatierung wieder auf



Ist eine Kombination der Optionen *Zellen verbinden* und *Über Auswahl zentrieren*. Ist ein Bereich benachbarter Zellen selektiert, beispielsweise A1:A3 und enthält die Zelle ganz links, also A1, einen Text wie »Überschrift«, wird dieser Text über all diese Zellen zentriert und A1:A3 zu einer einzigen Zelle vereinigt.



Um daraus wieder drei einzelne Zellen zu machen, müssen Sie die Zelle erneut selektieren und im Register *Ausrichtung* die betreffenden Optionen deaktivieren.



Richtet Text entsprechend der linken vertikalen Ausrichtungsvariante aus und passt die Zeilenhöhe entsprechend an bzw. macht die Vertikalausrichtung wieder rückgängig



Richtet Text entsprechend der mittleren vertikalen Ausrichtungsvariante aus – durch Rotation um 90 Grad gegen den Uhrzeigersinn gedreht – und passt die Zeilenhöhe entsprechend an bzw. macht die Vertikalausrichtung wieder rückgängig



Richtet Text entsprechend der rechten vertikalen Ausrichtungsvariante aus – durch Rotation um 90 Grad im Uhrzeigersinn gedreht – und passt die Zeilenhöhe entsprechend an bzw. macht die Vertikalausrichtung wieder rückgängig

6.8 Rahmen

Jede Zelle kann ihren eigenen Rahmen besitzen. Darunter sind bereits eine einfache Linie am oberen, unteren, linken oder rechten Rand der Zelle zu verstehen oder beliebige Kombinationen dieser vier möglichen Randbegrenzungen, zum Beispiel Linien an allen vier Rändern, was einen geschlossenen Rahmen um die Zelle oder den Bereich herum ergibt.

Zusätzlich können Sie die Strichstärke und die Farbe dieser Linien bestimmen. Für all diese Formatiermöglichkeiten ist das Register *Rahmen* des Befehls `FORMAT|ZELLEN...` zuständig (Abbildung 6.34).

Abbildung 6.34:
Rahmen-
formatierungen



Dieses Dialogfeld bietet sehr vielfältige Möglichkeiten. Wichtig ist, dass

- ➔ *Keine* alle Rahmenlinien gleichzeitig entfernt, egal auf welche Weise sie entstanden
- ➔ und Sie vor dem Klicken auf irgendeinen Rahmenknopf zunächst unter *Art* auf den gewünschten Linientyp klicken und unter *Farbe* die Linienfarbe auswählen, wobei »Automatisch« die Standardtextfarbe von Windows verwendet (siehe Kapitel 6.5.1, »Schriftarten und -farben«).

Außen legt einen Rahmen um alle vier Ränder der aktuellen Zelle herum. Ist ein Bereich selektiert, wird der Rahmen stattdessen um den gesamten Bereich herum gelegt und nicht etwa um jede einzelne Zelle.

Innen ist nur wählbar, wenn ein Bereich selektiert ist, und zeichnet Rahmenlinien *zwischen* den einzelnen Zellen ein. Klicken Sie auf *Außen* und (!) auf *Innen*, wird beides kombiniert, sodass sich an jedem Zellrand eine Rahmenlinie befindet.

Klicken Sie auf *Außen* und/oder *Innen*, werden unter *Rahmen* jene Schaltflächen eingedrückt, die die betreffenden Teillinien symbolisieren.

All diese Schaltflächen sind Umschaltflächen und können auch separat benutzt werden: Klicken Sie auf die oberste Schaltfläche, wird am oberen Zellrand eine Rahmenlinie gezeichnet. Klicken Sie erneut darauf, wird diese obere Rahmenlinie wieder entfernt; wurden zusätzlich mit Hilfe der restlichen Schaltflächen untere, mittlere, diagonale oder sonstige Rahmenlinien eingezeichnet, bleiben diese jedoch unverändert erhalten.

Statt auf diese Schaltflächen können Sie auch direkt im Vorschaufenster auf die gewünschten Positionen klicken, wodurch dort sofort die entsprechenden Rahmenlinien eingezeichnet bzw. – wenn Sie auf eine bereits vorhandene Linie klicken – wieder entfernt werden.



Erneut eine praktische Anwendung: Mit Rahmenlinien können Sie in der Mini-Buchführung die Überschriften und die Summen separat einrahmen, mit ein paar zusätzlichen vertikalen Rahmenlinien die Einnahmen von den Ausgaben trennen und einen Gesamtrahmen um die eigentliche Tabelle herum legen und zusätzlich noch die eigentliche Berechnung der Umsatzsteuervorauszahlung einrahmen, die sich darunter befindet.

Deaktivieren Sie dann noch das Kontrollkästchen *Gitternetzlinien* im Register *Ansicht* des Befehls EXTRAS|OPTIONEN..., um die Gitternetzlinien der Tabelle auszublenden, sieht das Ganze etwa so aus (Abbildung 6.35).

Abbildung 6.35:
Format.xls,
»Tabelle6«

	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12					Netto	MWSt			
13					- Steuerpflichtige Umsätze mit 16%	3327,16	532,34		
14					- Steuerpflichtige Umsätze mit 7%	1745,40	122,18		
15	Gesamt					5072,56	654,52		
16	Vorsteuerbeträge						462,95		
17	Umsatzsteuervorauszahlung						191,57		

Recht hübsch, denke ich – vor allem jedoch wesentlich übersichtlicher als zuvor!



Dieses Symbol der Formatleiste öffnet eine Liste verschiedener Rahmen-Symbole. Der selektierte Typ wird der aktiven Zelle bzw. dem selektierten Bereich zugewiesen.



Legt einen geschlossenen Rahmen um den selektierten Bereich herum



Erzeugt eine Rahmenlinie am linken Zellrand bzw. löscht eine bereits vorhandene linke Rahmenlinie



Erzeugt eine Rahmenlinie am rechten Zellrand bzw. löscht eine bereits vorhandene rechte Rahmenlinie



Erzeugt eine Rahmenlinie am oberen Zellrand bzw. löscht eine bereits vorhandene obere Rahmenlinie



Erzeugt eine Rahmenlinie am unteren Zellrand bzw. löscht eine bereits vorhandene untere Rahmenlinie



Erzeugt eine doppelte Rahmenlinie am unteren Zellrand bzw. löscht eine bereits vorhandene untere Rahmenlinie



Am Ende der Liste befindet sich die Schaltfläche Rahmenlinien zeichnen..., die die Rahmenlinie-Symboleleiste öffnet: Deren Stift-Symbol zeichnet beim Anklicken eines Zellrahmens eine Linie (Bereiche: Maustaste gedrückt halten und am entsprechenden Bereichsrand entlangfahren).

Im Listenfeld der Leiste können Sie die Dicke bzw. das Muster der Linie auswählen, mit dem *Farb*-Symbol die Linienfarbe bestimmen und mit dem Radiergummi eine Rahmenlinie durch Anklicken wieder entfernen (Maus-taste gedrückt halten und über Rahmenlinien fahren = radiert alle Linien wieder weg) bzw. bei gedrückt gehaltener Maustaste alle Rahmenlinien entfernen.

6.9 Muster und Blatthintergrund

Das Register *Muster* ist für das Muster zuständig, mit dem der Zellhintergrund ausgefüllt wird, und für die Farbe dieses Musters (Abbildung 6.36).

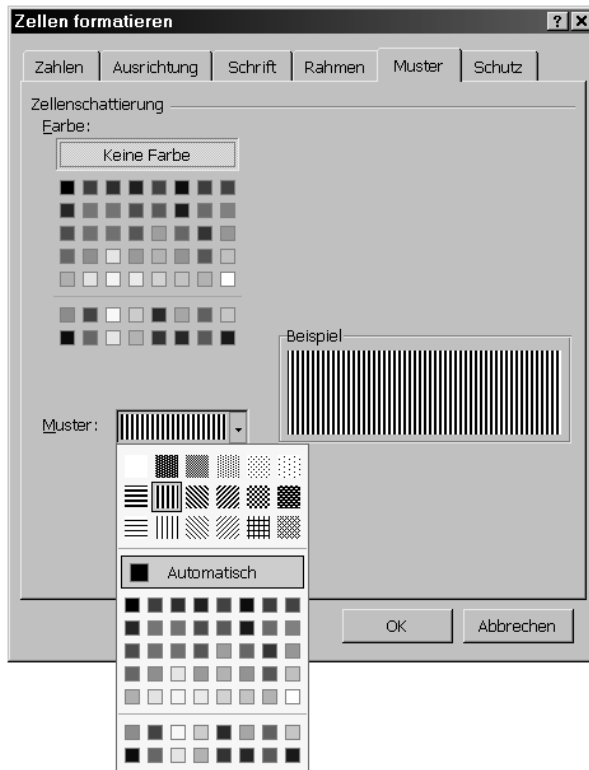


Abbildung 6.36:
Musterformatie-
rungen

Farbe bestimmt zunächst die Farbe des Zellhintergrunds. *Muster* öffnet eine ähnliche Liste und legt ein aus Punkten oder Linien bestehendes Muster über die Zelle.

Die Art dieses Musters bestimmen Sie per Anklicken eines der schwarzweißen Muster in der oberen Hälfte dieser Liste. Die untere Hälfte definiert die Farbe dieser Punkte oder Linien.

Einige Beispiele:

- ➔ Wählen Sie im Feld *Farbe* eine Farbe für den Zellhintergrund, wird die Zelle mit der betreffenden Farbe ausgefüllt.
- ➔ Wählen Sie *nur* im Feld *Muster* eines der schwarzweißen Muster, bleibt der normalerweise weiße Zellhintergrund erhalten und ein schwarzes Muster aus Punkten oder Linien wird über die Zelle gelegt.
- ➔ Wählen Sie *nur* im Feld *Muster* eine der Farben aus, passiert überhaupt nichts, da diese Selektion die Farbe der Punkte und Linien bestimmt, aus denen ein Muster besteht, und somit ohne Auswahl eines Musters keinerlei Wirkung besitzt.
- ➔ Wählen Sie eines der Muster *und zusätzlich* eine der darunter gezeigten Farben, bleibt der normalerweise weiße Zellhintergrund ebenfalls erhalten und ein Muster aus Punkten oder Linien in der von Ihnen gewählten Farbe wird über die Zelle gelegt.
- ➔ Wählen Sie eines der Muster, *zusätzlich* eine der darunter gezeigten Farben und *zusätzlich* eine der Farben unter *Farbe*, wird der Zellhintergrund in der gewählten Farbe eingefärbt und ein Muster aus Punkten oder Linien in der gewählten Farbe über die Zelle gelegt.



In mit Farbe gefüllten oder mit einem Muster überzogenen Bereichen werden übrigens keine Gitternetzlinien mehr dargestellt.

Zwei nützliche Symbole der Kategorie *Format* des *Anpassen*-Dialogfelds sehen Sie hier:



Färbt den Zellvordergrund dunkelgrau ein bzw. hebt eine dunkelgraue Einfärbung wieder auf



Färbt den Zellvordergrund hellgrau ein bzw. hebt eine hellgraue Einfärbung wieder auf

Blatthintergrund

Eine sehr komfortable Technik, den Hintergrund aller Zellen festzulegen, bietet der Befehl `FORMAT|BLATT|HINTERGRUND...`. Das Dialogfeld zum Öffnen von Dateien erscheint, wobei als Dateityp *Alle Grafiken* vorselektiert ist.

Sie können nun eine beliebige Grafikdatei aussuchen (beispielsweise eine der BMP-Dateien im Windows-Verzeichnis), die anschließend als Blatthintergrund verwendet wird (Abbildung 6.37).

Um den Ausgangszustand wiederherzustellen, wählen Sie den Befehl `FORMAT|BLATT|HINTERGRUND LÖSCHEN`.

	A	B	C	D	E	F	G
1		Einnahmen		Ausgaben			
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirt.
3		230,00	3859,50	2,00	19,80	284,00	18,00
4		1637,58		4,00	53,20	76,60	39,00
5				3,70		73,00	
6						68,90	
7							
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	58,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	8,00
10	Netto	1745,40	3327,16	9,70	62,93	433,10	50,00

Abbildung 6.37:
Blatthintergrund

6.10 Formatvorlagen

Sie können beliebige Formatdefinitionen zu einer Formatvorlage kombinieren, die Sie mit einem Mausklick jederzeit einer Zelle oder einem Bereich zuweisen können.

6.10.1 Definition und Zuweisung

Die Definition einer solchen Vorlage ist am einfachsten, wenn Sie zuvor einer Zelle die gewünschten Formate zuweisen und sie anschließend als Beispiel verwenden.

Nehmen wir an, Überschriften wollen Sie prinzipiell fett gedruckt darstellen und rechtsbündig ausrichten. Dann selektieren Sie eine entsprechend formatierte Zelle, in der folgenden Abbildung A2, und wählen FORMAT|FORMATVORLAGE... (Abbildung 6.38).

Das Listenfeld *Formatvorlagenname* enthält die Namen der vordefinierten Formatvorlagen. Vorselektiert ist immer *Standard*, Excels Standardvorlage.

Kümmern Sie sich bitte nicht darum, sondern tippen Sie entsprechend der Abbildung einen Namen wie »Überschriften« ein und aktivieren Sie OK. Damit haben Sie eine Formatvorlage namens »Überschriften« definiert, die alle Formatierungsmerkmale der zuvor selektierten Zelle enthält.

Sie können diese Vorlage jederzeit einer Zelle oder einem Bereich zuweisen, indem Sie diese(n) selektieren, wieder FORMAT|FORMATVORLAGE... wählen, im Listenfeld Ihre Formatvorlage »Überschriften« auswählen und OK aktivieren: Das Dialogfeld wird geschlossen und der Zelle bzw. dem Bereich werden alle Formateigenschaften zugewiesen, die diese Vorlage enthält; sie wird also nicht nur fett formatiert, sondern zusätzlich auch rechtsbündig ausgerichtet.

Abbildung 6.38:

Der Befehl
FORMAT|FORMAT-
VORLAGE...



Standard

Die einfachste Technik zur Zuweisung einer Formatvorlage: Sie selektieren den fraglichen Bereich und wählen danach die gewünschte Vorlage aus dem Listefeld des abgebildeten Symbols aus (Kategorie *Format* des *Anpassen*-Dialogfelds).

Aktivieren Sie im Befehl FORMAT|FORMATVORLAGE... statt OK die Schaltfläche *Hinzufügen*, wird die definierte Formatvorlage zwar der Liste hinzugefügt; im Gegensatz zu OK erfolgt jedoch keine Zuweisung dieser Vorlage an die aktive Zelle und das Dialogfeld wird auch nicht geschlossen.

Stattdessen können Sie darin anschließend einen weiteren Vorlagennamen eintippen und wie im folgenden Abschnitt erläutert die Eigenschaften dieses Formats definieren, seine Komponenten.

Integrierte Formate

Statt eigener können Sie auch eine der integrierten Vorlagen zuweisen. Am nützlichsten ist die Zuweisung der Formatvorlage »Standard«: Sie enthält Excels Standardformatierungen, die für alle Zellen gleichermaßen nach dem Anlegen einer neuen Tabelle gelten.

Die Zuweisung dieser Vorlage macht daher alle von diesem Standard abweichenden Formatierungen der aktiven Zelle bzw. des selektierten Bereichs wieder rückgängig.

6.10.2 Formatkomponenten

Die Kontrollkästchen unter *Formatvorlage enthält* bestimmen, welche Formatkomponenten die selektierte Vorlage umfasst.

Zunächst sind das alle Komponenten; der zu formatierenden Zelle wird das gleiche Zahlenformat, die gleiche Schriftart, dieselbe Ausrichtung, ein identischer Rahmen und der gleiche Zellschutz zugewiesen, den auch die Beispielzelle besitzt.

Diese Beeinflussung aller Formatierungskomponenten ist jedoch meist ein wenig zu weitgehend.

Betrachten Sie die vorhergehende Abbildung: Weisen wir C2 die Formatvorlage »Überschriften« zu, wird der darin enthaltene Text wie in A2 dargestellt, also fett und rechtsbündig ausgerichtet – und der zuvor vorhandene Rahmen entfernt, da dieser in der zugewiesenen Formatvorlage nicht enthalten war (Abbildung 6.39).

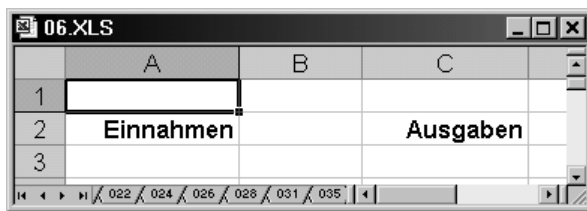


Abbildung 6.39:
Zu viele Formatkomponenten

Die Lösung: Bei der Definition einer neuen Formatvorlage lassen Sie unter *Formatvorlage enthält* nur jene Formatkomponenten aktiviert, die tatsächlich bei der Zuweisung der Vorlage beeinflusst werden sollen und deaktivieren alle anderen Komponenten, die beim Zuweisen der Vorlage unverändert bleiben sollen.

6.10.3 Ändern, Löschen und Zusammenführen

Angewandt auf das Beispiel, in dem mit der Vorlage nur für Fettdarstellung und rechtsbündige Ausrichtung gesorgt werden soll, dürften entsprechend nur die beiden Kontrollkästchen *Schriftart* und *Ausrichtung* aktiviert werden.

Also ändern Sie die bereits definierte Formatvorlage »Überschriften« entsprechend, indem Sie das Dialogfeld erneut öffnen, die Vorlage »Überschriften« selektieren, alle nicht in Frage kommenden Formatkomponenten deaktivieren und das Dialogfeld mit *OK* schließen (Abbildung 6.40).

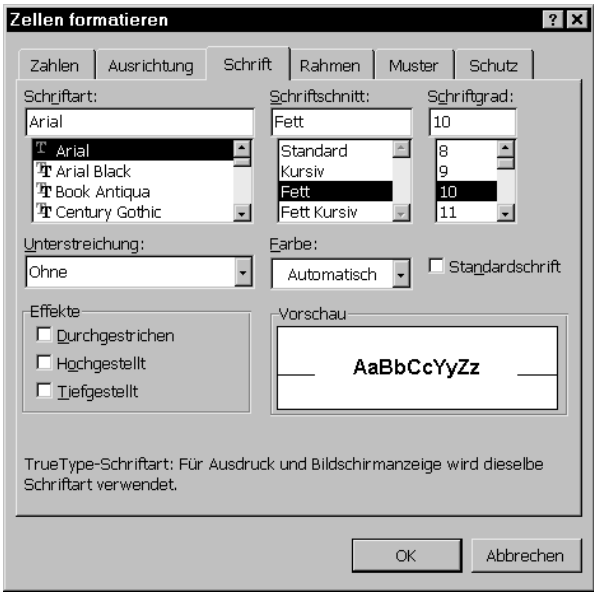
Falls Sie momentan keine geeignete Beispielzelle haben, aktivieren Sie die Schaltfläche *Ändern...*, die das bekannte Dialogfeld zum Formatieren von Zellen öffnet (Abbildung 6.41).

Mit den verschiedenen Registern können Sie die Eigenschaften der selektierten Vorlage unabhängig von der Formatierung der Beispielzelle festlegen.

Abbildung 6.40:
Formatvorlage
ändern



Abbildung 6.41:
Formatierungen
festlegen



Ändern Sie die Eigenschaften einer im Listenfeld des Formatvorlagen-Dia-
logfelds enthaltenen Vorlage und beenden Sie die Änderung mit OK, so
werden die geänderten Formateigenschaften sofort allen Zellen der Tabelle
zugewiesen, die mit der betreffenden Vorlage formatiert wurden.

Die Schaltfläche *Löschen* entfernt die momentan selektierte Formatvorlage
aus der Liste, wodurch allen Zellen, denen momentan diese Vorlage zuge-
wiesen ist, wieder die Vorlage »Standard« zugewiesen wird.

**Formate
zusammenführen**

Formatvorlagen sind immer mit der aktiven Arbeitsmappe verbunden. Defi-
nieren Sie eine Vorlage »Überschriften«, während gerade die Mappe A.XLS
aktiv ist, ist diese Vorlage zwar in allen Tabellen dieser Mappe verfügbar,
aber nicht in einer anderen Mappe als B.XLS.

Neu- oder umdefinierte Formatvorlagen sind somit nur für die Mappe gültig, die während der Definition aktiv war. Oft wäre es jedoch nützlich, Vorlagen von einer Mappe zu einer anderen übertragen zu können, um auch dort eine Formatvorlage wie »Überschriften« ohne erneute Definition anwenden zu können. Genau das ist die Aufgabe der Schaltfläche *Zusammenführen...* .

Wenn Sie eine neue Mappe anlegen, wählen Sie FORMAT|FORMATVORLAGE... und aktivieren diese Schaltfläche, erscheint eine Liste aller momentan im Speicher vorhandenen Arbeitsmappen außer der aktiven Mappe selbst (Abbildung 6.42).

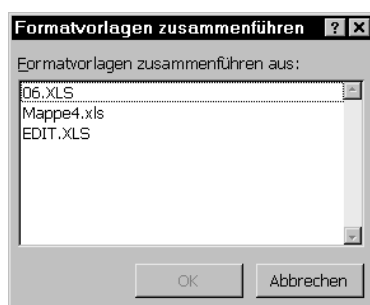


Abbildung 6.42:
Formatvorlagen
zusammenführen

Suchen Sie sich die Mappe aus, in der die Vorlage »Überschriften« definiert ist, und wählen Sie OK. Excel überträgt alle Formatvorlagen der ausgewählten Mappe in die momentan aktive Arbeitsmappe.

6.10.4 Formatierungs-Voreinstellungen ändern

Vielleicht wollen Sie Zahlen prinzipiell im Format »0,00« darstellen, finden wenig Gefallen an der Standardschrift oder sind ein Liebhaber farbiger Arbeitsblätter.

Dann ändern Sie einfach die Formatvorlage »Normal«, die Excel jeder Zelle einer neu angelegten Mappe zuweist.

Da sich jede Änderung einer Formatvorlage automatisch auf alle mit ihr formatierten Zellen auswirkt, wirkt sich die Änderung von »Normal« entsprechend auf die gesamte Mappe aus, solange Sie den darin enthaltenen Zellen noch keine anderen Vorlagen zugewiesen haben.

Auf eines sollten Sie allerdings achten: Wenn Sie in der Vorlage »Normal« ein bestimmtes Zahlenformat wie »0,00« wählen, müssen Sie Zellen, in denen Sie ein Datum oder eine Uhrzeit eintragen, anschließend doch per Hand formatieren. Statt dem Datum beziehungsweise der Uhrzeit wird sonst die zugehörige serielle Zahl im gewählten Format »0,00« dargestellt (siehe Kapitel 6.4.2, »Datums- und Zeitformate«).





Wollen Sie die neuen Standardeinstellungen von nun an für jede neue Mappe verwenden, so speichern Sie die Mappe im Dateiformat MUSTERVORLAGE unter dem Namen MAPPE.XLT im Verzeichnis XLSTART.

Von nun an wird Excel beim Anlegen einer neuen Arbeitsmappe diese Datei als Vorlage verwenden und die darin definierte Vorlage »Standard« als Standard-Formatvorlage.

6.11 AutoFormatieren

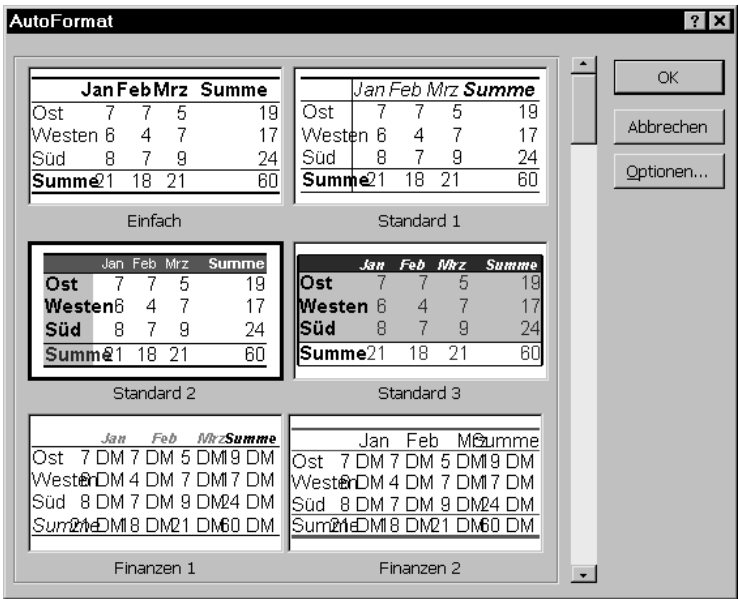
Die AutoFormatieren-Funktion ermöglicht es, einer Tabelle eines von vielen vorgegebenen Formatierungsschemata zuzuweisen.



AutoFormat analysiert den Bereich um die aktive Zelle herum und versucht herauszufinden, welche Bereiche Daten enthalten und welche zugehörige Spalten-/Zeilenüberschriften und Summen – und hebt letztere deutlich hervor.

Das klappt jedoch nur, wenn Ihre Tabellen entsprechend dem eingebauten Schema der AutoFormatierung aufgebaut sind. Excel zeigt Ihnen dieses Schema, wenn Sie FORMAT|AUTOFORMAT... wählen oder auf das zugehörige Symbol der Kategorie *Format* des *Anpassen*-Dialogfelds klicken (Abbildung 6.43).

Abbildung 6.43:
Erwartetes Tabe-
lenschema



Das Dialogfeld zeigt die verschiedenen Formatvarianten. Alle Beispiele gehen von einer Tabelle aus, in der

- ➔ sich die Daten in einem rechteckigen Bereich befinden,
- ➔ die Zeile darüber Spaltenüberschriften enthält,
- ➔ die Spalte links daneben Zeilenüberschriften enthält,
- ➔ die Zeile darunter Spaltensummen enthält.

Wird dieser Aufbau eingehalten, können Sie die entsprechende Tabelle problemlos mit der AutoFormatierung bearbeiten. Hält eine Tabelle dieses Schema ein, wird sie exakt wie dieses Beispiel formatiert.

Wird beispielsweise der Cursor auf Zelle B2 von »Tabelle5« der Datei FORMAT.XLS gesetzt und die AutoFormat-Funktion angewandt, selektiert diese daraufhin den Bereich B1:I6 und formatiert ihn nach Wahl von OK entsprechend dem gewählten Formatierungsschema, beispielsweise »Einfach« (Abbildung 6.44).

	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12					Netto	MWSt			
13		- Steuerpflichtige Umsätze mit 16%		####	532,34				
14		- Steuerpflichtige Umsätze mit 7%		####	122,18				
15	Gesamt			####	654,52				
16	Vorsteuerbeträge				462,95				
17	Umsatzsteuervorauszahlung				191,57				
18									

Abbildung 6.44:
Resultierende
Formatierung

Die Spaltenüberschriften werden korrekt erkannt und durch Rahmenlinien vom Zahlenbereich dazwischen abgehoben. Die Summenzeilen unterhalb dieses Bereichs wurden jedoch nicht erkannt und daher auch nicht formatiert.

Ursache dafür ist die leere Zeile 7 zwischen dem Zahlenbereich und den Summenformeln darunter. Aufgrund dieser Leerzeile erkennt AutoFormat nicht, dass es sich um einen zusammenhängenden Bereich handelt und die Bereichsselektion und Formatierung fällt nicht wie gewünscht aus.



Es ist jederzeit möglich, alle erfolgten Formatierungen gleichzeitig mit BEARBEITEN|RÜCKGÄNGIG:... aufzuheben und den Ausgangszustand wiederherzustellen. Voraussetzung ist allerdings, dass Sie diesen Befehl direkt nach der Formatierung geben.

Starten Sie danach einen weiteren Versuch: Selektieren Sie den Bereich A1:I10, wählen Sie erneut FORMAT|AUTOFORMAT... und aktivieren Sie OK, so wird der gesamte zuvor selektierte Bereich korrekt analysiert und formatiert (Abbildung 6.45).

Abbildung 6.45:
Formatierung des
selektierten
Bereichs

	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12					Netto	MWSt			
13					- Steuerpflichtige Umsätze mit 16%	#####	532,34		
14					- Steuerpflichtige Umsätze mit 7%	#####	122,18		
15	Gesamt				#####	654,52			
16	Vorsteuerbeträge					462,95			
17	Umsatzsteuervorauszahlung					191,57			



Allgemein: Ist vor der Wahl von FORMAT|AUTOFORMAT... kein Bereich selektiert, analysiert AutoFormat die Gegend um die aktive Zelle herum und selektiert und formatiert anschließend einen nach Meinung der Funktion geeigneten Bereich.

Wurde zuvor ein Bereich selektiert, formatiert AutoFormat stattdessen exakt diesen Bereich, was unter anderem immer dann notwendig ist, wenn sich vor einer Summenzeile eine oder mehrere Leerzeilen befinden.

Format-
komponenten
auswählen

Wie die Abbildung zeigt, nimmt AutoFormat übrigens keinerlei Rücksicht auf Zellen außerhalb des selektierten Bereichs: Die Spaltenbreiten werden an die im selektierten Bereich enthaltenen Werte mit der Funktion »Optimale Breite« angepasst, auch wenn sie – siehe Spalte E – danach zu gering für Werte außerhalb des Bereichs sind.

Um sich entsprechende Nacharbeiten zu ersparen, aktivieren Sie *Optionen...* und selektieren im nun erweiterten Dialogfeld, welche Formatierungskomponenten der Vorlage dem selektierten Bereich zugewiesen werden (Abbildung 6.46).

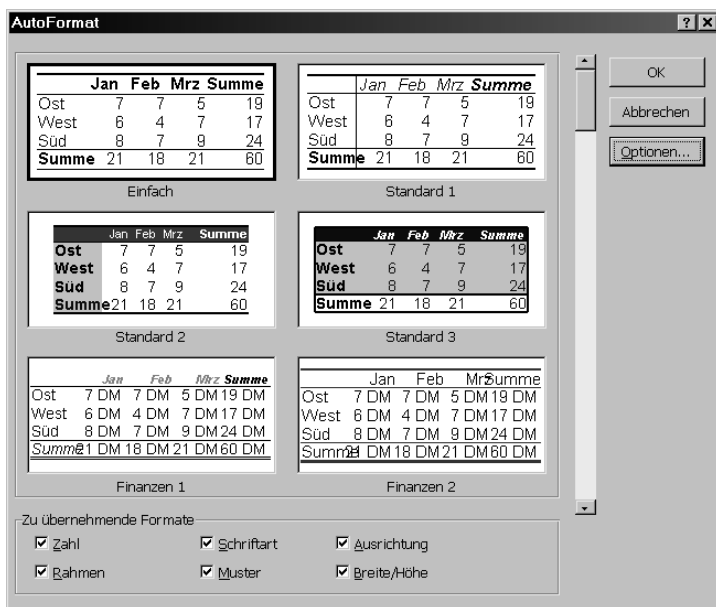


Abbildung 6.46:
Auswahl der
Formatierungs-
komponenten

Vorselektiert sind immer alle Komponenten. Haben Sie jedoch bereits mühsam eine Komponente wie die Spaltenbreiten per Hand optimiert, deaktivieren Sie einfach das entsprechende Kontrollkästchen *Breite/Höhe*, um diese Formatierungskomponente unverändert zu lassen.

AutoFormat ist in der Lage, oft auch Zwischenergebnisse korrekt zu erkennen. Dazu prüft Excel, welche Zellen im selektierten Bereich anstelle von Werten Formeln enthalten.

*Zwischen-
ergebnis-For-
matierung*

Beziehen sich die Bezüge einer solchen Formel ausschließlich auf Zellen, die sich in der gleichen Spalte *entweder* oberhalb *oder* (nicht beides gleichzeitig) unterhalb der Formelzelle befinden, nimmt AutoFormat an, dass die betreffende Zeile Zwischenergebnisse enthält, die zusätzlich zu den Endergebnissen am unteren Bereichsrand hervorgehoben werden sollen.

Selektieren Sie beispielsweise in der Demo-Tabelle irgendeine Zelle im Bereich E13:F17 und wählen Sie FORMAT|AUTOFORMAT..., so wird genau dieser Bereich, der die Endauswertung enthält, selektiert und formatiert (Abbildung 6.47).

AutoFormat erkennt die in Zeile 12 enthaltenen Spaltenüberschriften und fügt Trennstriche zwischen ihnen und den folgenden Zahlen ein.

Zeile 15 enthält die Formeln $=E13+E14$ und $=F13+F14$, die sich auf die Zeilen 13 und 14 beziehen. Daher wird Zeile 15 wiederum von diesen beiden Zeilen abgehoben und so weiter.

Abbildung 6.47:
Zwischen-
ergebnisse

06.XLS								
	A	B	C	D	E	F	G	H
1	Einnahmen			Ausgaben				
2	Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3	230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4	1637,58		4,00	53,20	76,50	396,00		
5			3,70		73,00			
6					68,90			
7								
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76
11								
12					Netto	MWSt		
13	- Steuerpflichtige Umsätze mit 16%				3327,16	532,34		
14	- Steuerpflichtige Umsätze mit 7%				1745,40	122,18		
15	Gesamt				5072,56	654,52		
16	Vorsteuerbeträge					462,95		
17	Umsatzsteuervorauszahlung					191,57		

6.12 Bedingte Formatierung

Der Befehl `FORMAT|BEDINGTE FORMATIERUNG...` ermöglicht es, in einem zuvor selektierten Bereich nur jene Zellen zu formatieren, auf die bestimmte Bedingungen zutreffen.

Das ist eine sehr elegante Technik, um in einem Bereich beispielsweise jene Zahlen hervorzuheben, die einen bestimmten Wert über- oder unterschreiten. Da die bedingte Formatierung dynamisch ist, wird die Formatierung dieser Zellen sofort wieder aufgehoben, wenn sich der Zellinhalt derart ändert, dass die Bedingung nicht mehr erfüllt ist, und sofort wiederhergestellt, wenn sie irgendwann später wieder erfüllt ist.

Zellwerte prüfen

Vor der Anwahl von `FORMAT|BEDINGTE FORMATIERUNG...` müssen Sie den Bereich selektieren, dem die bedingte Formatierung zugewiesen werden soll, in den folgenden Beispielen A1:C4 (Abbildung 6.48).

Im linken Listenfeld wählen Sie zunächst *Zellwert ist*. Ist rechts daneben *zwischen* selektiert und geben Sie in den beiden rechten Feldern entsprechend der Abbildung 100 bzw. 300 ein, werden alle Zellen formatiert, die Werte zwischen 100 und 300 enthalten.

Wie diese Formatierung erfolgt, legen Sie mit *Format...* fest: ein Dialogfeld mit den bekannten Registern *Schrift*, *Rahmen* und *Muster* erscheint, wobei die Formatmerkmale jedoch nur eingeschränkt anwählbar sind. Schließen Sie dieses und das vorhergehende Dialogfeld mit OK, so werden allen Zellen mit Werten zwischen 100 und 300 wie in der Abbildung die gewählten Formate zugewiesen.

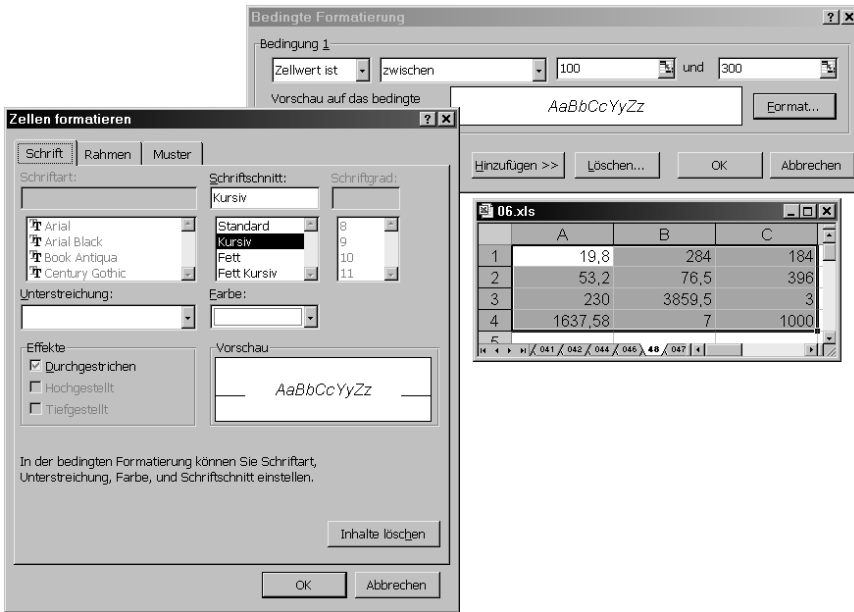


Abbildung 6.48:
Bedingte
Formatierung

Jedes Mal, wenn Sie auf *Hinzufügen >>* klicken, wird das Dialogfeld erweitert und Sie können eine weitere Bedingung eingeben, um beispielsweise auch allen Zellen im selektierten Bereich, die kleiner sind als 10, ein anderes Format zuzuweisen, das Sie wie zuvor mit einer eigenen *Format...*-Schaltfläche festlegen (Abbildung 6.49).

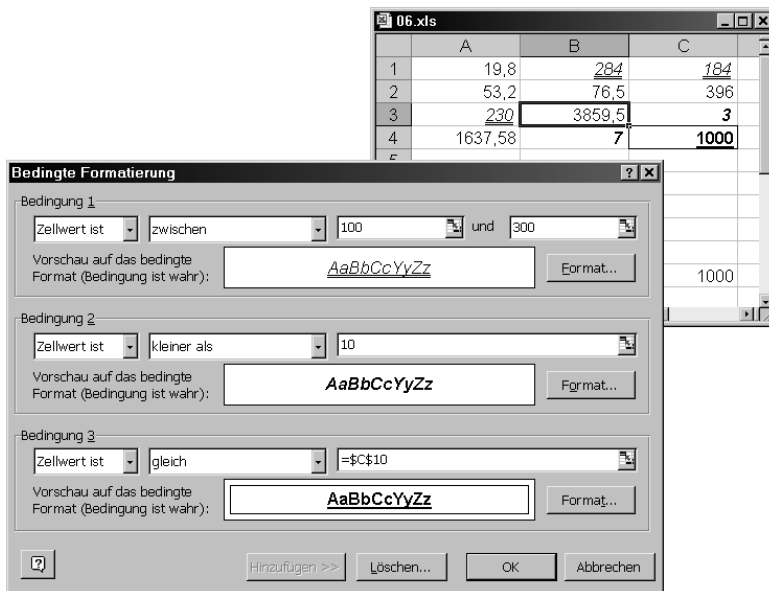


Abbildung 6.49:
Mehrere
Bedingungen

Die letzte Bedingung zeigt, dass Sie auch Formeln wie $=\$C\$10$ als Bedingungen verwenden können. Im Beispiel wird mit dieser Formel allen Zellen des Bereichs, die den gleichen Wert wie C10 enthalten, das zugehörige Format zugewiesen, im Beispiel also den Wert 1000.



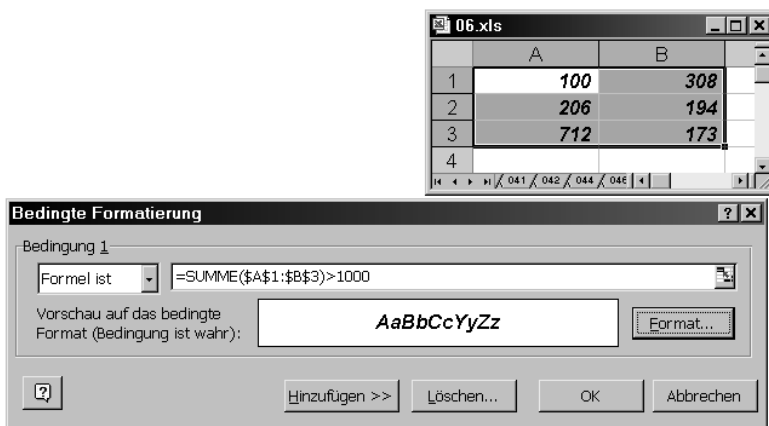
Relative Bezüge werden bei diesem Vergleich angepasst. In der Formel $=C10$ ist die Spalten- und die Zeilenangabe relativ. In diesem Fall wird die Zelle A1 zwar wie zuvor mit dem Inhalt von C10 verglichen und entsprechend formatiert, wenn sie den gleichen Wert enthält. A2 wird jedoch mit dem Inhalt von C11 verglichen, A3 mit dem Inhalt von C12 etc. Analog dazu wird B1 mit D10 verglichen, B2 mit D11 etc.

Entsprechend würde bei Verwendung eines Bezugs mit einer Mischung aus absoluten/relativen Koordinaten ($=C\$10$ bzw. $=\$C10$) nur die Spalten- bzw. nur die Zeilenangabe angepasst.

Formeln prüfen

Die Vorgabe *Zellwert ist* im linken Listenfeld prüft für jede einzelne Zelle des selektierten Bereichs, ob die Bedingung auf diese Zelle zutrifft und, falls ja, weist sie ihr die gewählte Formatierung zu. Verwenden Sie stattdessen das zweite Listenelement *Formel ist*, so wird die Bedingung genau einmal geprüft. Ist sie erfüllt, wird allen Zellen des Bereichs die Formatierung zugewiesen (Abbildung 6.50).

Abbildung 6.50:
Formel als
Bedingung



Die Bedingung muss aus einer Formel bestehen, die entweder den Wert **WAHR** oder **FALSCH** ergibt.

Im Beispiel ergibt die Formel $=SUMME(\$A\$1:\$B\$3)>1000$ den Wert **WAHR**, da die Summe der Werte im angegebenen Bereich größer als 1000 ist. Daraus resultiert, dass der komplette selektierte Bereich formatiert wird.

Ändern Sie die darin enthaltenen Werte derart, dass die Summe nicht mehr größer als 1000 ist, wird die Formatierung sofort aufgehoben.

Das Entfernen einer bedingten Formatierung setzt voraus, dass zuvor wieder der betreffende Bereich selektiert wird. Wissen Sie nicht mehr genau, wo dieser beginnt und endet, setzen Sie den Cursor irgendwo in diesen Bereich, wählen BEARBEITEN|GEHE ZU..., klicken auf Inhalte... und aktivieren Bedingte Formate, wodurch der betreffende Bereich selektiert wird.



Danach können Sie FORMAT|BEDINGTE FORMATIERUNG... wählen und mit Löschen... beliebige für diesen Bereich festgelegte bedingte Formatierungen wieder löschen.

7 Drucken

Nach dem Gestalten des Layouts wollen Sie Ihre Tabelle sicher irgendwann auch ausdrucken. Im folgenden Kapitel erläutere ich, wie Sie angeschlossene Drucker direkt aus Excel heraus installieren und einrichten, beispielsweise Papierformate auswählen oder die zu verwendende Auflösung bestimmen.

Bevor Sie Ihre Tabelle ausdrucken, können Sie sie in der Seitenansicht unter die Lupe nehmen und sich das voraussichtliche Druckergebnis zeigen lassen.

In der Seitenansicht können Sie sogar nachträglich das Layout der Tabelle verändern, beispielsweise durch Ziehen an den entsprechenden horizontalen bzw. vertikalen Linien die Seitenränder oder die Spaltenbreiten einstellen.

Vor allem lernen Sie im folgenden Kapitel, wie Sie alle denkbaren Aspekte des Seitenlayouts festlegen können. Zum Beispiel, wie Sie

- ➔ übergroße Tabellen durch Verkleinern an das zu verwendende Papierformat anpassen,
- ➔ Tabellen statt im üblichen Hoch- im Querformat ausdrucken,
- ➔ Seitenränder und den Inhalt von Kopf- und Fußzeilen definieren
- ➔ und auszudruckende Bereiche festlegen, wenn nicht wie üblich die gesamte Tabelle gedruckt werden soll.

Darüber hinaus zeige ich Ihnen, wie Sie die Positionen, an denen Seitenumbrüche erfolgen, individuell festlegen können, falls Ihre Tabelle zu lang oder breit ist, um komplett auf ein Blatt zu passen.

Und ich stelle Ihnen den Druck-Manager vor, der Ihre Druckaufträge im Hintergrund abwickelt, sodass Sie ungestört weiterarbeiten können, ohne das Ende des Ausdrucks abwarten zu müssen.

7.1 Druckerinstallation

Haben Sie mehrere Drucker, können Sie vor dem Ausdruck wählen, welchen davon Sie gerade benutzen wollen – vorausgesetzt, zuvor wurden alle Drucker korrekt installiert.

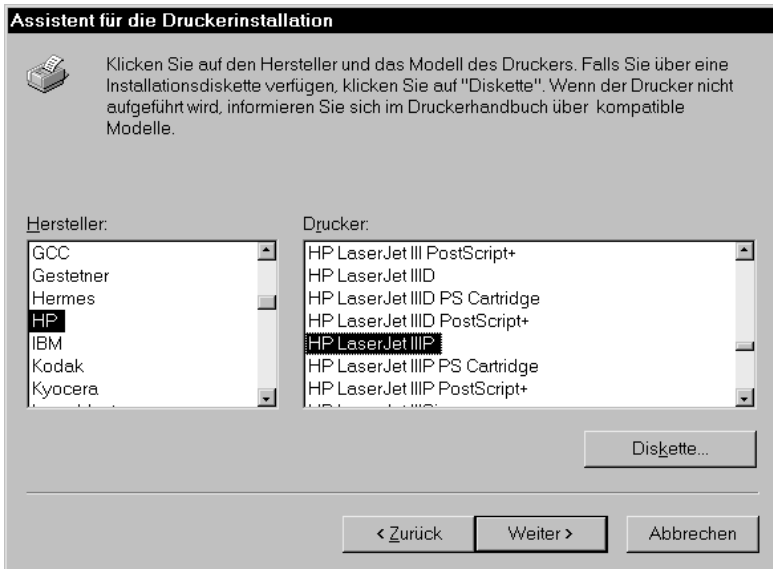
Neuer Drucker

Wahrscheinlich haben Sie Ihren Drucker bereits bei der Installation von Windows bekannt gegeben. Falls nicht: Sie können Drucker jederzeit per Doppelklick auf *Neuer Drucker* (zuvor START|EINSTELLUNGEN|DRUCKER wählen) auch nachträglich installieren.

Ein Assistent meldet sich, der Sie zunächst fragt, ob es sich um einen lokalen (an den aktuellen Rechner angeschlossenen) oder aber um einen Netzwerkdrucker handelt (einen Drucker, der an einen anderen Rechner angeschlossen ist, der mit dem aktuellen verbunden ist).

Danach können Sie den zu installierenden Drucker aus einer vorgegebenen Liste auswählen (Abbildung 7.1).

Abbildung 7.1:
Druckertyp
auswählen



Selektieren Sie links den Hersteller Ihres Druckers und danach rechts den Drucker selbst.

Ist Ihr Drucker nicht aufgeführt, besitzen Sie jedoch vom Hersteller eine Installationsdiskette mit den benötigten Windows-Druckertreibern, so klicken Sie stattdessen auf *Diskette...*

Leiten Sie anschließend den nächsten Installationsschritt ein, um diese Treiber zu installieren, entweder von Ihrer CD-ROM oder aber von den Windows-Installationsdisketten.

Danach geben Sie an, an welcher Schnittstelle der Drucker angeschlossen ist, meist am parallelen Anschluss *LPT1* (Abbildung 7.2).

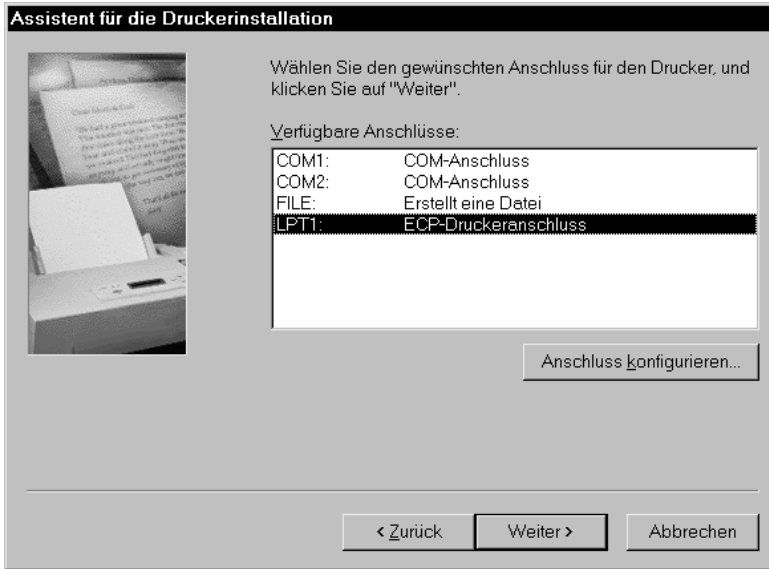


Abbildung 7.2:
Schnittstelle
bekanntgeben

Kennen Sie die genaue Schnittstellenbezeichnung nicht, gehen Sie so vor: Selektieren Sie LPT1 oder COM1, je nachdem, ob der Drucker an eine serielle (COM) oder eine parallele (LPT) Schnittstelle angeschlossen ist, und versuchen Sie, die vom Assistenten einige Schritte später vorgeschlagene Testseite auszudrucken.

**:-)
TIPP**

Misslingt das, blättern Sie mit < Zurück bis zur Schnittstellen-Konfiguration zurück, selektieren eine andere Schnittstelle und versuchen es erneut solange, bis der Ausdruck der Testseite klappt.

Auch wenn mehrere Drucker installiert sind, kann immer nur einer aktiv sein. Daher müssen Sie im folgenden Schritt angeben, ob der betreffende Drucker als Standarddrucker eingesetzt wird (Abbildung 7.3).

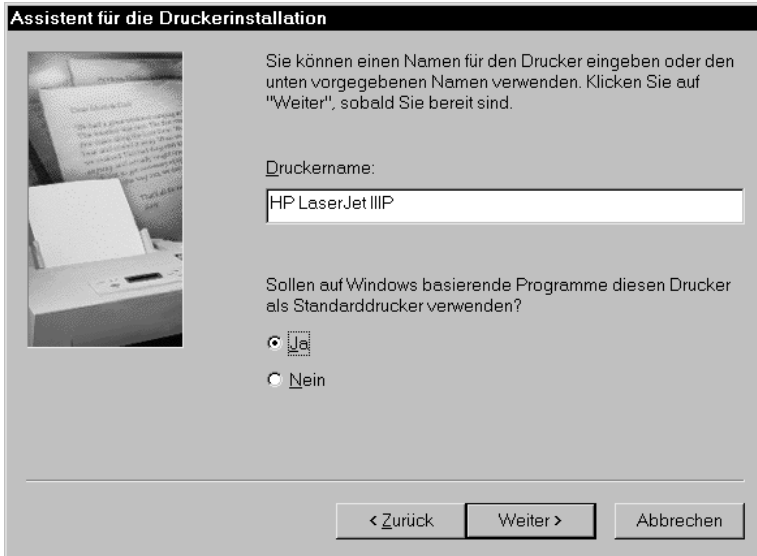
Ja führt dazu, dass alle Windows-Programme automatisch diesen Drucker benutzen, solange Sie nicht angeben, dass ein anderer Drucker benutzt werden soll.

Danach kann die erwähnte Testseite gedruckt werden und die Installation ist damit beendet.

Sie können jederzeit einen anderen der im Ordner »Drucker« angezeigten Drucker als Standarddrucker angeben. Dazu öffnen Sie den Ordner, klicken den betreffenden Drucker mit der rechten Maustaste an und wählen im zugehörigen Kontextmenü ALS STANDARD DEFINIEREN.

**f
INFO**

Abbildung 7.3:
Standarddrucker



7.1.1 Drucker einrichten

Nach der Installation eines Druckers kann näher bestimmt werden, wie er sich beim Ausdruck verhalten soll. Dazu öffnen Sie den Ordner »Drucker« (Befehl **START|EINSTELLUNGEN|DRUCKER**), klicken den einzurichtenden Drucker mit der rechten Maustaste an und wählen im zugehörigen Kontextmenü **EIGENSCHAFTEN** (Abbildung 7.4).

Der Inhalt dieses Dialogfelds hängt vom betreffenden Drucker ab. Die am häufigsten vorkommenden und wichtigsten Optionen sind die folgenden:

Register *Grafik*

Im abgebildeten Register *Grafik* legen Sie fest, wie Grafiken ausgedruckt werden.

Auflösung definiert, wie fein beziehungsweise grob der Ausdruck wird. Windows stellt alle Auflösungen zur Verfügung, die Ihr Drucker beherrscht. Je höher die Punktzahl, desto kleiner ist jeder einzelne Punkt und umso besser ist somit die Auflösung.

Deutlich niedriger ist allerdings auch die Druckgeschwindigkeit, sodass es vor allem beim Entwurf eines Arbeitsblatts, bei dem viele Probeausdrucke stattfinden, sinnvoll sein kann, dabei noch nicht mit der höchsten Auflösung zu arbeiten.

Register *Papier*

Die Option *Einzug* des Registers *Papier* kennen Sie sicher von anderen Programmen her. Die übliche Wahl ist die zwischen Einzelblatt (Manuelle Zufuhr) und Endlospapier (Traktor).

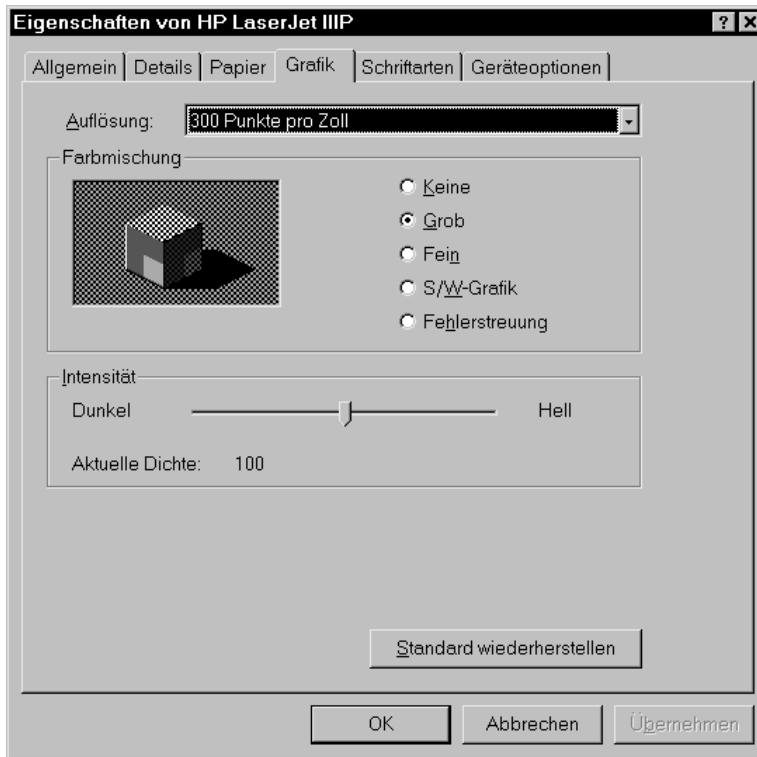


Abbildung 7.4:
Druckereigen-
schaften

Im ersten Fall fordert Excel Sie nach dem Bedrucken eines Blatts auf, ein neues Blatt Papier einzulegen, im zweiten wird fortlaufend weitergedruckt, bis die gesamte Tabelle ausgegeben ist.

Unter *Papierzufuhr* geben Sie speziell bei Laserdruckern die Papierquelle an. Mein eigener uralter HP IIIIP besitzt einen oberen und einen unteren Papier-schacht. Ich kann zum Beispiel in den unteren Schacht rein weißes Papier und in den oberen vorgedruckte Briefbögen einlegen und mit dieser Einstellung angeben, welcher der beiden Schächte beim Drucken benutzt werden soll.

Die Einstellung der Option *Papiergröße* ist zweifellos nicht immer gleich und wird ab und zu von Ihnen verändert, weil Sie zum Beispiel abwechselnd Briefe oder Formulare drucken.

Papiergröße

Allerdings wird diese Option bereits ab der Version 3.0 von Excel nicht mehr benötigt, da Sie das Papierformat nun zusätzlich mit dem Befehl `DATEI|SEITE EINRICHTEN...` angeben können. Das ist sehr vorteilhaft, da alle Einstellungen, die Sie mit diesem Befehl vornehmen, zusammen mit der aktuellen Tabelle gespeichert werden, Sie also für jede Tabelle eigene Einstellungen definieren können.

Format Das gleiche gilt für die Option *Ausrichtung*. Im Hochformat druckt Excel so viele Spalten wie möglich nebeneinander und die restlichen Spalten auf einem der folgenden Blätter.

Querformat dreht den Ausdruck um 90 Grad, sodass die auf einem Blatt druckbare Spaltenanzahl von der Papierlänge statt von der Breite abhängt. Entsprechend können oftmals auch recht breite Blätter noch vollständig auf einem Blatt ausgegeben werden.

Dafür ist der Ausdruck im Querformat speziell bei Nadeldruckern langsamer und oft extrem unsauber!

**Register
Geräteoptionen**

Druckerspeicher wird vor allem dann wichtig, wenn Sie Ihren Drucker nachträglich erweitern. Zum Beispiel von einem Mbyte auf zwei, drei oder vier Mbyte. Dann sollten Sie nicht vergessen, Windows die neue Speichergröße bekannt zugeben.



Mit 512 Kbyte oder ähnlich mickrigen Werten erhalten Sie beim Ausdruck komplexer Excel-Tabellen eventuell Probleme, wenn Sie in der höchsten Auflösung drucken.

Bei Laserdruckern wird nämlich meist die gesamte Seite im Druckerspeicher aufbereitet und 512 Kbyte reichen dazu bei maximaler Auflösung oft nicht aus. Dann sind Sie gezwungen, im Register *Grafik* eine niedrigere Auflösung zu wählen, nach dem Motto: »besser ein minderwertiger als gar kein Ausdruck«.

**Register
Schriftarten**

Excel und viele andere Windows-Programme können außer den serienmäßig eingebauten Schriften auch nachträglich im Drucker installierte Schriftkassetten verwenden, aber nur, wenn Sie die Erweiterung unter *Kassetten* bekanntgeben.

Register Details

Das Register *Details* enthält alle Informationen über die Druckerinstallation. Sie können darin beispielsweise nachträglich die Schnittstellenangabe verändern, neue Druckertreiber einbinden, oder ein Zeitlimit wählen: Damit teilen Sie Windows mit, wie lange es warten soll, bis es mangels Rückmeldung vom Drucker zum Schluss kommt, dass dieser nicht bereit ist, zum Beispiel auf *OFF Line* steht oder ausgeschaltet ist.

Diese Option ist sehr nützlich, falls Ihr Drucker zum Aufbereiten komplexer Grafiken möglicherweise länger braucht als von Windows erwartet und Windows fälschlicherweise zum Schluss kommt, mit dem Drucker würde etwas nicht stimmen, obwohl dieser nur ein wenig länger für die Aufbereitung der komplexen Grafik braucht.

Mit der Schaltfläche *Spool-Einstellungen...* öffnen Sie ein weiteres Dialogfeld, in dem Sie festlegen, ob der Druck im Hintergrund erfolgen soll (in einer Warteschlange; bei schnellen Rechnern empfehlenswert, da sofort mit Excel weitergearbeitet werden kann), in welchem Datenformat gedruckt wird (EMF ist am schnellsten) und ob – falls vorhanden – der Ausdruck durch Verwendung bidirektionaler Parallelschnittstellen an Rechner und Drucker beschleunigt werden soll.

7.2 Ausdrucken

Das Drucker-Symbol der Standardleiste druckt die aktive Tabelle entsprechend der Standardeinstellungen von Excel aus.



DATEI|DRUCKEN... öffnet zuvor das folgende Dialogfeld (Kurtaste **Strg** + **P** oder **Strg** + **⇧** + **F12** oder **Alt** + **Strg** + **⇧** + **F2**) (Abbildung 7.5).

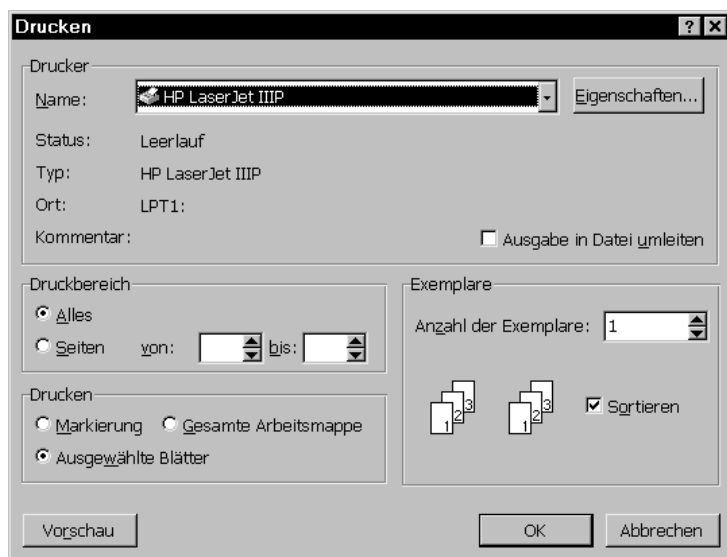


Abbildung 7.5:
Dialogfeld zum
Ausdruck einer
Tabelle

Unter *Drucker* ist der Windows-Standarddrucker vorgegeben. Stattdessen können Sie jedoch einen beliebigen anderen installierten Drucker wählen.

Drucker

Die Schaltfläche *Eigenschaften...* ermöglicht Ihnen, die aktuellen Drucker-einstellungen (Papierformat, aktuelle Auflösung etc.) vor dem Ausdruck zu verändern.

Markierung druckt nur den bzw. die (Mehrfachselektion) selektierten Bereiche.

Drucken

Ausgewählte Blätter druckt die aktive bzw. die selektierten Tabellen einer Arbeitsmappe und *Gesamte Arbeitsmappe* druckt alle nicht leeren Tabellen dieser Mappe.

Druckbereich Beim Druck längerer Bereiche werden diese über mehrere Seiten verteilt. *Alles* druckt in diesem Fall alle Seiten einer Tabelle, *Seiten* dagegen nur die Seiten *von* bis *bis*, zum Beispiel nur Seite 3 bis Seite 5.

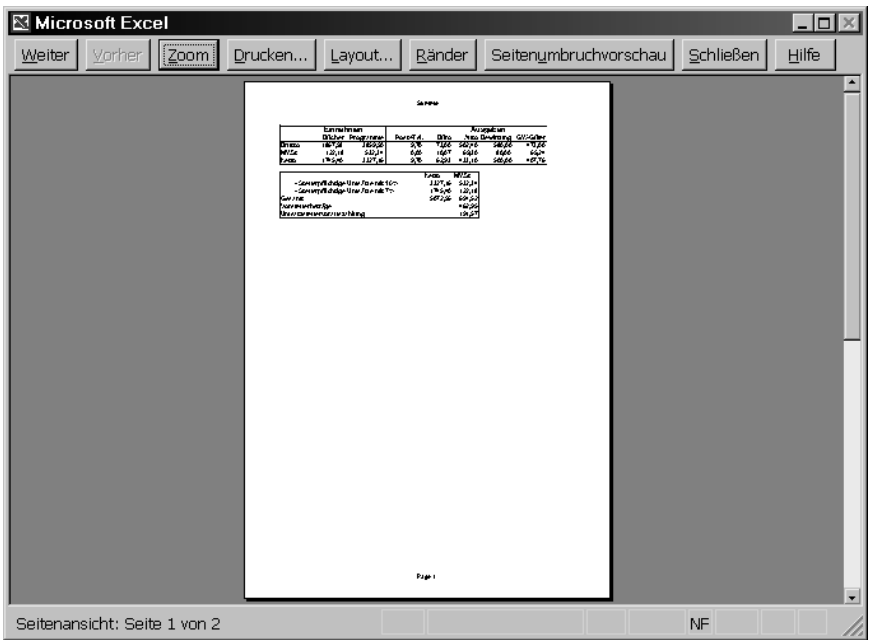
Exemplare *Exemplare* legt fest, wie oft die Tabelle ausgedruckt wird. Das Kontrollkästchen *Sortieren* bestimmt, ob identische Seiten direkt aufeinander folgend ausgedruckt werden, Excel bei fünf Exemplaren also zunächst fünfmal die Seite 1 druckt, danach fünfmal die Seite 2 und so weiter.

Aktivieren Sie *Ausgabe in Datei umleiten*, erscheint nach OK ein Dialogfeld, in dem Sie der Druckdatei einen Namen geben. Später können Sie diese Druckdatei auf das gewünschte Windows-Drucker-Symbol ziehen, um sie tatsächlich zu Papier zu bringen.

OK startet den Ausdruck mit den aktuellen Einstellungen.

7.2.1 Seitenansicht

Abbildung 7.6:
Prüfen des
Seitenlayouts



So sieht das voraussichtliche Druckergebnis aus, wenn momentan die Tabelle 6 von FORMAT.XLS aktiv ist.

Diese Seitenansicht können Sie mit dem zugehörigen Symbol der Standardleiste ebenso wie mit dem Befehl DATEI|SEITENANSICHT aktivieren (und aus dem Dialogfeld des in Kürze erläuterten Befehls DATEI|SEITE EINRICHTEN... heraus).



Ist Ihre Tabelle sehr lang oder breit, reicht ein Blatt wahrscheinlich nicht und sie wird wie diese über mehrere Blätter verteilt ausgedruckt. Dann zeigt die Schaltfläche *Weiter* die zweite Seite und danach eine eventuelle dritte Seite, und *Vorher* blättert genau umgekehrt wieder seitenweise zurück.

Sind Sie nach dieser Prüfung mit der Seitengestaltung einverstanden, klicken Sie auf *Drucken...*, um das *Drucken*-Dialogfeld zu öffnen, ohne zuvor DATEI|DRUCKEN... wählen zu müssen.

Da Sie in der Seitenansicht sehen, welche Tabellenteile sich auf welchen Seiten befinden, können Sie nun auch die Option *Seiten* des *Drucken*-Dialogfelds sinnvoll einsetzen, um nur ganz bestimmte Seiten auszudrucken.

Klicken Sie mit dem inzwischen als Lupen-Symbol dargestellten Mauscursor auf irgendeine Stelle einer Seite, die Sie näher interessiert, so vergrößert Excel den betreffenden Bereich (Abbildung 7.7).

Lupenfunktion

Einnahmen		Ausgaben			
Bücher	Programme	Post+Tel.	Büro	Auto	Bewir
230,00	3859,50	2,00	19,80	284,00	18
1637,58		4,00	53,20	76,50	38
		3,70		73,00	
				68,90	
Brutto	1867,58 3859,50	9,70	73,00	502,40	58
MWSt	122,18 532,34	0,00	10,07	69,30	8
Netto	1745,40 3327,16	9,70	62,93	433,10	58

Abbildung 7.7:
Die Lupenfunktion

Den in diesem Vergrößerungsmodus gezeigten Ausschnitt können Sie mit den Scrollbalken oder den Cursortasten in alle Richtungen verschieben. Klicken Sie erneut irgendeine Stelle an oder auf *Zoom*, so wird der Vergrößerungsmodus wieder deaktiviert.

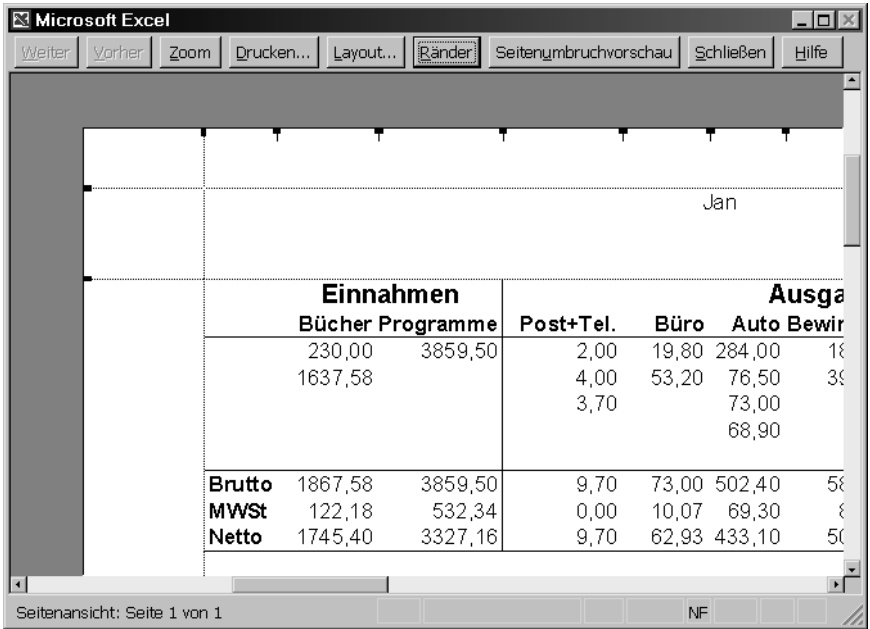


Die Schaltfläche *Seitenumbruch-Vorschau* aktiviert bzw. deaktiviert die in Kapitel 7.3.5, »Manueller Seitenumbruch«, erläuterte gleichnamige Tabellenansicht, in der Sie die Seitenumbrüche auf einfache Art und Weise verändern können.

7.2.2 Interaktives Layouten

In der Seitenansicht können Sie die Seitenränder, die Spaltenbreiten und die Kopf- und Fußzeilen manipulieren, wenn Sie auf die Schaltfläche *Ränder* klicken (Abbildung 7.8).

Abbildung 7.8:
Randeinstellungen
mit der Maus



Die gepunkteten Linien kennzeichnen die Blattränder. Durch Ziehen an diesen Linien können Sie die Tabelle in beliebige Richtungen verschieben und den Abstand zur Kopf- beziehungsweise Fußzeile beeinflussen.

Sie können sogar die Spaltenbreiten verändern, indem Sie die am oberen Blattrand angedeuteten Spaltentrennlinien nach rechts oder links ziehen.

Alle Änderungen der Randeinstellungen werden automatisch in das Dialogfeld des Befehls *DATEI|SEITE EINRICHTEN...* übernommen.

7.3 Seitenlayout

DATEI|SEITE EINRICHTEN... öffnet ebenso wie die *Layout...*-Schaltfläche der Seitenansicht das nachfolgend abgebildete Dialogfeld (allerdings sind darin nur bei Wahl von DATEI|SEITE EINRICHTEN... die beiden zusätzlichen Schaltflächen *Drucken...* und *Seitenansicht* enthalten, mit denen direkt aus dem Dialogfeld heraus der Druck eingeleitet bzw. die Seitenansicht aktiviert werden kann).

Darin vorgenommene Einstellungen beziehen sich immer auf das aktive Arbeitsblatt und werden zusammen mit ihm gespeichert und wieder geladen.

7.3.1 Papierformat

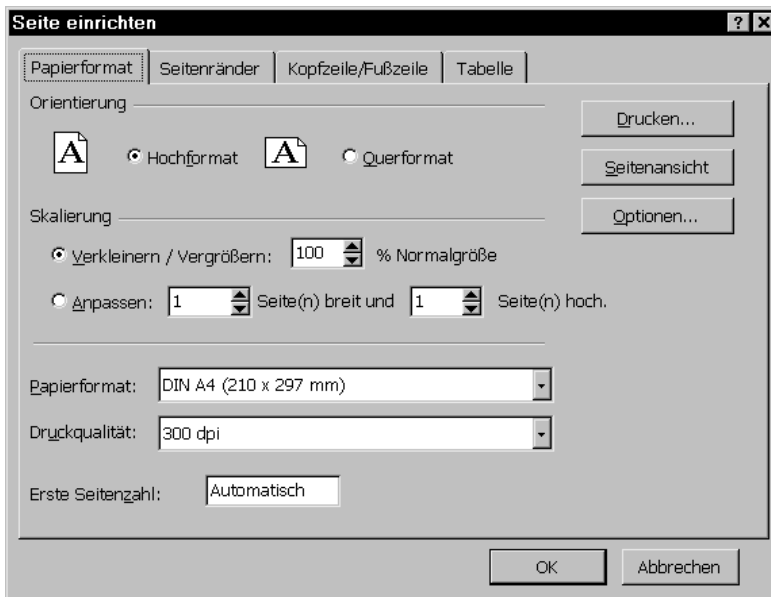


Abbildung 7.9:
Register »Papier-
format«

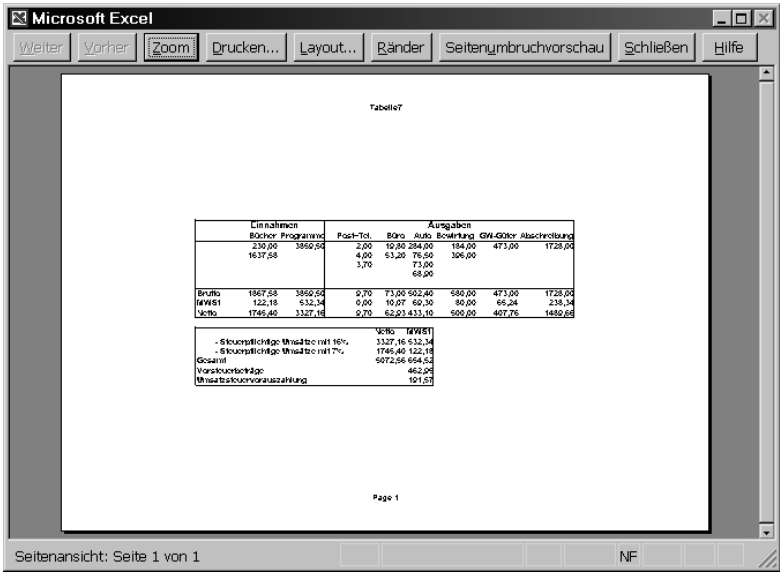
Unter *Orientierung* ist immer *Hochformat* voreingestellt. Sie sollten diese Einstellung nur dann in *Querformat* ändern, wenn Sie eine sehr breite Tabelle drucken wollen und nicht alle Spalten auf das Blatt passen. Aktivieren Sie das Querformat, dann passt die Tabelle oft noch problemlos auf ein einziges Blatt (Abbildung 7.10).

Orientierung

Diese Option hilft jedoch nur, wenn wie in diesem Fall eine Tabelle nur zu breit und nicht zusätzlich auch zu lang ist. Dann sollten Sie lieber die Rubrik *Skalierung* benutzen.

Skalierung

Abbildung 7.10:
Format.xls,
»Tabelle7«



Verkleinern/Vergrößern verkleinert bzw. vergrößert den Ausdruck durch entsprechende Anpassung der Schriftgrade und der Spaltenbreiten auf den angegebenen Prozentsatz der Originalgröße.

Aktivieren Sie *Anpassen*, berechnet Excel den Verkleinerungsfaktor selbst, der benötigt wird, um den Ausdruck sowohl in der Breite als auch in der Höhe auf der angegebenen Seitenzahl unterzubringen.



Geben Sie jeweils eine 1 ein, so wird eine Tabelle so weit verkleinert, dass sie gerade noch auf ein einziges Blatt passt. Ein Beispiel dafür finden Sie in Kapitel 7.3.5, »Manueller Seitenumbruch«.

Papierformat *Papierformat* muss an das Format des von Ihnen verwendeten Papiers angepasst werden, beispielsweise an das DIN-A4-Format.

Erste Seitenzahl *Erste Seitenzahl* legt fest, welche Nummer die erste ausgedruckte Seite erhält, die anschließend in der Fußzeile erscheint.

Das ist normalerweise eine 1, Sie können der ersten Seite jedoch auch die Nummer 2, 3 etc. geben.

An dieser Startnummer orientieren sich übrigens die in Kürze erläuterten Formatcodes für die Seitenzahl.

Druckqualität *Druckqualität* bestimmt die Qualität und damit auch die Geschwindigkeit des Ausdrucks. Die verfügbaren Einstellungen hängen vom installierten Drucker ab.

Wichtig bei Laserdruckern: Erhalten Sie beim Ausdruck eine Fehlermeldung und teilen Ihnen entweder diese Fehlermeldung selbst oder über das Druckerdisplay mit, dass der Drucker nicht genug Speicher besitzt, um ein Dokument in der gewünschten Auflösung zu drucken, müssen Sie eine niedrigere Qualitätsstufe und damit eine geringere Auflösung wählen.

Die Schaltfläche *Optionen...* öffnet das Dialogfeld zur Anpassung der aktuellen Druckereinstellungen (Grafikauflösung etc.).

*Drucker-
einstellungen*

7.3.2 Seitenränder

Das Register *Seitenränder* ist selbsterklärend (Abbildung 7.11).

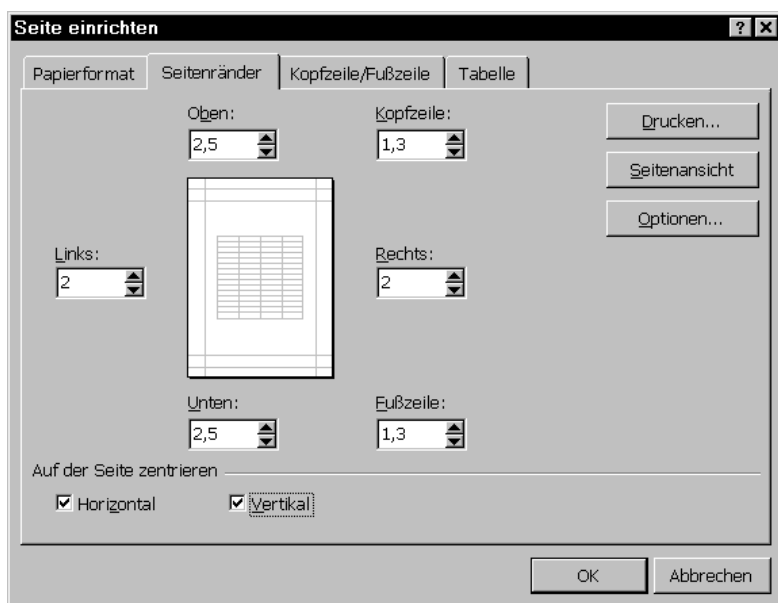


Abbildung 7.11:
Register
»Seitenränder«

Hinzu kommt, dass nach dem Anklicken eines der Randfelder die zugehörige Randlinie im Demonstrationfeld hervorgehoben wird, sodass Sie sehen, welche Randeinstellung Sie gerade verändern.

Während die vier Felder *Oben*, *Unten*, *Rechts* und *Links* den Abstand der Tabelle zum Blattrand bestimmen, legen *Kopfzeile* und *Fußzeile* den Abstand der Kopf- bzw. Fußzeile zum oberen/unteren Blattrand fest.

Normalerweise müssen Sie die Standardeinstellungen nicht ändern. Außer Sie haben besonders breite oder lange Arbeitsblätter und wollen, dass diese auf möglichst wenigen Blättern ausgedruckt werden. Dann setzen Sie alle vier Randeinstellungen auf Null.

Zentrieren

Sehr nützlich sind die Zentrierungs-Kontrollkästchen *Horizontal* und *Vertikal*.

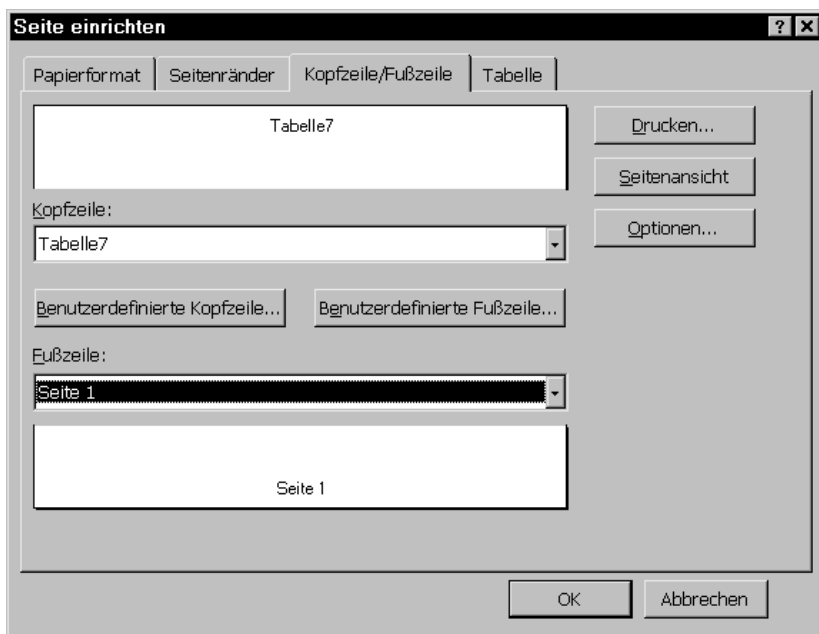
Vielleicht wollen Sie Tabellen oft in der Blattmitte ausdrucken, mit identischem Abstand vom linken und rechten beziehungsweise oberen und unteren Rand. Wenn ja, aktivieren Sie einfach diese beiden Kontrollkästchen, statt langwierig mit den vier Randeinstellungen herumzuexperimentieren.

7.3.3 Kopfzeile/Fußzeile



Statt zunächst DATEI|SEITE EINRICHTEN... zu wählen und im Dialogfeld das Register Kopfzeile/Fußzeile zu aktivieren, können Sie dieses Register mit dem Befehl ANSICHT|KOPF- UND FUSSZEILE... auch direkt aufschlagen (Abbildung 7.12).

Abbildung 7.12:
Register »Kopfzeile/
Fußzeile«

**Standard**

In den zu *Kopfzeile* und *Fußzeile* gehörenden geschlossenen Listefeldern können Sie zwischen mehreren Elementen für die Kopf- und Fußzeile ein passendes auswählen. Jedes dieser Elemente steht stellvertretend für einen anderen Kopf- bzw. Fußzeilentext.

Vorgegeben sind unter anderem die Elemente »(keine)« (leere Kopf- bzw. Fußzeile), die Anzeige des Dateinamens oder des Namens der aktuellen Tabelle und ein Element, das sowohl den Tabellennamen als auch gleichzeitig die Seitennummer anzeigt.

Wenn Sie ein Element auswählen, wird das zugehörige Vorschauenfenster sofort entsprechend aktualisiert. Zusätzlich können Sie mit *Seitenansicht* die Seitenansicht oder mit *Drucken...* das entsprechende Dialogfeld aktivieren.

Reichen Ihnen diese Elemente nicht aus, definieren Sie mit *Benutzerdefinierte Kopfzeile...* bzw. *Benutzerdefinierte Fußzeile...* individuelle Kopf-/Fußzeilen (Abbildung 7.13).

*Individuelle
Definition*

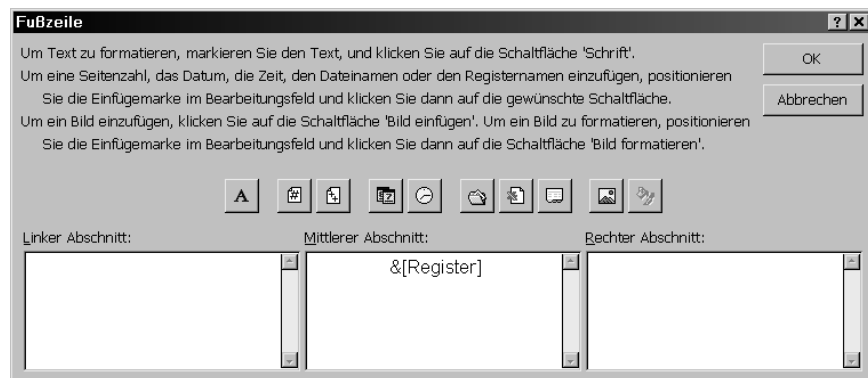


Abbildung 7.13:
Individuelle Kopf-/
Fußzeilen

Text, den Sie in der linken Box eingeben, wird linksbündig am oberen Blatt-
rand ausgerichtet.

Text, den Sie in der Mitte eingeben, wird zentriert, was wohl die am häu-
figsten verwendete Ausrichtung ist.

Text, den Sie in der rechten Box eingeben, wird rechtsbündig ausgerichtet.

 +  bewirkt bei der Texteingabe einen Zeilenumbruch und ermöglicht
Ihnen die Eingabe mehrzeiliger Texte.

Aktivieren Sie beispielsweise »Tabelle7« der Datei FORMAT.XLS und geben
Sie als Kopfzeile in der mittleren Box den Text »Dies ist«  +  »ein
Test« ein, ist das Resultat der entsprechende zweizeilige Kopfzeilentext
(Abbildung 7.14).

Mithilfe der verschiedenen Symbole oberhalb der drei Textfelder können
spezielle Effekte erreicht werden.

Mit dem linken Symbol, dem A, können Sie die Schrift definieren, die Excel
für die Kopfzeile verwendet. Zuvor müssen Sie den zu formatierenden Kopf-
zeilentext selektieren, danach klicken Sie auf das Symbol und erhalten als
Resultat das bekannte *Schriftart*-Dialogfeld zur Auswahl von Schriftart/-stil/
-größe und besonderer Schriftattribute.

Abbildung 7.14:
Zentrierter
mehrzeiliger Text

Microsoft Excel

Start Vorherige Zoom Drucken... Layout... Bänder Seitenrhythmus-Vorschau Schließen Hilfe

Dies ist
ein Text

Einnahmen			Ausgaben					
Bucher	Programme	Post+Tel.	Buro	Auto	Bewirtung	GW.Güter	Abschreibung	
230,00	3659,50	2,00	19,80	264,00	184,00	473,00	1728,00	
1637,58		4,00	53,20	76,50	396,00			
		3,70		73,00				
				68,90				
Brutto	1967,58	3659,50	9,70	73,00	502,40	500,00	473,00	1728,00
MWSt:	122,18	532,34	0,00	10,07	88,30	80,00	65,24	238,34
Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66

	Netto	MWSt
- Steuerpflichtige Umsätze mit 16%	3327,16	532,34
- Steuerpflichtige Umsätze mit 7%	1745,40	122,18
Gesamt	5072,56	654,52
Vorsteuerbeträge		462,95
Umsatzsteuerabzug		191,57

Page 1

Seitenansicht: Seite 1 von 1

Formatcodes

Die restlichen Symbole fügen nach dem Anklicken an der aktuellen Cursorposition einen Formatcode in den Text ein, der eine Stellvertreterfunktion besitzt und im Ausdruck durch folgenden Text ersetzt wird:



Seitennummer fügt an der aktuellen Cursorposition den Code »&[Seite]« ein, der bewirkt, dass an dieser Stelle die Seitennummer gedruckt wird.



Seitenanzahl fügt den Code »&[Seiten]« ein, der für die Gesamtseitenanzahl steht.



Datum fügt den Code »&[Datum]« ein, das aktuelle Datum.



Uhrzeit fügt den Code »&[Zeit]« ein, die aktuelle Uhrzeit.



Pfad und Dateiname fügt den Code »&[Pfad]&[Datei]« ein, den Pfad und den Dateinamen, also C:\EIGENE DATEIEN\TEST.XLS oder ähnlich.



Dateiname fügt den Code »&[Datei]« ein, den Dateinamen, also TAB1.XLS, TEST.XLS oder ähnlich.



Tabellenname fügt den Code »&[Register]« ein, den Arbeitsblattnamen, also »Summe« oder ähnlich.

Bild einfügen öffnet ein *Dateiauswahl*-Dialogfeld, in dem Sie eine einzufügende Grafikdatei auswählen können und fügt anschließend den Code »&[Grafik]« ein, der das betreffende Bild repräsentiert.



Bild formatieren öffnet ein Dialogfeld, das es Ihnen ermöglicht, ein zuvor eingefügtes Bild zu formatieren, beispielsweise seine Größe und Helligkeit zu definieren.



Ebenso wie Text können Sie jeden dieser Codes in eine beliebige Box einfügen. Und soll das einen Formatcode einleitende Zeichen »&« selbst ausgedruckt werden, geben Sie einfach »&&« ein.

In der folgenden Abbildung wurde die linke (linksbündige Ausrichtung) Box angeklickt und danach das Datums-Symbol angeklickt, das den Code »&[Datum]« einfügt (Abbildung 7.15).

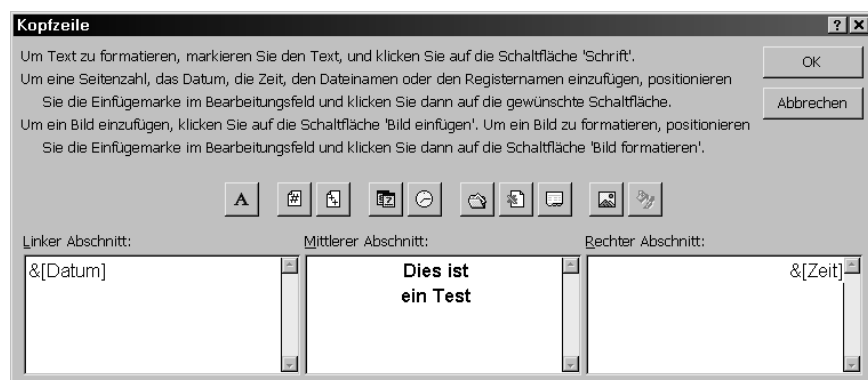


Abbildung 7.15:
Kopf- und Fußzeilen-
lencodes

Anschließend gab ich in der mittleren Box den Text »Dies ist ein Test« ein (mit Zeilenumbruch und folgender Schriftformatierung) und fügte in der rechten Box mit dem Uhren-Symbol den Uhrzeitcode »&[Zeit]« ein.

Das Resultat sehen Sie nicht nur in der Seitenansicht, sondern bereits direkt nach der Definition der Kopf-/Fußzeile im übergeordneten Dialogfeld in einer Art Mini-Seitenansicht (Abbildung 7.16).

7.3.4 Tabelle

Die im Register Tabelle vorhandenen Optionen sind nur dann vollständig verfügbar, wenn Sie den Befehl DATEI|SEITE EINRICHTEN... wählen! Öffnen Sie dieses Dialogfeld dagegen aus der Seitenansicht heraus mit der Schaltfläche Layout..., so sind die Optionen Druckbereich, Drucktitel und Kommentare im Gegensatz zur Abbildung nicht verfügbar und mehrere Schaltflächen fehlen (Abbildung 7.17).



Abbildung 7.16:
Ausdruck mit Kopf-
zeilencodes

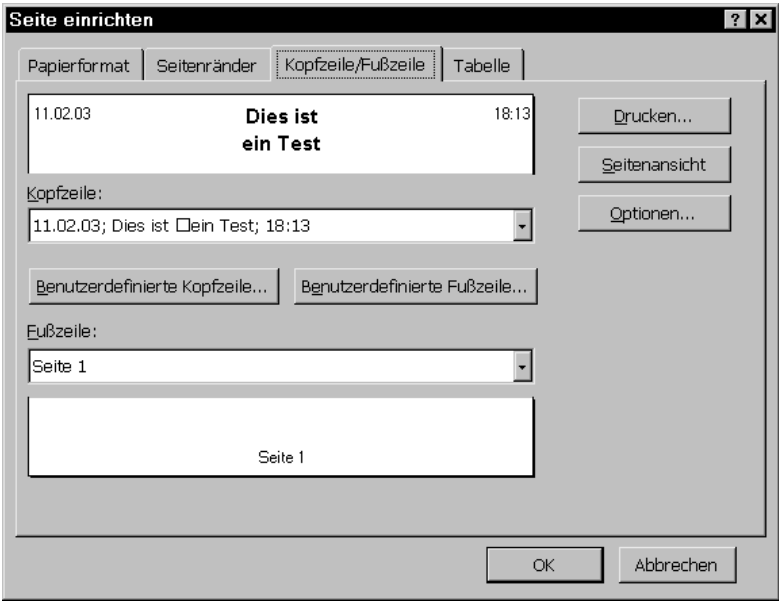
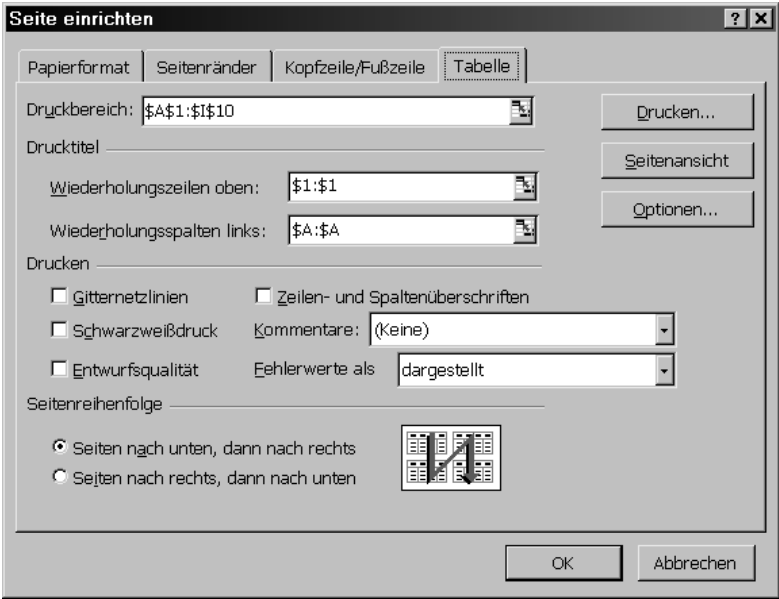


Abbildung 7.17:
Register »Tabelle«



Druckbereich

Wollen Sie nicht die gesamte Tabelle drucken, dann können Sie unter *Druckbereich* einen oder mehrere auszudruckende Bereiche in der Form »Bereich1;Bereich2;...;BereichN« angeben, beispielsweise »A1:I10;E4:F7«, um nur die beiden Bereiche A1:I10 und E4:F7 auszudrucken.

Einfacher: Sie aktivieren dieses Eingabefeld, selektieren mit der Maus den ersten Bereich und bei gedrückter **Strg**-Taste den zweiten Bereich etc. Bei jeder weiteren Bereichsselektion fügt Excel das Semikolon und die zugehörige Bereichsreferenz automatisch in das Eingabefeld ein.

Ist das Dialogfeld beim Zeigen auf die Bereiche im Weg, klicken Sie auf die bunte Schaltfläche am rechten Rand von *Druckbereich*, um es zu verkleinern, und nach dem Zeigen auf den Bereich erneut darauf, um die alte Größe wiederherzustellen.

Mit Hilfe dieses Symbols, das dem Befehl DATEI|DRUCKBEREICH|DRUCKBEREICH FESTLEGEN entspricht, können Sie Druckbereiche festlegen, ohne zuvor das Dialogfeld zu öffnen: Selektieren Sie in der Tabelle die auszuführenden Bereiche und klicken Sie auf dieses Symbol (Kategorie *Datei* des *Anpassen*-Dialogfelds), fügt Excel die betreffenden Bereichsbezüge in das Feld *Druckbereich* des (geschlossenen) Dialogfelds ein.



Soll irgendwann später wieder die gesamte Tabelle gedruckt werden, löschen Sie den Inhalt dieses Felds entweder manuell oder wählen einfach den Befehl DATEI|DRUCKBEREICH|DRUCKBEREICH AUFHEBEN.

Drucktitel definiert, welche Zeilen beziehungsweise Spalten auf jedem einzelnen Blatt erscheinen sollen.

Drucktitel

Ein Beispiel: Sie wollen nur den Bereich A1:I10 von »Tabelle6« der Datei FORMAT.XLS drucken. Also aktivieren Sie *Druckbereich* und selektieren diesen Bereich.

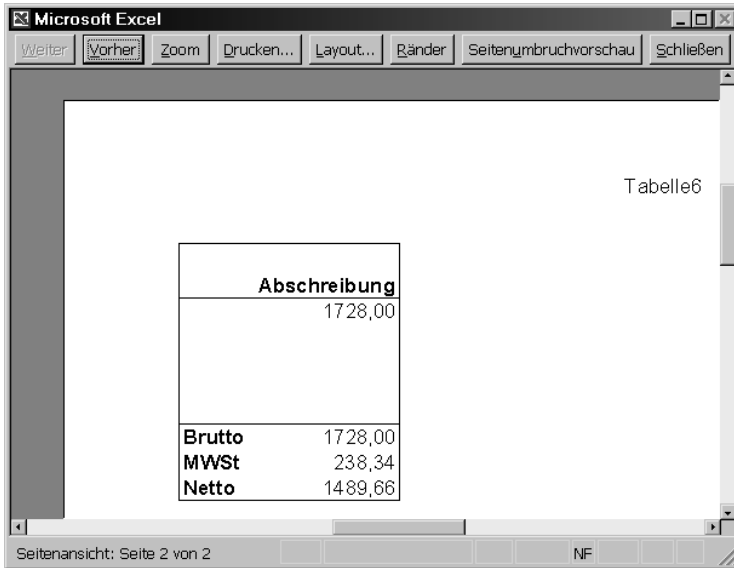
Ist er zu breit, um auf einem Blatt ausgedruckt zu werden, würde beispielsweise A1:H10 auf dem ersten und I1:I10 (die Spalte »Abschreibung«) auf dem zweiten Blatt ausgedruckt werden.

Leider sind auf diesem zweiten Blatt die Zeilenüberschriften »Brutto«, »MWSt.« und »Netto« nicht vorhanden, was den Überblick erschwert.

Kein Problem: Sie aktivieren *Wiederholungsspalten links* und klicken irgendeine Zelle von Spalte 1 an. Excel fügt in das Eingabefeld die Spaltenreferenz »\$A:\$A« ein, mit folgenden Auswirkungen beim Ausdruck der zweiten Seite (Abbildung 7.18).

Analog dazu können Sie mit *Wiederholungszeilen oben* eine oder mehrere Zeilen mit Spaltenüberschriften wiederholen, wenn eine Tabelle zwar nicht zu breit, dafür jedoch zu lang ist, um auf einem Blatt ausgedruckt zu werden.

Abbildung 7.18:
Spalte mit Zeilen-
überschriften
wiederholen



Bei einem Fall wie in Kapitel 7.3.5, »Manueller Seitenumbruch«, bei dem eine Tabelle zu lang und zu breit ist, verwenden Sie beide Optionen, um die Spalten- und die Zeilenüberschriften auf den nachfolgenden Blättern zu wiederholen.

Zeilen- und Spalten- überschriften

Die Optionen *Zeilen- und Spaltenüberschriften* und *Gitternetzlinien* sind selbsterklärend. Nur wenn beide aktiviert sind, werden die Zeilennummern bzw. die Spaltenbezeichnungen und die feinen Trennlinien zwischen Spalten und Zeilen gedruckt.

Kommentare

Unter *Kommentare* können Sie auswählen, ob Zellkommentare überhaupt nicht ausgedruckt werden sollen (»Keine«), »Am Ende des Blattes«, oder dort, wo sie auch in der Tabelle zu sehen sind (»Wie auf dem Blatt angezeigt«).

Fehlerwerte als

Unter *Fehlerwerte als* legen Sie fest, was gedruckt wird, wenn Zellen Fehlerwerte enthalten; ob der Fehlerwert originalgetreu gedruckt werden soll oder ob stattdessen eine leere Zelle gedruckt werden soll (also nichts) etc.

Entwurfsqualität

Entwurfsqualität beschleunigt den Ausdruck durch Verzicht auf grafische Elemente. In der letzten Abbildung würden beispielsweise weder Zellrahmen noch Gitternetzlinien gedruckt.

Schwarzweiß- druck

Zellfarben werden auf Druckern, die nicht farbfähig sind, in Muster umgesetzt. Mit dem Resultat, dass ein auf dem Bildschirm wunderschön gelb oder blau formatierter Zellvordergrund als Muster ausgedruckt wird, das die Lesbarkeit des Zellinhalts meist extrem verschlechtert.

Aktivieren Sie *Schwarzweißdruck*, verzichtet Excel gänzlich darauf, Zellfarben beim Ausdruck in Muster umzusetzen und druckt ausschließlich Schwarz auf Weiß. Sie können Ihre Tabellen nun farbig gestalten, ohne im Ausdruck eine schlechtere Lesbarkeit in Kauf nehmen zu müssen.

Seitenreihenfolge legt fest, in welcher Reihenfolge Excel Tabellen ausdruckt, die sich über mehrere Blätter erstrecken (siehe Kapitel 7.3.5, »Manueller Seitenumbruch«).



Diese Option besitzt keinen Einfluss auf das Druckergebnis, sondern nur auf die Reihenfolge, in der eine Tabelle gedruckt wird, die sich über mehrere Blätter erstreckt.

Beide Optionen drucken zunächst die linke obere Ecke der Tabelle. Mit *Seiten nach unten, dann nach rechts* wird die Tabelle anschließend von oben nach unten abgearbeitet: Die zweite Seite enthält den Tabellenabschnitt darunter, die dritte den wiederum darunter liegenden Abschnitt und so weiter, bis die Tabelle in der gesamten Länge ausgedruckt ist, aber noch nicht in der vollen Breite, was nun auf den folgenden Blättern geschieht.

Seiten nach rechts, dann nach unten arbeitet die Tabelle genau umgekehrt von links nach rechts ab und druckt erst beim Erreichen des rechten Tabellenrands den darunter liegenden Teil ebenfalls von links nach rechts auf die einzelnen Blätter verteilt.

Die Schaltfläche *Optionen...* öffnet übrigens das *Druckereinrichtung*-Dialogfeld, und *Drucken...* das *Drucken*-Dialogfeld.

7.3.5 Manueller Seitenumbruch

Zum Seitenlayout gehört auch das manuelle Einfügen von Seitenumbrüchen.

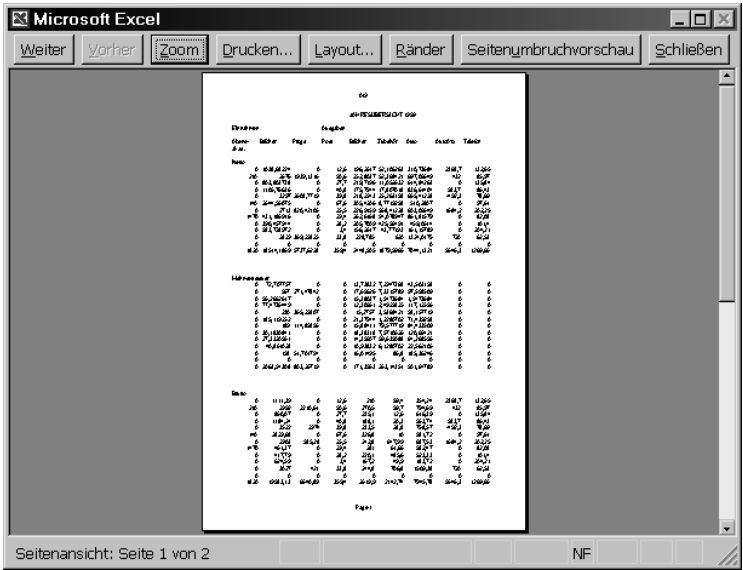
Automatischer Umbruch

Excel legt die Seitenunterteilung in Abhängigkeit von der eingestellten Papierlänge und -breite und den eingestellten Seitenrändern selbstständig fest: Bei sehr langen Tabellen bedruckt Excel das erste Blatt vom eingestellten oberen bis zum unteren Rand, setzt den Ausdruck auf einem zweiten Blatt fort und so weiter.

Nehmen wir jedoch an, Ihre Tabelle ist in bestimmte zusammengehörige Blöcke unterteilt, die nicht zerrissen werden sollen, wie in der folgenden einfachen Buchführung.

Sie enthält einen Block mit den Nettoeinnahmen/-ausgaben aller Monate eines Jahres. Durch ein paar Leerzeilen getrennt folgt ein weiterer Block mit den zugehörigen Umsatzsteuereinnahmen/-ausgaben, dem wiederum ein dritter Block mit den Bruttoeinnahmen/-ausgaben für jeden Monat folgt (Abbildung 7.19).

Abbildung 7.19:
Demotabelle mit
»Anpassen«
aktiviert



In dieser Abbildung passt die umfangreiche Tabelle, die von A1 bis M55 reicht, auf ein einziges Blatt; aber nur, weil ich die bereits erläuterte Option *Anpassen* aktivierte, bei der Excel die Tabelle so weit verkleinert, bis sie auf die von mir angegebene Blattanzahl 1 passt.

Dabei werden jedoch die mühsam formatierten Schriftgrade zunichte gemacht und die Schrift wird verkleinert!

Soll die Demo-Tabelle jedoch ohne *Anpassen* in den Original-Schriftgraden ausgedruckt werden, benötigt Excel dazu in diesem Fall vier Blätter statt eines einzigen und die Tabelle enthält mehrere Seitenumbrüche.

Wo ein solcher Seitenumbruch stattfindet, ermittelt Excel, wenn diese Berechnung tatsächlich notwendig wird, also erst beim Ausdruck der Tabelle oder bei der Vorschau darauf in der Seitenansicht.

Schauen Sie sich die Tabelle anschließend wieder an, werden die ermittelten Seitenumbrüche durch gestrichelte Umbruchlinien angezeigt (Abbildung 7.20).

Zwischen Zeile 47 und Zeile 48 findet ein horizontaler und zwischen Spalte I und Spalte J ein vertikaler Umbruch statt.

Die Tabelle wird also in vier Blöcke unterteilt. Der linke obere Block wird auf dem ersten Blatt ausgedruckt, der linke untere Block auf dem zweiten Blatt, der rechte obere Block auf dem dritten und der rechte untere Block auf dem vierten Blatt – wobei die Ausgabereihenfolge wie erläutert von der Option *Seitenreihenfolge* abhängt.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	Abend- skod.	Bücher	Progs	Post	Bücher	Zubehör	Auto	Auto0%	Telefon	Bewir- tung	GW- Güter	Lohn	Abschrei- bung	
9	Netto													
10	0	1039	0	12,6	196,262	52,1053	310,737	2368,7	132,65	328,947	0	0	0	
11	210	2676	1939,1	50,6	252,804	52,3684	697,096	432	85,97	534,396	0	0	0	
12	0	803,8	0	27,7	219,72	11,0526	614,843	0	135,84	596,316	0	0	51049,1	
13	0	1107	0	40,8	175,794	17,807	836,614	933,7	86,43	321,316	0	0	0	
14	0	3297	2608,8	39,8	218,224	25,2632	665,412	-492,3	78,69	839,43	0	0	0	
15	140	2645	0	67,6	305,421	8,77193	510,281	0	97,61	685,781	0	0	0	
16	0	2713	820,42	25,5	226,916	568,412	603,096	1684,2	202,25	423,509	0	0	0	
17	1470	431,2	0	29,4	262,617	54,0789	861,816	0	82,08	508,333	0	0	0	
18	0	390,5	0	28,2	205,701	425,965	459,061	0	101,4	404,825	0	0	1014,04	
19	0	583,7	0	3,4	156,262	43,7719	161,158	0	204,21	43,5088	0	0	1666,67	
20	0	2829	369,3	33,8	228,785	620	1324,02	720	62,53	142,962	0	0	0	
21	0	0	0	0	0	0	0	0	0	0	0	0	0	
22	1820	18514	5737,6	359,4	2448,5	1879,6	7044,13	5646,3	1269,66	4829,33	0	0	53729,8	
23														
24														
25	Mehrwertsteuer													
26	0	72,71	0	0	13,7383	7,29474	43,5032	0	0	46,0526	0	0	0	
27	0	957	271,48	0	17,9963	7,33158	97,5935	0	0	74,814	0	0	0	
28	0	56,27	0	0	15,3804	1,54737	1,54737	0	0	83,4842	0	0	7146,88	
29	0	77,47	0	0	12,3056	2,49298	117,126	0	0	44,9642	0	0	0	
30	0	230	365,23	0	15,2757	3,53684	93,1577	0	0	117,52	0	0	0	
31	0	185,1	0	0	21,3794	1,22807	71,4393	0	0	96,0093	0	0	0	
32	0	189	114,86	0	15,8841	79,5777	84,4335	0	0	59,232	0	0	0	
33	0	30,18	0	0	18,3832	7,57105	120,654	0	0	71,1667	0	0	0	
34	0	27,33	0	0	14,3991	59,6351	64,2686	0	0	56,6754	0	0	141,965	
35	0	40,86	0	0	10,9383	6,12807	22,5621	0	0	6,09123	0	0	233,333	
36	0	198	51,702	0	16,015	86,8	185,362	0	0	20,0175	0	0	0	
37	0	0	0	0	0	0	0	0	0	0	0	0	0	
38	0	2064	803,27	0	171,395	263,144	901,648	0	0	676,107	0	0	7522,18	
39														
40														
41	Brutto													
42	0	1111	0	12,6	210	59,4	354,24	2368,7	132,65	375	0	0	0	
43	210	2992	2210,6	50,6	270,5	59,7	794,69	432	85,97	609,2	0	0	0	
44	0	860,1	0	27,7	235,1	12,6	616,39	0	135,84	679,8	0	0	58196	
45	0	1184	0	40,8	188,1	20,3	953,74	933,7	86,43	366,3	0	0	0	
46	0	3522	2974	39,8	233,5	28,8	758,57	-492,3	78,69	956,95	0	0	0	
47	140	2830	0	67,6	326,8	10	581,72	0	97,61	781,79	0	0	0	
48	0	2903	935,28	25,5	242,8	647,99	687,53	1684,2	202,25	482,8	0	0	0	
49	1470	461,4	0	29,4	281	61,65	982,47	0	82,08	579,5	0	0	0	
50	0	417,8	0	28,2	220,1	485,6	523,33	0	101,4	461,5	0	0	1156	
51	0	624,6	0	3,4	167,2	49,9	183,72	0	204,21	49,6	0	0	1900	
52	0	3027	421	33,8	244,8	708,8	1509,38	720	62,53	163	0	0	0	
53	0	0	0	0	0	0	0	0	0	0	0	0	0	
54	0	0	0	0	0	0	0	0	0	0	0	0	0	
55	1820	19933	6540,9	359,4	2619,9	2142,74	7945,78	5646,3	1269,66	5505,44	0	0	61252	

Abbildung 7.20:
Seitenumbruchs-
linie

Wenn die Originalschriftgrade erhalten bleiben sollen, sind vier Blätter zwar unumgänglich – aber nicht zwangsläufig mit dieser katastrophalen Seitenaufteilung. Dass der vertikale Seitenumbruch beispielsweise mitten im unteren der drei Zahlenblöcke stattfindet, können Sie ändern.

*Manueller
Umbruch*

Im Beispiel wäre es sinnvoll, den horizontalen Umbruch zwischen den zweiten und den dritten Zahlenblock zu verlegen, beispielsweise zwischen Zeile 41 und Zeile 42. Dadurch würde vermieden, dass der dritte Block über zwei Blätter zerrissen ausgedruckt wird.

Zusätzlich könnten wir den vertikalen Umbruch so legen, dass die Tabelle von der Breite her ungefähr hälftig aufgeteilt wird, zum Beispiel zwischen die Spalten F und G.

Um Seitenwechsel manuell festzulegen, wählen Sie ANSICHT|SEITENUMBRUCH-VORSCHAU oder klicken in der Seitenansicht auf die gleichnamige Schaltfläche (Abbildung 7.21).

*Die Seitenum-
bruchs-Vorschau*

Abbildung 7.21:
Die Seitenumbruch-
Vorschau

JAHRESÜBERSICHT 2009												
Einnahmen												
Ausgaben												
Abend-skad.	Bücher	Progs	Post	Bücher	Zubehör	Auto	Auto0%	Telefon	Bewirtung	Gw/- Güter	Lohn	Abschreibung
Netto	0	1038,682	0	12,6	196,262	52,1053	310,737	2368,7	132,65	328,947	0	0
210	2676	1939,13	50,6	252,804	52,3684	697,096	432	85,97	534,386	0	0	0
0	803,8037	0	27,7	219,72	11,0526	614,843	0	135,84	596,316	0	0	51049,12
0	1106,766	0	40,8	175,794	17,807	836,614	933,7	86,43	321,316	0	0	0
0	3297	2608,77	39,8	218,224	25,2632	665,412	-492,3	78,69	839,43	0	0	0
140	2644,561	0	67,6	305,421	8,77193	510,281	0	97,61	685,781	0	0	0
0	2713	820,421	25,5	226,916	966,412	603,096	1684,2	202,25	423,509	0	0	0
1470	431,1869	0	29,4	252,517	54,0789	861,816	0	82,08	508,333	0	0	0
0	390,4579	0	28,2	205,701	425,965	459,061	0	101,4	404,825	0	0	1014,035
0	583,729	0	3,4	156,262	43,7719	161,158	0	204,21	43,5088	0	0	1666,667
0	2829	369,298	33,8	228,785	620	1324,02	720	62,53	142,982	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
1820	18514,19	3147,92	359,4	2448,5	1879,6	7044,13	5646,3	1269,66	1429,33	0	0	53729,82
Mehrwertsteuer												
0	72,70776	0	0	13,7383	7,29474	43,5032	0	0	46,0526	0	0	0
0	957	271,478	0	17,6963	7,33158	97,5935	0	0	74,814	0	0	0
0	56,26626	0	0	15,3894	1,54737	1,54737	0	0	83,4842	0	0	7146,877
0	77,47364	0	0	12,3056	2,49298	117,126	0	0	44,9842	0	0	0
0	230	365,228	0	15,2757	3,53684	93,1577	0	0	117,52	0	0	0
0	185,1193	0	0	21,3794	1,28807	71,4393	0	0	96,0093	0	0	0
0	189	114,859	0	15,8841	79,5777	84,4335	0	0	59,2912	0	0	0
0	30,18308	0	0	18,3832	7,57105	120,654	0	0	71,1667	0	0	0
0	27,33206	0	0	14,3991	59,6351	64,2666	0	0	56,6754	0	0	141,9649
0	40,86103	0	0	10,9383	6,12807	22,5621	0	0	6,09123	0	0	233,3333
0	198	51,7018	0	16,015	86,6	185,362	0	0	20,0175	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	2063,943	803,267	0	171,395	263,144	901,648	0	0	676,107	0	0	7522,175
Brutto												
0	1111,39	0	12,6	210	59,4	354,24	2368,7	132,65	375	0	0	0
210	2992	2210,61	50,6	270,5	59,7	794,69	432	85,97	609,2	0	0	0
0	860,07	0	27,7	235,1	12,6	616,39	0	135,84	679,8	0	0	58196
0	1184,24	0	40,8	188,1	20,3	953,74	933,7	86,43	366,3	0	0	0
0	3522	2974	39,8	233,5	23,8	758,57	-492,3	78,69	956,95	0	0	0
140	2829,68	0	67,6	326,8	10	581,72	0	97,61	784,79	0	0	0
0	2903	935,28	26,5	242,8	647,99	687,53	1684,2	202,25	422,5	0	0	0
1470	461,37	0	29,4	281	61,65	982,47	0	82,08	579,5	0	0	0
0	417,79	0	28,2	220,1	485,6	523,33	0	101,4	461,5	0	0	1156
0	624,59	0	3,4	167,2	49,9	183,72	0	204,21	49,6	0	0	1900
0	3027	421	33,8	244,8	706,6	1509,38	720	62,53	163	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
1820	19933,13	6540,89	359,4	2619,9	2142,74	7945,78	5646,3	1269,66	5505,44	0	0	61252

Excel verkleinert die Tabelle (natürlich können Sie sie mit dem *Zoom*-Symbol wieder vergrößern) und zeigt außer den eingefügten Seitenwechseln (dicke horizontale bzw. vertikale Linien) zusätzlich auch die Reihenfolge der einzelnen Seiten im Ausdruck an.

Die Seitenumbruchlinien können Sie mit der Maus packen und verschieben, zum Beispiel die horizontale Linie entsprechend der Abbildung zwischen die Zeilen 41 und 42 ziehen und die vertikale Linie zwischen die Spalten F und G.

Die Inhalte der nun ausgedruckten Blätter (vorausgesetzt, im Register *Tabelle* ist unter *Seitenreihenfolge* die Option *Seiten nach rechts, dann nach unten* selektiert, da sich die Ausgabereihenfolge sonst entsprechend ändert):

- ➔ Blatt 1: der Bereich A1:F41
- ➔ Blatt 2: der Bereich A42:F55 (die Tabelle endet unten bei Zeile 55)
- ➔ Blatt 3: der Bereich G1:M41 (die Tabelle endet rechts bei Spalte M)
- ➔ Blatt 4: der Bereich G42:M55

Mit ANSICHT|NORMAL schalten Sie wieder die Normalansicht ein.

Das Ziehen an der Umbruchlinie in der Seitenumbruch-Vorschau ist eine vereinfachte Version des Befehls EINFÜGEN|SEITENUMBRUCH, den Sie ebenfalls zum manuellen Festlegen von Seitenwechseln verwenden können. Dieser Befehl fügt immer *oberhalb der aktiven Zelle einen horizontalen und links von ihr einen vertikalen* Seitenumbruch ein.

*Seitenwechsel
einfügen*

Um wie geschehen den horizontalen Umbruch zwischen die Zeilen 41 und 42 zu verlegen und den vertikalen Umbruch zwischen die Spalten F und G, hätten Sie daher die Zelle G42 selektieren und EINFÜGEN|SEITENUMBRUCH wählen können.

Ist vor Anwahl von EINFÜGEN|SEITENUMBRUCH eine beliebige Zelle von Spalte A aktiv, befindet sich also links keine weitere Zelle, so wird nur ein horizontaler Umbruch eingefügt.

Hätten Sie beispielsweise vor der Wahl von EINFÜGEN|SEITENUMBRUCH die Zelle A42 (»Brutto«) selektiert, würde der horizontale Umbruch daher ebenfalls zwischen Zeile 41 und Zeile 42 erfolgen, der von Excel selbstständig eingefügte vertikale Umbruch bliebe jedoch unverändert.

Ist vor der Anwahl eine Zelle von Zeile 1 aktiv, befindet sich also darüber keine weitere Zelle, wird genau umgekehrt nur ein vertikaler Umbruch eingefügt.

Hätten Sie G1 selektiert, würde der vertikale Umbruch daher ebenfalls zwischen Spalte F und Spalte G eingefügt. Diesmal bliebe jedoch der zuvor von Excel eingefügte horizontale Umbruch unverändert.

Manuelle Seitenwechsel können Sie mit dem Befehl EINFÜGEN|SEITENUMBRUCH AUFHEBEN wieder entfernen, egal ob Sie die Seitenwechsel in der Seitenumbruch-Vorschau oder mit EINFÜGEN|SEITENUMBRUCH einfügten.

*Manuellen
Umbruch
aufheben*

Um einen horizontalen Seitenumbruch wieder aufzuheben, selektieren Sie irgendeine Zelle unmittelbar unterhalb der Umbruchlinie und wählen EINFÜGEN|SEITENUMBRUCH AUFHEBEN.

Um einen vertikalen Umbruch aufzuheben, selektieren Sie zuvor eine Zelle unmittelbar rechts neben der Umbruchlinie.

Und um einen kombinierten Umbruch aufzuheben, selektieren Sie jene Zelle, die sich rechts neben der vertikalen und unmittelbar unterhalb der horizontalen Umbruchlinie befindet.

8 Arbeitsmappen und Vorlagen

Bisher kennen Sie nur die Grundlagen des Umgangs mit Arbeitsmappen. Sie wissen, dass Mappen Blätter enthalten (meist Tabellen) und wie ein bestimmtes Blatt aktiviert wird.

Das folgende Kapitel erläutert die Feinheiten, die Sie speziell dann kennen sollten, wenn Sie mehrere Mappen nicht exklusiv, sondern zusammen mit anderen Anwendern benutzen, die auf Ihren Rechnern im gleichen Moment wie Sie selbst damit arbeiten. Bei derartigem Teamwork an Excel-Arbeitsmappen können Sie

- ➔ eine Ihrer Arbeitsmappen an einen im Netzwerk mit Ihnen verbundenen Kollegen senden bzw. eine von einem solchen Kollegen erhaltene Mappe nach Begutachtung an den nächsten Kollegen weiterleiten
- ➔ Arbeitsmappen freigeben, damit nicht nur Sie selbst, sondern gleichzeitig(!) auch Ihre Kollegen damit arbeiten können
- ➔ gezielt festlegen, welche (von Ihnen selbst oder einem gleichzeitig damit arbeitenden Kollegen vorgenommene) Änderungen Excel protokollieren und anzeigen soll – und diese Änderungen gegebenenfalls im Nachhinein einzeln wieder verwerfen
- ➔ und Kopien der gleichen Mappe auf mehreren Rechnern benutzen (z.B. auf dem Desktop im Büro und auf dem Notebook unterwegs) und diese Kopien bei Bedarf zusammenführen, um alle Änderungen an der einen Mappe in die andere übertragen zu lassen und umgekehrt, sodass anschließend beide Kopien wieder auf dem aktuellen Stand sind.

Anschließend erläutere ich, wie mehrere Blätter einer Arbeitsmappe zu Gruppen zusammengefasst und gemeinsam editiert werden, damit beispielsweise das Löschen eines Bereichs in der Tabelle »Test« dazu führt, dass dieser Bereich auch in allen anderen Blättern dieser Gruppe gelöscht wird.

Sie lernen, wie Blätter umbenannt oder ausgeblendet werden, wie Sie in einer Mappe Blätter verschieben, kopieren, einfügen oder löschen.

Zusätzlich erläutere ich die Feinheiten des Fenster-Handlings, beispielsweise wie Sie mehrere Fenster einrichten, in denen unterschiedliche Teile der glei-

chen Tabelle sichtbar sind, oder wie Sie das Wegscrollen von Zeilen- und Spaltenüberschriften beim Blättern in einer Tabelle verhindern können.

Anschließend geht es um externe Bezüge. Ich zeige Ihnen, wie Sie in Formeln Bezüge verwenden, die sich auf Zellen in einer anderen Tabelle beziehen – egal, ob sich die betreffende Tabelle in der gleichen oder in einer anderen Arbeitsmappe befindet, die möglicherweise noch nicht einmal auf dem gleichen Rechner gespeichert ist, sondern auf einem FTP-Server im Internet.

Im letzten Abschnitt erfahren Sie, welche Möglichkeiten Vorlagen bieten, wie Sie damit – basierend auf einer einmal gestalteten Tabelle – per Knopfdruck jederzeit neue Tabellen des gleichen Typs erzeugen können und wie Sie die mit Excel selbst ausgelieferten Vorlagen benutzen.

8.1 Teamarbeit – Arbeitsmappen gemeinsam benutzen

8.1.1 Mappen versenden und weiterleiten

Ist Outlook, Outlook Express oder Microsoft Exchange installiert, gibt es den Befehl DATEI|SENDEN AN, dessen Unterbefehle Ihnen ermöglichen, eine Kopie der aktiven Arbeitsmappe

- ➔ an mehrere Empfänger zu senden, das heißt sie allen Empfängern gleichzeitig zu schicken
- ➔ oder aber die Mappe weiterzuleiten, das heißt sie dem ersten Empfänger zu schicken, der sie wiederum an den nächsten weiterleitet und so weiter.



Die einfachste Technik zum Versenden einer Mappe besteht darin, DATEI|SENDEN AN|E-MAILEMPFÄNGER zu wählen oder einfach auf das abgebildete Symbol zu klicken.

Enthält die aktuelle Mappe nur ein Blatt mit Daten und sind die restlichen Blätter leer, wird Outlook Express nun vorübergehend in Excel integriert: unterhalb der Excel-Symbolleiste erscheinen die Outlook Express-Symbolleiste und darunter die Outlook Express-Zeilen zur Angabe/Auswahl des bzw. der Empfänger (Abbildung 8.1).

Sie wählen wie gewohnt den/die Empfänger aus und klicken auf *Dieses Blatt senden*. Daraufhin übergibt Excel das Blatt als neue Nachricht an Outlook Express, wo es sich nun im Ausgangskorbchen befindet.

Wollen Sie das Blatt doch nicht versenden, klicken Sie erneut auf das E-Mail-Symbol und Outlook verschwindet wieder aus dem Excel-Fenster.

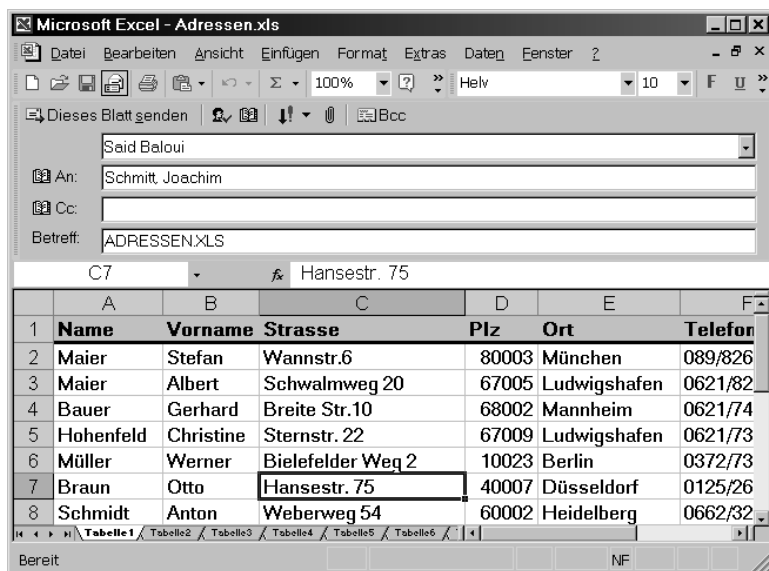


Abbildung 8.1:
Outlook Express-
Integration

Das Blatt wird übrigens nicht als Anlage an die neue Nachricht angehängt, sondern als Objekt in den Textbereich (Textkörper) selbst integriert.



Wollen Sie das Blatt nicht in den Textbereich einfügen, sondern es auf die konservative Weise einer Nachricht als Anlage beifügen, wählen Sie stattdessen DATEI|SENDEN AN|E-MAILEMPFÄNGER (ALS ANLAGE)... bzw. klicken auf das zugehörige Symbol.



Daraufhin erscheint Outlook Express in voller Größe und das Blatt wird als Anlage an eine neu erstellte Nachricht angehängt. Sie müssen nur noch den Empfänger auswählen, bei Bedarf einen erläuternden Text eintippen und die Nachricht versenden.

Gibt es in der aktuellen Mappe mehrere Blätter, die Daten enthalten, fragt Excel bei Verwendung von DATEI|SENDEN AN|E-MAILEMPFÄNGER bzw. des zugehörigen Symbols zunächst nach, ob Sie das momentan aktive Blatt in den Textkörper einer neuen Nachricht einfügen oder aber die gesamte Mappe als Anlage versenden wollen. Was mit diesen beiden Optionen gemeint ist, wissen Sie ja nun.

Mit dem Befehl DATEI|SENDEN AN|E-MAILEMPFÄNGER (ZUR ÜBERARBEITUNG)... können Sie das Dokument gezielt zur Begutachtung, Kommentierung und Änderung versenden. Sie werden gefragt, ob Sie eine freigegebene Version der Mappe speichern wollen. Bejahen Sie das, wird diese freigegebene Kopie mit dem Kommentar »Bitte überarbeiten...« an eine E-Mail angehängt.



Nach dem Öffnen der Mappe werden beim Empfänger automatisch die Überarbeitungstools eingeblendet. Nach beendetem Editieren klickt der Empfänger auf *Bearbeitung zurücksenden...*, um die Mappe an Sie zurückzusenden (und Sie können anschließend gegebenenfalls wie in den folgenden Kapiteln beschrieben die von ihm vorgenommenen Änderungen verfolgen und individuell annehmen oder ablehnen).



Wollen Sie eine bestimmte Mappe immer wieder an die gleichen Empfänger schicken, empfiehlt es sich, zunächst einen Verteiler zu erstellen, der festlegt, an welche Empfänger die Mappe in welcher Reihenfolge weitergeleitet wird.

Dazu wählen Sie `DATEI|SENDEN AN|VERTEILEREMPFÄNGER...` und definieren im zugehörigen Dialogfeld die Empfänger der Mappe in der gewünschten Reihenfolge.

Die Schaltfläche *Adresse...* ermöglicht jeweils die Auswahl eines weiteren Empfängers aus dem Adressbuch und *Entfernen* das Entfernen eines zuvor ausgewählten Empfängers aus der Liste.

Mit den nach oben/unten gerichteten Pfeilen können Sie die Reihenfolge der Empfänger verändern.

Sind Sie mit der Empfängerauswahl fertig, beenden Sie die Erstellung des Verteilers mit der Schaltfläche *Verteiler hinzufügen*.

Haben Sie auf diese Weise für die aktive Mappe einen Verteiler erstellt oder handelt es sich bei dieser Mappe genau umgekehrt um eine an Sie weitergeleitete Mappe, ist der Befehl `DATEI|SENDEN AN|VERTEILEREMPFÄNGER...` durch `DATEI|SENDEN AN|WEITERE EMPFÄNGER...` ersetzt, mit dem Sie die Empfängerliste und andere Weiterleitungs-Optionen nachträglich bearbeiten können.

Der Befehl `DATEI|SENDEN AN|MAILEMPFÄNGER (ALS ANLAGE)...` ist nun durch `DATEI|SENDEN AN|NÄCHSTER EMPFÄNGER...` ersetzt, der die Mappe an den nächsten im Verteiler festgelegten Empfänger schickt.

Haben Sie selbst den Verteiler erstellt, ist das der erste von Ihnen definierte Empfänger; haben Sie genau umgekehrt eine Mappe mit Verteiler erhalten, schickt dieser Befehl die Mappe dem laut Verteiler nach Ihnen kommenden Empfänger.

8.1.2 Arbeitsmappen freigeben

Versuchen Sie, im Netzwerk eine Mappe zu öffnen, die bereits von einem anderen Anwender geöffnet wurde, so erscheint die Meldung »Dateireservierung«, die bedeutet, dass die betreffende Mappe bereits bearbeitet wird und von Ihnen nicht geöffnet werden kann.

Sie haben nun die Option, die Mappe schreibgeschützt zu öffnen, den anderen Anwender zu benachrichtigen (auf dass er die Mappe schließt, damit sie anschließend von Ihnen geöffnet werden kann) oder den Versuch abubrechen.

Freigegebene Arbeitsmappen können jedoch von mehreren Anwendern gleichzeitig bearbeitet werden – was mit möglichst wenigen Einschränkungen jedoch nur mit Mappen möglich ist, die im aktuellen Excel-Format gespeichert sind (alte Mappen also bitte zuvor in diesem Format speichern, sonst macht Excel das automatisch, was etwas verwirrend ist).

Um eine Mappe freizugeben, wählen Sie **EXTRAS|ARBEITSMAPPE FREIGEBEN...** und aktivieren im Register *Status* des zugehörigen Dialogfelds *Bearbeitung ... zulassen ...* (Abbildung 8.2).

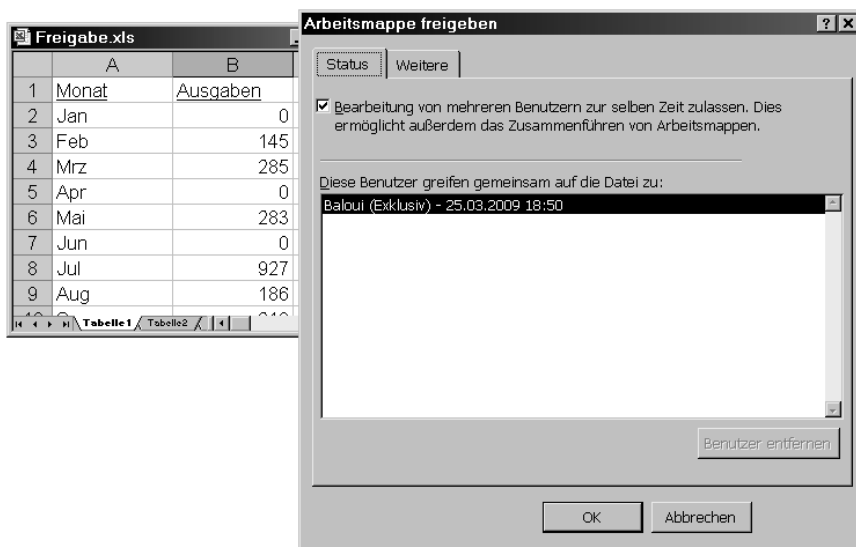


Abbildung 8.2:
Arbeitsmappe
freigeben

Nach dem Schließen des Dialogfelds wird die Mappe gespeichert und in der Titelleiste erscheint der zusätzliche Hinweis »[Freigegeben]«, ebenso wie bei jedem anderen Benutzer, der diese Mappe anschließend öffnet und der sie nun gleichzeitig mit Ihnen oder weiteren Anwendern bearbeiten kann.

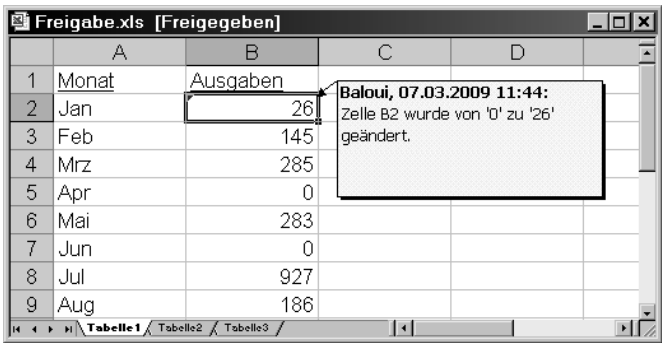
Diese Benutzer werden übrigens alle angezeigt, wenn Sie den Befehl erneut wählen, sodass Sie sich jederzeit darüber informieren können, wer außer Ihnen gerade mit der Mappe arbeitet.

Ab jetzt wird die freigegebene Mappe jedesmal, wenn Sie sie speichern, automatisch aktualisiert.

Angenommen, B2 enthielt den Wert 0, ein anderer Benutzer trug in B2 jedoch den Wert 26 ein und speicherte die Mappe inzwischen. Speichern Sie Ihre Mappe, wird sie dabei automatisch aktualisiert und der neue Inhalt von B2 erscheint darin – zusammen mit einem Hinweis, dass die Mappe beim Speichern aktualisiert wurde.

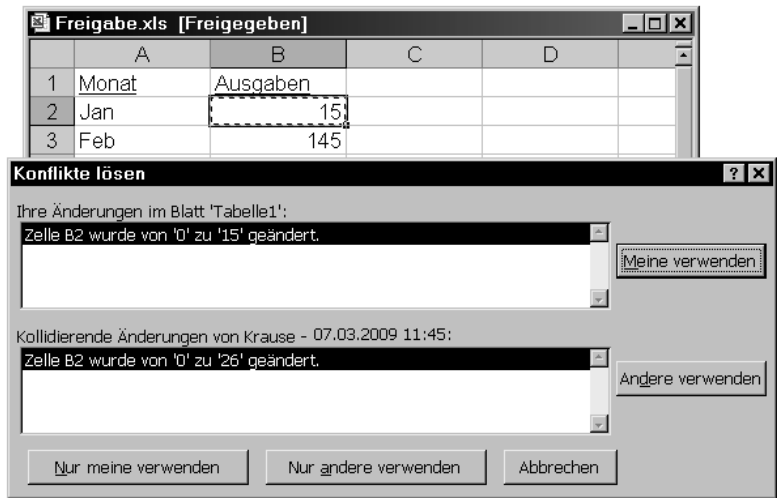
Die aktualisierten Zellen werden durch einen dünnen Rahmen hervorgehoben. Ist der Cursor auf einer aktualisierten Zelle, teilt Ihnen Excel in Form einer Notiz mit, wer den Zelleninhalt wann veränderte und welchen neuen Wert er eintrug (Abbildung 8.3):

Abbildung 8.3:
Aktualisierung



Möglicherweise erscheint beim Speichern einer freigegebenen Mappe stattdessen der folgende Hinweis (Abbildung 8.4):

Abbildung 8.4:
Kollision



Ursache dafür ist eine Kollision. In diesem Beispiel passierte Folgendes: B2 enthielt ursprünglich den Wert 0. Der Benutzer »Krause« änderte das; er trug den Wert 26 ein (unteres Listenfeld) und speicherte die Mappe danach.

Sie selbst änderten jedoch ebenfalls den Inhalt dieser Zelle und gaben den Wert 15 ein (oberes Listenfeld), ohne zu wissen, dass Krause den ursprünglichen Wert 0 bereits veränderte.

Die Frage ist nun, welcher Wert endgültig in der Datei gespeichert werden soll, Ihre 15 oder die von Krause eingegebene 26: *Meine verwenden* speichert Ihre Änderung, *Andere verwenden* dagegen die Änderung von Krause.

Auf die gleiche Weise wird nach der Beantwortung dieser Frage jeder weitere Konflikt nach der gewünschten Lösungsart abgefragt. Entweder beantworten Sie jede Frage einzeln oder Sie geben Excel mit einer der beiden unteren Schaltflächen bekannt, wie dieser und alle anderen aufgetretenen Konflikte einheitlich behandelt werden sollen.

Im Register Weitere von EXTRAS|ARBEITSMAPPE FREIGEBEN... können Sie das Verhalten von Excel bei der gemeinsamen Bearbeitung freigegebener Mappen genauer festlegen, unter anderem, ob Kollisionen nur beim Speichern angezeigt werden sollen oder aber in regelmäßigen Zeitabständen, beispielsweise alle zehn oder 15 Minuten.



8.1.3 Änderungen verfolgen und rückgängig machen

Wie Sie soeben sahen, werden Änderungen, die andere Anwender an der gerade von Ihnen bearbeiteten freigegebenen Mappe vornahmen, von Excel erkannt, wenn Sie Ihre Mappe speichern, und durch einen Rahmen um die aktualisierten Zellen herum markiert und durch eine Notiz hervorgehoben.

Wollen Sie jedoch nicht ständig auf alle Änderungen hingewiesen werden, sondern nur auf ganz bestimmte (oder genau umgekehrt nicht nur auf die Änderungen durch andere Benutzer, sondern auch auf Ihre eigenen!), so wählen Sie EXTRAS|ÄNDERUNGEN NACHVERFOLGEN|ÄNDERUNGEN HERVORHEBEN... (Abbildung 8.5).

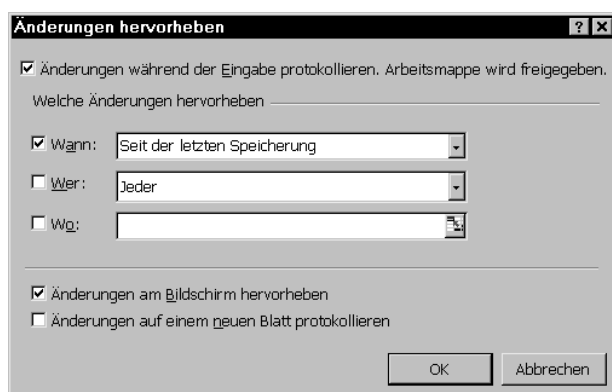


Abbildung 8.5:
Zu verfolgende
Änderungen fest-
legen

Das Kontrollkästchen *Änderungen während der Eingabe protokollieren...* muss aktiviert sein. Sind die Kontrollkästchen der drei folgenden Listenfelder deaktiviert, werden wie zuvor alle Änderungen ständig hervorgehoben.

Stattdessen können Sie jedoch mit der Option *Seit der letzten Speicherung* des obersten Listenfelds festlegen, dass beispielsweise nach jedem Speichern der Mappe alle Hervorhebungen verschwinden und die Verfolgung von vorn beginnen soll oder aber mit *Seit...*, dass nur Änderungen ab einem bestimmten Datum hervorgehoben werden, wobei das aktuelle Datum vorgegeben wird, jedoch individuell geändert werden kann, oder aber festlegen, dass alle *Noch nicht überprüft*-Änderungen hervorgehoben werden, also nur Änderungen, die noch nicht wie gleich erläutert von Ihnen begutachtet wurden.

Im mittleren Listenfeld legt *Jeder* fest, dass Änderungen beliebiger Benutzer hervorgehoben werden. Mit *Jeder außer mir* würden alle außer Ihren eigenen Änderungen hervorgehoben.

Zusätzlich sind in diesem Listenfeld die Namen aller anderen Benutzer enthalten, die gerade mit der Mappe arbeiten; wählen Sie einen davon aus, werden nur dessen Änderungen hervorgehoben.

Unter *Wo* können Sie einen Bereich wie A\$5:\$C\$10 angeben, auf den das Hervorheben von Änderungen beschränkt werden soll.

OK schließt das Dialogfeld, speichert die aktive Mappe und gibt sie automatisch frei, wenn sie noch nicht freigegeben ist.

Andere Anwender können sie nun wie zuvor erläutert gleichzeitig benutzen und wenn Sie die Mappe speichern, werden Sie auf die von diesen Anwendern vorgenommenen Änderungen hingewiesen, sofern sie unter die zuvor von Ihnen festgelegten Verfolgungsbedingungen fallen, beispielsweise im Bereich \$A\$5:\$C\$10 erfolgten.

Egal, ob Sie zuvor mit EXTRAS|ÄNDERUNGEN NACHVERFOLGEN|ÄNDERUNGEN HERVORHEBEN... näher festlegten, welche Änderungen hervorgehoben werden sollen oder einfach nur die aktive Mappe freigaben; in jedem Fall können Sie danach mit EXTRAS|ÄNDERUNGEN NACHVERFOLGEN|ÄNDERUNGEN ANNEHMEN ODER ABLEHNEN... die Änderungen einzeln einsehen und akzeptieren oder rückgängig machen (Abbildung 8.6).

Im zugehörigen Dialogfeld wählen Sie wie erläutert, in welchem Bereich wann und von wem vorgenommene Änderungen Sie einsehen wollen. Anschließend wird jede dieser Änderungen einzeln aufgelistet. Sie können eine beliebige selektieren und mit den gleichnamigen Schaltflächen annehmen oder ablehnen oder aber alle diese Änderungen annehmen bzw. ablehnen (Abbildung 8.7).

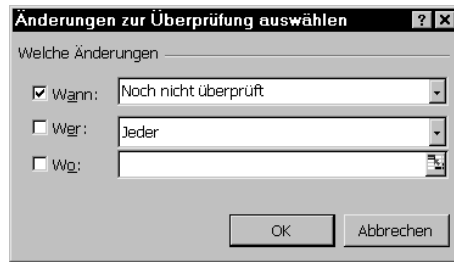


Abbildung 8.6:
Einzusehende
Änderungen
festlegen

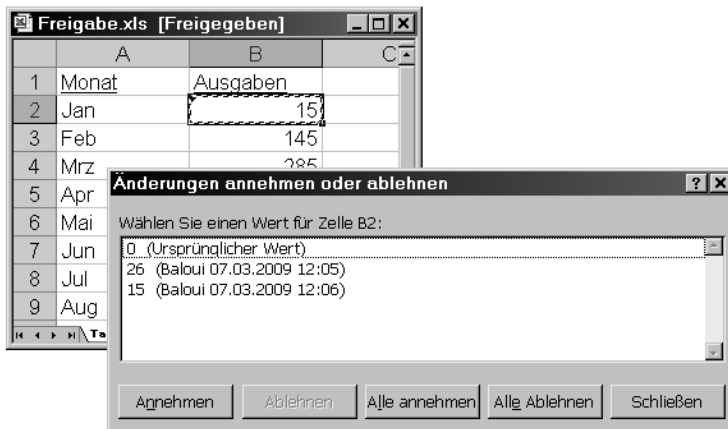


Abbildung 8.7:
Änderungen
akzeptieren
oder verwerfen

Die eingesehenen und akzeptierten Änderungen sind nun endgültig und können nicht mehr rückgängig gemacht werden.

Wollen Sie sich alle Änderungen auch im Nachhinein noch in Ruhe ansehen können, wählen Sie EXTRAS|ÄNDERUNGEN NACHVERFOLGEN|ÄNDERUNGEN HERVORHEBEN..., deaktivieren die Kontrollkästchen aller drei Listenfelder und aktivieren das Kontrollkästchen Änderungen auf einem neuen Blatt protokollieren.



Excel fügt daraufhin eine Tabelle namens *Änderungsprotokoll* in die Mappe ein und protokolliert darin alle Änderungen.

Die resultierende Liste informiert darüber, welcher Benutzer wann wo etwas änderte, wie der alte Wert war, welchen neuen Wert er eingab etc. (Abbildung 8.8).

Die beiden Befehle EXTRAS|ÄNDERUNGEN NACHVERFOLGEN|ÄNDERUNGEN HERVORHEBEN... und EXTRAS|ÄNDERUNGEN NACHVERFOLGEN|ÄNDERUNGEN ANNEHMEN ODER ABLEHNEN... können Sie auch anwenden, wenn Sie der alleinige Benutzer einer Mappe sind, um Ihre eigenen Änderungen verfolgen zu lassen und nachträglich überprüfen und gegebenenfalls wieder rückgängig machen zu können.



Abbildung 8.8:
Änderungsprotokoll

	A	B	C	D	E	F	G	H	I
1	Aktionsnum	Datum	Zeit	Wer	Änderung	Blatt	Bereich	Neuer Wert	Alter Wert
2	1	07.03.09	11:40	Baloui	Änderung in der Zelle	Tabelle1	B2	26	0
3	2	07.03.09	11:50	Baloui	Änderung in der Zelle	Tabelle1	B2	15	26
4	3	07.03.09	11:50	Baloui	Änderung in der Zelle	Tabelle1	B2	233	283
5									
6	Das Protokoll hört mit den Änderungen, die am 07.03.2009 um 11:50 gespeichert wurden, auf.								
7									

8.1.4 Arbeitsmappen zusammenführen

Mit einer freigegebenen Arbeitsmappe können nicht nur im Netzwerk mehrere Anwender gleichzeitig arbeiten. Ebenso gut ist es möglich, eine Kopie der betreffenden Mappe auf einem Notebook mitzunehmen und zu bearbeiten.

Alle Änderungen an der Mappe können später jederzeit mit EXTRAS|ARBEITSMAPPEN VERGLEICHEN UND ZUSAMMENFÜHREN... wieder in einer einzigen Mappe zusammengeführt werden.

Nehmen wir an, die Mappe BUCHFRG.XLS wird freigegeben. A und B kopieren beide die Mappe unter dem Namen A_BUCHFRG.XLS bzw. B_BUCHFRG.XLS auf ihre Notebooks und arbeiten am Wochenende mit ihren Kopien zu Hause. Dort tragen sowohl A als auch B zusätzliche Daten in die beiden Mappen ein bzw. ändern vorhandene Daten.

Am nächsten Tag sollen beide Mappen im Büro miteinander abgeglichen werden und die beiden Notebooks werden wieder per Netzwerk miteinander verbunden.

Dazu öffnet einer der beiden Anwender seine Mappe, beispielsweise A seine Mappe A_BUCHFRG.XLS, und wählt EXTRAS|ARBEITSMAPPEN VERGLEICHEN UND ZUSAMMENFÜHREN... .

Ein *Dateiauswahl*-Dialogfeld erscheint, in dem A die Mappe auswählen soll, die mit der aktuellen Mappe zusammengeführt werden soll. A wählt entsprechend die Mappe B_BUCHFRG.XLS aus, die sich auf dem Notebook von B befindet.

Alle Änderungen, die B an B_BUCHFRG.XLS vornahm, werden daraufhin in As Mappe A_BUCHFRG.XLS übertragen und analog zu Bild 8.3 hebt Excel von B geänderte Zellen hervor und teilt mit, welche Änderungen B vornahm.

A kann anschließend mit EXTRAS|ÄNDERUNGEN NACHVERFOLGEN|ÄNDERUNGEN ANNEHMEN ODER ABLEHNEN... jede einzelne dieser Änderungen separat akzeptieren oder rückgängig machen.

8.2 Gruppen bilden und editieren

Oft sind mehrere in einer Mappe enthaltene Tabellen völlig identisch aufgebaut, zum Beispiel Mappen, die die Daten eines Jahres enthalten, unterteilt in zwölf Monate, wobei für jeden Monat eine eigene Tabelle verwendet wird. Nehmen Sie als Beispiel folgende Tabelle (Abbildung 8.9).

	A	B	C	D	E	F	G	H	I
1		Einnahmen			Ausgaben				
2		Bücher Programme		Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12				Netto	MWSt				
13		- Steuerpflichtige Umsätze mit 16%		3327,16	532,34				
14		- Steuerpflichtige Umsätze mit 7%		1745,40	122,18				
15	Gesamt			5072,56	654,52				
16	Vorsteuerbeträge				462,95				
17	Umsatzsteuervorauszahlung				191,57				
18									

Abbildung 8.9:
Monats-Tabelle
einer Buchführung

Diese Tabelle enthält die Buchführung eines Freiberuflers oder Selbstständigen für den Monat Januar. Nachdem Sie die Texte und Formeln eingegeben haben, benötigen Sie elf Kopien der Tabelle für die Monate Februar bis Dezember.

Eine einfache, aber umständliche Möglichkeit besteht darin, den Bereich A1:I17 zu selektieren, KOPIEREN zu wählen, in der zweiten Tabelle A1 zu aktivieren und den kopierten Bereich mit EINFÜGEN|ZELLEN... dort einzufügen und ihn danach auf die gleiche Weise in die Tabellen 3 bis 12 zu kopieren.

Allgemein: Sie können problemlos tabellenübergreifende Editierungen vornehmen, indem Sie in Tabelle A eine Editieraktion ausführen (beispielsweise einen Bereich kopieren), danach Tabelle B aktivieren und dort die zugehörige Aktion durchführen (den kopierten Bereich einfügen). Dabei ist es egal, ob sich die beiden Tabellen A und B in der gleichen oder in verschiedenen Arbeitsmappen befinden.



Allerdings gibt es spezielle Techniken zur Manipulation von Arbeitsmappen, die unsere Aufgabe wesentlich erleichtern, vor allem durch die Bildung einer Arbeitsblattgruppe.

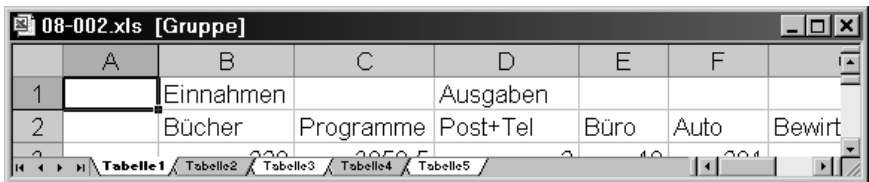
Blatt hinzufügen

Mit Hilfe der Arbeitsmappen-Registerkartenleiste am unteren Rand einer Arbeitsmappe können Sie beliebig viele der darin enthaltenen Blätter selektieren und zu einer Gruppe zusammenfassen.

Um zusätzlich zum momentan aktiven Blatt ein weiteres zu selektieren, klicken Sie in der Registerkartenleiste auf dessen Namen und halten dabei die **[Strg]**-Taste gedrückt.

Um beispielsweise die Blätter »Tabelle1«, »Tabelle3« und »Tabelle5« zu einer Gruppe zusammenzufassen, gehen Sie so vor: Aktivieren Sie das erste dieser Blätter, »Tabelle1«. Halten Sie die **[Strg]**-Taste gedrückt und klicken Sie auf die Registerkarte von »Tabelle3« und danach auf die von »Tabelle5« (Abbildung 8.10).

Abbildung 8.10:
Mehrere Blätter
selektieren



Die Registerkarten der drei Tabellen werden hervorgehoben und in der Titelleiste befindet sich der Hinweis »[Gruppe]«. Beides weist darauf hin, dass die drei Tabellen nun eine Gruppe bilden.

Die zuletzt aktive Tabelle ist weiter aktiv, worauf die Fettschrift des zugehörigen Blattnamens »Tabelle1« in der Registerkarte hinweist.

Blatt lösen

Sie können ein Blatt jederzeit wieder aus der Gruppe entfernen, indem Sie erneut bei gedrückter **[Strg]**-Taste auf seinen Namen klicken.

Klicken Sie bei gedrückter **[⇧]**-Taste auf die Registerkarte des momentan aktiven Blatts, so wird die Gruppenbildung vollständig aufgehoben.

Benachbarte Blätter

Die Selektion mehrerer benachbarter Blätter können Sie sich vereinfachen: Aktivieren Sie das erste oder letzte dieser Blätter und halten Sie die **[⇧]**-Taste gedrückt, während Sie auf den Namen jenes Blatts klicken, das sich am entgegengesetzten Ende befindet.

Um beispielsweise die Blätter »Tabelle2«, »Tabelle3« und »Tabelle4« zu selektieren, aktivieren Sie »Tabelle2«, drücken die **[⇧]**-Taste und klicken auf »Tabelle4« (Abbildung 8.11).

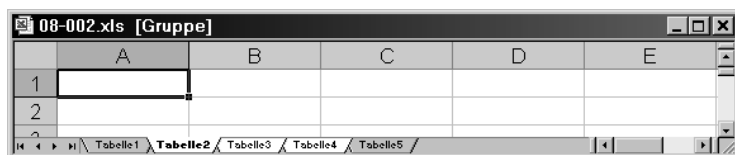


Abbildung 8.11:
Benachbarte Blätter
zu einer Gruppe
zusammenfassen

Das aktive Blatt können Sie wechseln, indem Sie die Registerkarte eines anderen zur aktuellen Gruppe gehörenden Blatts anklicken.

**Aktive Tabelle
wechseln**

Aktivieren Sie dagegen ein nicht selektiertes Blatt, verlassen Sie also die Gruppe, so wird die Selektion und damit die Gruppenbildung aufgehoben!

Jede Editierung, die Sie in einer beliebigen Tabelle einer Gruppe durchführen, führt Excel in gleicher Weise auch in den restlichen Tabellen der Gruppe aus:



- ➔ Verschieben Sie in einer der Tabellen einen Bereich, so wird er auch in den restlichen verschoben.
- ➔ Kopieren oder löschen Sie einen Bereich, so wird er auch in den restlichen Tabellen kopiert bzw. gelöscht.
- ➔ Selektieren Sie in einer der Tabellen einen Bereich und klicken Sie das *Fettschrift*-Symbol der Standardleiste an, so wird der betreffende Bereich in allen selektierten Tabellen fett formatiert.
- ➔ Sogar Eingaben werden in alle anderen selektierten Tabellen übernommen: Geben Sie in Zelle A5 von Tabelle 2 eine Zahl ein, so wird diese Zahl auch in die Zellen A5 aller anderen Tabellen der Gruppe eingetragen.

Sie könnten beispielsweise den Inhalt des Bereichs A1:C10 von Tabelle 1 in die Tabellen 2 bis 12 kopieren, indem Sie

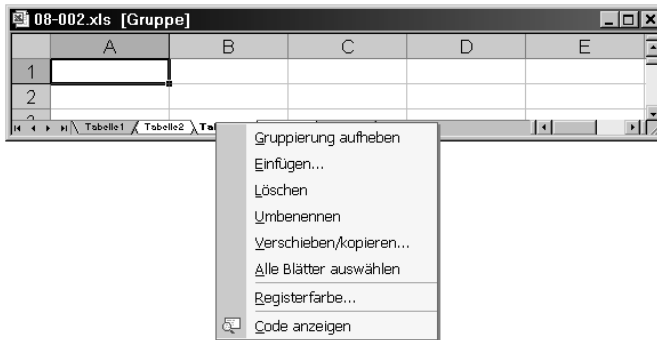
- ➔ in Tabelle 1 diesen Bereich selektieren und KOPIEREN wählen,
- ➔ danach die Tabellen 2 bis 12 zu einer Gruppe zusammenfassen und in irgendeiner dieser Tabellen die Zelle selektieren, die beim Einfügen als linke obere Ecke benutzt werden soll
- ➔ und mit EINFÜGEN|ZELLEN... den kopierten Bereich in alle selektierten Tabellen gleichzeitig an dieser Stelle einfügen.

8.3 Umbenennen, Aus-/Einblenden

Die wichtigsten Befehle zum Umgang mit Arbeitsmappenblättern sind im Kontextmenü enthalten, das sich öffnet, wenn Sie eine Registerkarte mit der rechten Maustaste anklicken.

Es zu öffnen ist auch dann möglich, wenn Sie zuvor mehrere Blätter selektierten: Klicken Sie eine der selektierten Registerkarten mit der rechten Maustaste an, so wird das betreffende Blatt aktiviert und das folgende Kontextmenü geöffnet (Abbildung 8.12).

Abbildung 8.12:
Register-Kontext-
menü



Die darin enthaltenen Befehle entsprechen verschiedenen Befehlen der Menüs BEARBEITEN, FORMAT oder EINFÜGEN, die ich auf den folgenden Seiten erläutere (abgesehen von den Befehlen CODE ANZEIGEN, der die Programmierungsumgebung Visual Basic aktiviert und das Code-Modul der betreffenden Tabelle anzeigt, vom Befehl ALLE BLÄTTER AUSWÄHLEN, der kein Menü-Äquivalent besitzt und alle Blätter der aktiven Mappe selektiert, vom Befehl GRUPPIERUNG AUFHEBEN, der Gruppenbildungen wieder aufhebt, und von REGISTERFARBE..., das ein Dialogfeld öffnet, in dem Sie eine Farbe auswählen können, mit der die Registerkarten unterstrichen werden).

Umbenennen

Jeder Tabelle einer Arbeitsmappe können Sie einen individuellen Namen geben. Dazu aktivieren Sie die betreffende Tabelle und wählen FORMAT|BLATT|UMBENENNEN oder doppelklicken einfach auf die Registerkarte einer Tabelle und können den Blattnamen daraufhin direkt in der Registerzunge editieren.

Tabellen aus-/ einblenden

FORMAT|BLATT|AUSBLENDEN blendet das momentan aktive Blatt aus (bzw. die gesamte Gruppe): Die zugehörige Registerkarte am unteren Fensterrand verschwindet und es ist nun kein Zugriff mehr auf dieses Blatt möglich.

Mit FORMAT|BLATT|EINBLENDEN... können Sie es wieder einblenden (bei Gruppen können die zugehörigen Tabellen nur einzeln wieder eingeblendet werden): Ein Dialogfeld öffnet sich, dessen Listenfeld die Namen aller Blätter der aktiven Mappe enthält, die ausgeblendet sind. Sie selektieren das gewünschte und blenden es mit OK wieder ein.

8.4 Löschen, Einfügen, Verschieben und Kopieren

BEARBEITEN|BLATT LÖSCHEN löscht das aktive Blatt beziehungsweise – wenn mehrere selektiert sind – alle selektierten Blätter. Ist die Mappe freigegeben, ist dieser Befehl verständlicherweise nicht anwählbar.

Löschen

Benötigen Sie in Ihrer Mappe nur die Tabelle »Jan«, können Sie die überflüssigen Blätter »Tabelle2« bis beispielsweise »Tabelle16« löschen, indem Sie sie selektieren (Registerkarte »Tabelle2« anklicken, danach bei gedrückter -Taste auf die Registerkarte »Tabelle16« klicken), und alle nun selektierten Tabellen gleichzeitig mit BEARBEITEN|BLATT LÖSCHEN entfernen (Abbildung 8.13).

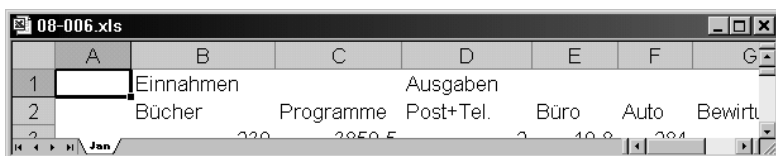


Abbildung 8.13:
Blätter löschen

EINFÜGEN|TABELLENBLATT fügt *unmittelbar vor dem aktiven Blatt* ein neues Tabellenblatt ein.

Einfügen

Benötigen Sie vor »Jan« eine zusätzliche Tabelle mit einer Jahresübersicht, aktivieren Sie die Tabelle »Jan« und wählen EINFÜGEN|TABELLENBLATT (Abbildung 8.14).

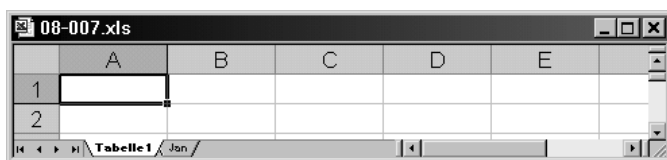


Abbildung 8.14:
Neue Tabelle einfügen

Um das aktive Blatt innerhalb einer Mappe zu verschieben, ziehen Sie die zugehörige Registerkarte bei gedrückter Maustaste zu jener, vor der es eingefügt werden soll, und lassen die Maustaste wieder los.

Verschieben

Soll sich »Tabelle1« hinter der Monats-Tabelle befinden, ziehen Sie die betreffende Registerkarte daher rechts neben die Tabelle »Jan« (Abbildung 8.15).

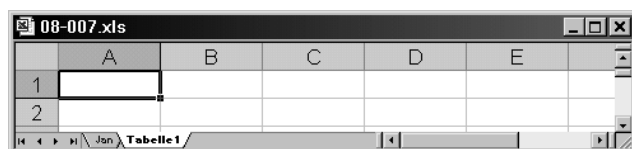


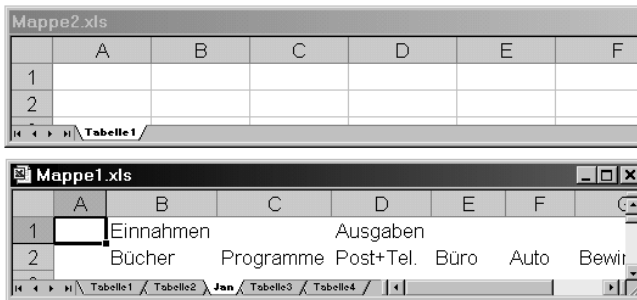
Abbildung 8.15:
Tabelle verschieben

Sie können auch mehrere Blätter gleichzeitig verschieben. Dazu selektieren Sie sie und verschieben anschließend die Registerkarte *irgendeines* der selektierten Blätter.

Um Blätter in eine andere Mappe zu verschieben, selektieren Sie sie und ziehen irgendeine der zugehörigen Registerkarten zu jener Position in der Registerleiste der anderen Mappe, an der die Blätter eingefügt werden sollen.

In der folgenden Abbildung wurde auf diese Weise die Tabelle »Jan« (siehe vorhergehende Abbildung) in eine andere Mappe verschoben (Abbildung 8.16).

Abbildung 8.16:
Blatt in andere
Mappe verschieben



Ziehen Sie das Blatt stattdessen in den leeren Raum außerhalb jeder Arbeitsmappe, so wird es aus der aktuellen Mappe entfernt und in eine automatisch neu angelegte Mappe eingefügt, die anschließend nur dieses eine Blatt enthält.

Kopieren

Um Blätter innerhalb einer Mappe oder in eine andere Mappe zu kopieren, gehen Sie genauso vor wie beim Verschieben, halten jedoch während des Ziehens die **[Strg]**-Taste gedrückt. Das ermöglicht Ihnen, Blätter

- ➔ innerhalb der Mappe zu kopieren,
- ➔ durch Ziehen zu einer anderen Mappe in diese zu kopieren
- ➔ oder durch Ziehen zu einem Punkt außerhalb einer Mappe eine neue Arbeitsmappe anlegen, in die die Blätter kopiert werden.

Enthält die Mappe, in die die Kopie eingefügt wird, bereits ein gleichnamiges Blatt (was immer der Fall ist, wenn Sie *innerhalb* einer Mappe kopieren), erhält die Kopie zwar den gleichen Namen, zur Unterscheidung jedoch eine zusätzliche Nummer.

Die Kopie von »Jan« heißt dann beispielsweise »Jan (2)« und eine weitere Kopie »Jan (3)« (Abbildung 8.17).

	A	B	C	D	E	F
1		Einnahmen		Ausgaben		
2		Bücher	Programme	Post+Tel.	Büro	Auto

Abbildung 8.17:
Blatt kopieren

Verschieben/Kopieren per Tastatur: BEARBEITEN|BLATT VERSCHIEBEN/KOPIEREN... öffnet ein Dialogfeld, in dem Sie angeben, vor welche Tabelle welcher Mappe die selektierten Blätter verschoben beziehungsweise – wenn Sie Kopie erstellen aktivieren – kopiert werden sollen (Abbildung 8.18).

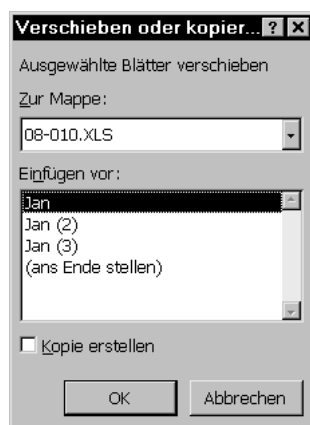


Abbildung 8.18:
Blatt Verschieben/
Kopieren-Dialogfeld

Der letzte Eintrag im Listenfeld lautet übrigens *(ans Ende stellen)* und fügt das Blatt ganz rechts in der Registerkartenleiste ein.

Unser Ausgangsproblem, die Januar-Tabelle elfmal zu kopieren, lässt sich inzwischen sehr einfach lösen: durch mehrfaches Kopieren dieser Tabelle innerhalb der Mappe mit den soeben gezeigten Techniken.

Anschließend benennen Sie die Kopien in »Feb«, »Mae« etc. um, ziehen die Jahresübersichts-Tabelle nach vorn, vor »Jan«, nennen sie »Summe«, und haben eine wunderschöne kleine Buchführung (Abbildung 8.19).

Sie müssen allerdings von der Kleinigkeit abgesehen, dass die Tabellen »Feb« bis »Dez« die gleichen Daten wie die Tabelle »Jan« enthalten, da beim Kopieren dieser Tabelle nicht nur die Struktur kopiert wurde, sondern natürlich auch alle darin enthaltenen Daten!

Also selektieren Sie die Tabellen »Feb« bis »Dez« und fassen sie zu einer Gruppe zusammen. Danach aktivieren Sie irgendeine dieser Tabellen, selektieren darin den Bereich A3:I7 und löschen den Inhalt dieses Bereichs mit **[Entf]**, wodurch er auch in allen anderen Tabellen der Gruppe gelöscht wird.

Abbildung 8.19:
Buchfrg.xls

	A	B	C	D	E	F	G	H	I
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12				Netto	MWSt				
13				- Steuerpflichtige Umsätze mit 16%	3327,16	532,34			
14				- Steuerpflichtige Umsätze mit 7%	1745,40	122,18			
15	Gesamt				5072,56	654,52			
16	Vorsteuerbeträge					462,95			
17	Umsatzsteuervorauszahlung					191,57			
18									

*Bereiche
verschieben/
kopieren*

Sie können nicht nur komplette Blätter von einer Tabelle oder Mappe in eine andere verschieben oder kopieren, sondern auch einzelne Bereiche.

Um einen selektierten Bereich in eine andere Mappe zu verschieben, ziehen Sie ihn einfach zum Fenster dieser Mappe und lassen ihn dort fallen. Halten Sie dabei die **[Strg]**-Taste gedrückt, so wird er nicht verschoben, sondern kopiert.

Um einen selektierten Bereich in ein anderes Blatt der gleichen Mappe zu verschieben oder zu kopieren, halten Sie die **[Alt]**-Taste gedrückt (zum Kopieren zusätzlich auch die **[Strg]**-Taste), während Sie den Bereich in der Registerkartenleiste zum gewünschten Blattnamen ziehen und dort fallen lassen.

8.5 Fenster-Handling

8.5.1 Mehrere Fenster für eine Mappe

Oft möchte man mehrere Tabellen einer Mappe *gleichzeitig* betrachten in der Buchführungsmappe möglicherweise die Tabellen »Summe« und eine der zwölf Monats-Tabellen.

Entweder blättern Sie ständig in der Mappe vor und zurück, aktivieren also jeweils die gewünschte Tabelle, oder aber Sie richten mit FENSTER[NEUES FENSTER ein zweites Fenster für diese Mappe ein (Abbildung 8.20).

Dass sich beide Fenster auf die gleiche Mappe beziehen, sehen Sie an den bis auf den Zusatz »:1« (erstes Fenster) beziehungsweise »:2« (zweites Fenster) identischen Namen. In beiden Fenstern können Sie nun unabhängig voneinander verschiedene Blätter der Mappe betrachten.

Buchfrg.xls:1

	A	B	C	D	E	F	G	H	I
1		Einnahmen			Ausgaben				
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			

Buchfrg.xls:2

	A	B	C	D	E	F
7					Netto	MWSt
8		- Steuerpflichtige Umsätze mit 16%			3327,16	532,34
9		- Steuerpflichtige Umsätze mit 7%			1745,40	122,18
10		Gesamt			5072,56	654,52
11		Vorsteuerbeträge				462,95
12		Umsatzsteuervorauszahlung				191,57

Abbildung 8.20:
Zwei Fenster für
eine Arbeitsmappe

Jedes Fenster steht stellvertretend für die gesamte Mappe. Voneinander unabhängige Editierungen in den beiden Fenstern sind daher nicht möglich: Eine Editierung, die in einem der beiden Fenster vorgenommen wird, erscheint – wenn darin das betreffende Blatt aktiviert wird – analog im zweiten Fenster.

Sie können beliebig viele Fenster auf einer Arbeitsmappe anlegen. Aktivieren Sie danach im Befehl FENSTER|ANORDNEN... die Option *Fenster der aktiven Arbeitsmappe*, so werden nur die Fenster der aktiven Mappe arrangiert und nicht die eventuell gleichzeitig geöffneten Fenster anderer Arbeitsmappen.

8.5.2 Aus- und Einblenden

Oft hat man vorübergehend eine ganze Menge Fenster im Speicher, arbeitet jedoch längere Zeit nur mit einem Bruchteil davon.

Beispielsweise arbeiten Sie zeitweise nur mit den Fenstern A und B, danach längere Zeit nur mit C und D, anschließend mit E und F und danach wieder mit den beiden ersten Fenstern A und B.

In derartigen Fällen können Sie mit dem Befehl FENSTER|AUSBLENDEN die Anzeige aller voraussichtlich für längere Zeit nicht benötigten Fenster unterbinden. Dieser Befehl blendet das momentan aktive Fenster aus.

Mit FENSTER|EINBLENDEN... können Sie verborgene Fenster jederzeit wieder aus der Versenkung holen, indem Sie im zugehörigen Listenfeld, das die Namen aller verborgenen Fenster enthält, jenes selektieren, das Sie wieder einblenden wollen.

8.5.3 Unterfenster



Scrollen Sie häufig in einer Tabelle, wollen aber ständig bestimmte Werte im Blick haben, vielleicht sogar Werte aus einer anderen als der momentan aktiven Tabelle, so fügen Sie sie ins Überwachungsfenster ein: Klicken Sie mit der rechten Maustaste auf die betreffende Zelle und wählen Sie im Kontextmenü ÜBERWACHUNG HINZUFÜGEN.



Das Überwachungsfenster wird eingeblendet und zeigt ständig die aktuellen Inhalte jener Zellen an (siehe Kapitel 38.3, »Fehler in Tabellen aufdecken«, Abschnitt »Überwachungsfenster«).

Geht es um bestimmte Bereiche statt um ein paar Zellen einer Tabelle und um dauerhafte Lösungen, sollten Sie besser Unterfenster verwenden, durch die ein Fenster horizontal und/oder vertikal geteilt wird. In jenen Teilen können Sie verschiedene Ausschnitte der gleichen Tabelle anzeigen und unabhängig voneinander in diesen Ausschnitten scrollen.



Am einfachsten richten Sie Unterfenster ein, indem Sie an den winzigen Kästchen am oberen Ende der vertikalen bzw. am rechten Ende der horizontalen Bildlaufleiste ziehen.

Alternativ dazu können Sie den Befehl FENSTER|TEILEN wählen und zuvor die Zelle selektieren, bei der die Teilung erfolgen soll.

Befindet sich diese Zelle mitten in der Tabelle, wird das Fenster links daneben und oberhalb von ihr geteilt, sodass vier Unterfenster entstehen (Abbildung 8.21).

Abbildung 8.21:
Aufteilung in vier
Unterfenster

	A	B	C	H	I
1		Einnahmen			
2		Bücher	Programme	GW-Güter	Abschreibung
3		230	3859,5	473	1728
4		1637,58			
5					
6					
7					
12					
13		- Steuerpflichtige Umsätze mit			
14		- Steuerpflichtige Umsätze mit			
15	Gesamt				
16	Vorsteuerbeträge				
17	Umsatzsteuervorauszahlung				

Befindet sich die Zelle in Spalte A, wird das Fenster nur horizontal geteilt: oberhalb der Zelle erscheint ein horizontaler Teilungsstrich. Befindet sich die Zelle in Zeile 1, wird es entsprechend nur vertikal geteilt.

Die vier Unterfenster sind nicht völlig unabhängig voneinander. Bei einem Bildlauf nach oben oder unten scrollt der Ausschnitt des aktuellen und des rechts bzw. links benachbarten Unterfensters in die entsprechende Richtung.

Bei einem Bildlauf nach rechts oder links scrollt der Ausschnitt des aktuellen und des darüber bzw. darunter liegenden Unterfensters.

Die beiden anderen Unterfenster bleiben jeweils unverändert. Dank dieser Eigenschaften können Sie die vier Unterfenster entsprechend der Abbildung so einrichten, dass in ihnen vier weit auseinander liegende Tabellenteile gleichzeitig sichtbar sind.

Bestehende Fensterteilungen manipulieren Sie durch Anklicken und Ziehen der Teilungslinien zur gewünschten Position.

Mit FENSTER|TEILUNG AUFHEBEN oder schneller per Doppelklick auf die entsprechende Teilungslinie heben Sie die Fensterteilung wieder auf. Wobei beim Doppelklick nur die tatsächlich angeklickte Teilungslinie verschwindet, also entweder die horizontale oder die vertikale und im Gegensatz zum entsprechenden Befehl nicht beide gleichzeitig!

8.5.4 Fenster fixieren

Soll der Inhalt eines Unterfensters dauernd sichtbar und darin kein Bildlauf mehr möglich sein, aktivieren Sie es und wählen FENSTER|FENSTER FIXIEREN oder klicken auf das zugehörige Symbol (Kategorie *Fenster u. Hilfe (?)* des *Anpassen*-Dialogfelds).



Sind noch keine Unterfenster eingerichtet, werden diese automatisch entsprechend den bei FENSTER|TEILEN erläuterten Regeln eingerichtet.

Wollen Sie beispielsweise nur ein Unterfenster einrichten, um darin in Zeile 1 und 2 enthaltene Überschriften zu fixieren, so klicken Sie A3 an und wählen FENSTER|FENSTER FIXIEREN (Abbildung 8.22).

Danach können Sie wie in der Abbildung gezeigt den Inhalt des unteren Fensters scrollen und sehen dennoch im oberen fixierten Fenster ständig die gewünschten Überschriften (auf Zeile 2 folgt in der Abbildung Zeile 11). Zusätzlich entfällt der hässliche dicke Trennstrich zwischen den Unterfenstern.

Die Fixierung können Sie mit FENSTER|FIXIERUNG AUFHEBEN oder durch erneutes Anklicken des Symbols wieder aufheben.

Abbildung 8.22:
Fixiertes Fenster
mit Spaltenüber-
schriften

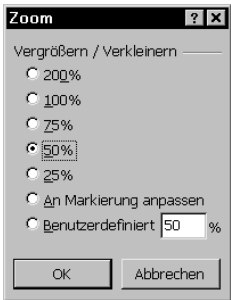
	A	B	C	D	E	F	G	H	I
1		Einnahmen			Ausgaben				
2		Bücher Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung	
11									
12				Netto	MWSt				
13		- Steuerpflichtige Umsätze mit 16%			3356,09	503,41			
14		- Steuerpflichtige Umsätze mit 7%			1745,40	122,18			
15		Gesamt			5101,49	625,59			
16		Vorsteuerbeträge				437,79			
17		Umsatzsteuervorauszahlung				187,80			
18									

8.5.5 Zoom-Funktion



ANSICHT|ZOOM... ermöglicht Ihnen ebenso wie das entsprechende Symbol der Standardleiste das Vergrößern oder Verkleinern von Tabellen (Abbildung 8.23).

Abbildung 8.23:
Die Zoom-Funktion



Unter *Benutzerdefiniert* können Sie einen beliebigen Vergrößerungsfaktor zwischen 10% und 400% eingeben oder aber einen der vorgegebenen Vergrößerungsfaktoren wählen.

Die Voreinstellung »100%« entspricht der Normaldarstellung. »50%« würde die Darstellungsgröße halbieren (Abbildung 8.24).

Abbildung 8.24:
Vergrößerungs-
faktor 50%

	A	B	C	D	E	F	G	H	I
1		Einnahmen			Ausgaben				
2		Bücher Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung	
3		230,00	3859,50	2,00	19,80	184,00	473,00	1728,00	
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8		Brutt	1867,58	3859,50	9,70	73,00	580,00	473,00	1728,00
9		MWSt	122,18	503,41	0,00	9,52	65,53	75,65	61,70
10		Netto	1745,40	3356,09	9,70	63,48	504,35	411,30	1502,61
11									
12				Netto	MWSt				
13		- Steuerpflichtige Umsätze			3356,09	503,41			
14		- Steuerpflichtige Umsätze			1745,40	122,18			
15		Gesamt			5101,49	625,59			
16		Vorsteuerbeträge				437,79			
17		Umsatzsteuervorauszahlung				187,80			
18									
19									

»200%« würde die Größe entsprechend verdoppeln.

Mit *An Markierung anpassen* berechnet Excel diesen Faktor selbstständig, und zwar so, dass ein vor Anwahl dieses Befehls selektierter Bereich gerade noch in voller Breite oder Höhe in das Tabellenfenster hineinpasst (Abbildung 8.25).

	B	C	D	E	F	G	H
1	Einnahmen			Ausgaben			
2	Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Gl
5			3,70		73,00		
6					68,90		
7							
8	1867,58	3859,50	9,70	73,00	502,40	580,00	473
9	122,18	503,41	0,00	9,52	65,53	75,65	61
10	1745,40	3356,09	9,70	63,48	436,87	504,35	411
11							
12				Netto	MWSt		
13	Steuerpflichtige Umsätze mit 16%			3356,09	503,41		
14	Steuerpflichtige Umsätze mit 7%			1745,40	122,18		

Abbildung 8.25:
An Markierung
anpassen

Das Vergrößern/Verkleinern betrifft nur die Bildschirmdarstellung, nicht den Ausdruck! Die eingestellten Schriftgrade und Zeilenhöhen bleiben trotz des gegenteiligen Eindrucks unverändert erhalten, wovon Sie sich in der Seitenansicht überzeugen können.



Die Kategorie *Ansicht* des *Anpassen*-Dialogfelds enthält zwei Lupen-Symbole, die ebenfalls zum Vergrößern bzw. Verkleinern gedacht sind:

Dieses Symbol vergrößert die Tabelle jeweils auf die doppelte Größe.



Dieses Symbol verkleinert die Tabelle jeweils auf die halbe Größe.



8.6 3D-Verknüpfungen

Häufig wollen Sie sich in einer Tabelle auf eine oder mehrere andere Tabellen der Mappe beziehen.

Ein Beispiel dafür ist die wenige Seiten zuvor erstellte kleine Buchführung: Die zwölf Tabellen »Jan« bis »Dez« enthalten Monatsübersichten.

Eine dreizehnte Tabelle »Summe« soll die Gesamtübersicht für das Jahr enthalten. In »Summe« müssen Sie sich somit auf Zellen beziehen, die in den Tabellen »Jan« bis »Dez« enthalten sind. Dazu verwenden Sie 3D-Verknüpfungen, die nach folgendem Schema gebildet werden:

Tabellenname!Zellreferenz

Enthält Zelle A1 einer Tabelle »Summe« beispielsweise die Formel

```
=Mai!B5,
```

so wird in dieser Zelle der Inhalt von Zelle B5 einer Tabelle »Mai« angezeigt, die sich in der gleiche Mappe befindet. Die Formel

```
=Mai!B5 + Jun!B6
```

würde dazu den Inhalt von Zelle B6 der Tabelle »Jun« addieren.

Ein Bezug auf einen Bereich besitzt die Form

```
Tabelle!name!Ecke_oben_links:Ecke_unten_rechts.
```

Da mehrere Zellen gleichzeitig referenziert werden, müssen Sie angeben, durch welche Operation die darin enthaltenen Werte verknüpft werden sollen, zum Beispiel, indem Sie diesen Bereich der Funktion *SUMME()* als Argument übergeben. Die Formel

```
=SUMME(Mai!A1:B2)
```

addiert die in den Zellen A1, A2, B1 und B2 von »Mai« enthaltenen Werte und zeigt beispielsweise das Ergebnis 10 an, wenn diese Zellen die Werte 1, 2, 3 und 4 enthalten. Entsprechend würde

```
=MITTELWERT(Mai!A1:B2)
```

in diesem Fall als Mittelwert 2,5 anzeigen.

Damit ist es kein Problem, die einige Seiten zuvor entwickelte Buchführungsmappe zu vervollständigen (Abbildung 8.26).

Abbildung 8.26:
Summen-Tabelle

B3 =SUM(Jan:Dez!B8)

	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher Programme		Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3	Brutto	1867,58							
4	MWSt								
5	Netto								
6									
7				Netto		MWSt			
8				- Steuerpflichtige Umsätze mit 16%					
9				- Steuerpflichtige Umsätze mit 7%					
10	Gesamt								
11	Vorsteuerbeträge								
12	Umsatzsteuervorauszahlung								
13									

Außer auf eine einzelne Tabelle können Sie sich mit 3D-Bezügen auf ganze Tabellenbereiche beziehen:

Tabellenname_A:Tabellenname_B!Zellreferenz

Ein gutes Beispiel dafür bietet die Zelle B3, die folgende Formel enthält:

=SUMME(Jan:Dez!B8)

Sie addiert die Inhalte der Zellen B8 der Tabellen »Jan« bis »Dez«.

Auch 3D-Bezüge können Sie per *Zeigen* eingeben, angewandt auf das Beispiel: Sie tippen »=SUMME(«. Danach aktivieren Sie »Jan« und klicken darin auf B8. Anschließend klicken Sie bei gedrückter -Taste auf den Tabellennamen »Dez«, um alle gewünschten Tabellen zu selektieren (»Jan« bis »Dez«).

Excel ergänzt die Formel zu »=SUMME(Jan:Dez!B8« und fügt nach dem Beenden der Eingabe mit  sogar die schließende Klammer »)« hinzu.

Diese Formel können Sie problemlos in die Bereiche C3:I3 und B4:I5 kopieren, da die Formel dabei angepasst wird und beispielsweise C3 anschließend die Formel

=SUMME(Jan:Dez!C8)

enthält. Kopieren Sie sie jedoch nicht mit EINFÜGEN, sondern mit INHALTE EINFÜGEN... und der Option »Formeln«, damit die rechten Zellrahmenlinien von Zelle C3 und I3 erhalten bleiben.

Geben Sie anschließend noch in E8 die Formel

=SUMME(Jan:Dez!E13)

ein und kopieren Sie sie nach E9:E10 und F8:F12, so ist die kleine Buchführung vollständig (Abbildung 8.27).

C4 =SUM(Jan:Dez!C9)

	A	B	C	D	E	F	G	H	I
1		Einnahmen			Ausgaben				
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
4	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
5	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
6									
7				Netto	MWSt				
8		- Steuerpflichtige Umsätze mit 16%		3327,16	532,34				
9		- Steuerpflichtige Umsätze mit 7%		1745,40	122,18				
10	Gesamt			5072,56	654,52				
11	Vorsteuerbeträge				462,95				
12	Umsatzsteuervorauszahlung				191,57				

Abbildung 8.27:
Buchfrg.xls,
»Summe«

Sie können sich nicht nur auf einzelne Zellen eines Tabellenbereichs beziehen, sondern auch auf Zellbereiche:

Tabellenname_A:Tabellenname_B!Ecke_oben_links:Ecke_unten_rechts

Entsprechend zeigt

=SUMME(Jan:Dez!A1:B2)

die Summe der Werte an, die die Zellen im Bereich A1:B1 der Tabellen »Jan« bis »Dez« enthalten.

Beispielsweise hätten Sie in Zelle F9 der Tabelle »Summe« der Buchführung die Formel

=SUMME(Jan:Dez!D9:I9)

verwenden können, um die gesamten Vorsteuerbeträge aller zwölf Monate zu addieren.

8.7 Externe Bezüge

Ab und zu werden Sie sich in einer Tabelle auf Zellen beziehen müssen, die sich in anderen Arbeitsmappen befinden. Wo sich diese Mappen befinden, ist egal; ob auf Ihrer eigenen Festplatte oder der eines anderen Rechners, auf den Sie über das Firmennetzwerk Zugriff haben, spielt keine Rolle.



Wie ich in Kapitel 26.3, »Verknüpfungen«, erläutere, dürfen sich diese Tabellen sogar irgendwo im Internet befinden. Selbst dann kann Excel auf den Inhalt dieser Zellen zugreifen, wobei natürlich per Modem eine Verbindung zum betreffenden FTP-Server hergestellt werden muss.

Angenommen, Sie verschieben die Tabelle »Summe« aus der Datei BUCHFRG.XLS, indem Sie den Tabellennamen »Summe« anklicken und die Tabelle an irgendeinen Punkt des Excel-Fensters ziehen, an dem sich keine Arbeitsmappe befindet.

Dann entfernt Excel die Tabelle aus BUCHFRG.XLS und legt eine neue Mappe an, in die sie eingefügt wird (Abbildung 8.28).

Dabei wird aus der zuvor in B3 enthaltenen Formel

=SUMME(Jan:Dez!B8)

die Formel

=SUMME([BUCHFRG.XLS]Jan:Dez!B8).

B3 =SUMME([BUCHFRG.XLS]Jan:Dez!B8)

Mappe1

	A	B	C	D	E	F	G	H	I
1		Einnahmen				Ausgaben			
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
4	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
5	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66

Buchfrg.xls

	A	B	C	D	E	F	G	H	I
1		Einnahmen				Ausgaben			
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6					68,90				
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66

Abbildung 8.28:
Externer Bezug auf
die Monats-
Tabellen

Excel macht aus dem Bezug auf die Zelle B8 der Tabellen »Jan« bis »Dez« automatisch einen externen Bezug, der nicht nur auf diese Zellen verweist, sondern zusätzlich auch auf die Mappe BUCHFRG.XLS, in der sie sich befinden.

Offenbar wird ein solcher externer Bezug mit dem von eckigen Klammern umschlossenen Namen der betreffenden Mappe eingeleitet, dem der Name der darin ausgewählten Tabelle bzw. des Tabellenbereichs folgt.

Allgemein besitzen externe Bezüge die Form

[Dateiname]Tabellename_A:Tabellename_B!Ecke_oben_links:Ecke_unten_rechts.

Schließen Sie die Arbeitsmappe mit den Monats-Tabellen, so ändert sich die in C3 enthaltene Formel erneut, beispielsweise so:

=SUMME('C:\EXCEL\DEMOS\BUCHFRG.XLS]Jan:Dez'!B8)

Ist nämlich die Mappe, auf die Bezug genommen wird, momentan nicht geöffnet, muss der Bezug die vollständige Pfadangabe enthalten, wobei Pfad und Tabellenname in Hochkommata eingeschlossen sind. Allgemein:

'PFAD[Dateiname]Tabellename_A:Tabellename_B!Ecke_oben_links:Ecke_unten_rechts

Beachten Sie bitte, dass es sich beim Namen der externen Arbeitsmappe exakt um jenen Namen handeln muss, der sich in der Titelleiste der betreffenden Mappe befindet!

**!!
STOP**

Legen Sie eine Mappe neu an, steht darin beispielsweise »Mappe5«. Ein externer Bezug auf Zelle B8 der Tabelle »Summe« dieser Mappe lautet demnach

=SUMME([Mappe5]Summe!B8).

Speichern Sie die Mappe, dann steht jedoch anschließend in der Titelleiste nicht mehr »Mappe5«, sondern »MAPPE5.XLS«, also der gleiche Name wie zuvor, allerdings in Großbuchstaben und mit dem Zusatz ».XLS«.

Gäben Sie einen externen Bezug auf diese Mappe *nach dem Speichern* ein, müssten Sie daher folgende Formel verwenden:

=SUMME([MAPPE5.XLS]Summe!B8)



Sie vermeiden dieses Problem, indem Sie es einfach Excel überlassen, den korrekten Bezug einzufügen, indem Sie externe Bezüge also nicht per Hand eingeben, sondern im Zeigemodus.

Öffnen Sie dazu die Mappe, auf die Sie sich beziehen wollen, tippen Sie die Formel bis zu jenem Punkt ein, an dem der Bezug folgt, beispielsweise »=A1+«, und zeigen Sie auf die bezogene Zelle der anderen Mappe, beispielsweise auf die Zelle B8 der Tabelle »Summe« in »Mappe5«.

Excel wird die Formel automatisch zu =A1+[Mappe5]Test!B8 ergänzen und Sie müssen nur noch  drücken.

Würde der Name der Mappe MAPPE5.XLS lauten, würde Excel die Formel stattdessen korrekt zu =SUMME([MAPPE5.XLS]Summe!B8) ergänzen.

Aktualisierung

Befindet sich die Tabelle »Summe« nicht in der gleichen Mappe wie die Monats-Tabellen, auf die sie sich bezieht, dann gibt es ein Problem: Sie können Excel starten, nur diese Monats-Tabellen öffnen und darin Werte eintragen, ohne dass die Tabelle »Summe« es mitbekommt, weshalb sie nun veraltete Summen enthält.

Um dieses Problem zu lösen, erhalten Sie beim Öffnen einer Mappe X, die externe Bezüge auf eine Mappe Y enthält, die momentan nicht geöffnet ist, folgende Meldung (siehe Abbildung 8.29).

Antworten Sie mit *Aktualisieren*, geht Excel nach dem Öffnen der Mappe alle darin enthaltenen externen Bezüge durch und sieht sich auf der Festplatte die in den betreffenden Zellen von Y enthaltenen Werte an, sodass X nun auf dem neuesten Stand ist, auch wenn Sie irgendwann zuvor den Inhalt von Y veränderten.

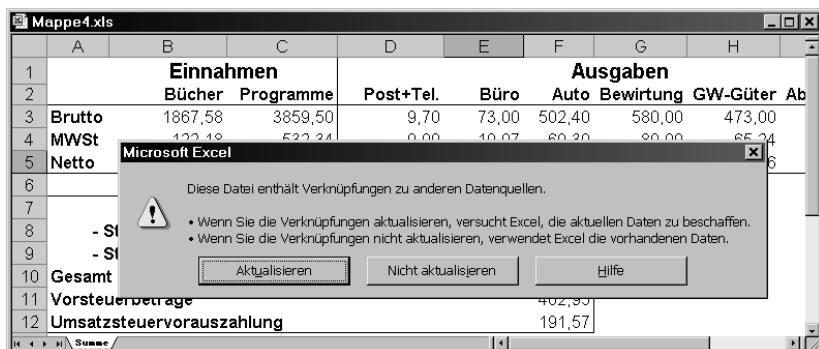


Abbildung 8.29:
Aktualisieren
externer Bezüge

Haben Sie jedoch sowieso vor, nicht nur X, sondern danach auch Y zu öffnen, können Sie ruhig mit *Nicht aktualisieren* antworten, da Excel nach dem Öffnen von Y sowieso alle momentan geöffneten Mappen aktualisiert, die sich auf diese Mappe beziehen, also auch X.

Diese Aktualisierung geht sehr weit. Angenommen, momentan ist nur Y geöffnet und Sie verschieben darin jene Zelle, auf die sich der in X enthaltene Bezug bezieht:

In diesem Fall schließen Sie Y anschließend und öffnen Sie X, so fragt Excel wie gezeigt, ob die Bezüge auf nicht geöffnete Dateien aktualisiert werden sollen.

Nehmen Sie das Angebot an, wird Y geöffnet und wenn Sie sich den externen Bezug ansehen, werden Sie feststellen, dass er nicht mehr die alte Zellreferenz enthält, sondern eine Referenz auf jene Position, an die Sie die betreffende Zelle in Y verschoben haben – und das, obwohl X währenddessen gar nicht geöffnet war und davon scheinbar überhaupt nichts mitbekam!

Sogar beim Umbenennen einer Datei funktioniert diese Aktualisierung. Angenommen, zwei Mappen sind geöffnet: X.XLS enthält einen externen Bezug `=Y.XLS/Test!B8` auf eine Zelle in Mappe Y.XLS.

*Bezugstabellen
umbenennen/
verlegen*

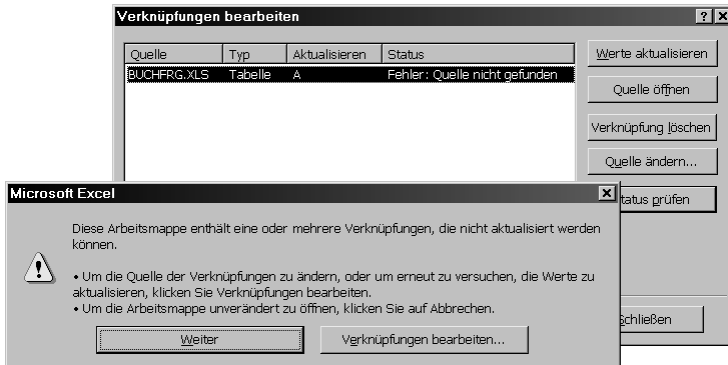
Sie speichern Y.XLS nun mit **SPEICHERN UNTER...** unter dem Namen Z.XLS und löschen anschließend die alte Mappe Y.XLS. Excel ändert daraufhin den externen Bezug in `=Z.XLS/Test!B8` ab und berücksichtigt damit, dass die Bezugstabelle nun nicht mehr Y.XLS heißt, sondern Z.XLS.

Allerdings klappt das nur, wenn die Mappe X.XLS, die den externen Bezug enthält, geöffnet ist, während Sie Y.XLS umbenennen oder in ein anderes Verzeichnis verlegen!

Ist X.XLS nicht geöffnet oder nehmen Sie die Umbenennung/Verlegung gar außerhalb von Excel vor, beispielsweise mit dem Windows-Datei-Manager, so ist Excel nicht in der Lage, im externen Bezug in Mappe X.XLS den Dateinamen Y.XLS durch Z.XLS zu ersetzen.

Öffnen Sie X.XLS irgendwann später, scheitert Excel daher beim Versuch, die darin enthaltenen externen Bezüge zu aktualisieren und bietet Ihnen an, die Mappe nicht aktualisiert zu öffnen oder aber die darin enthaltenen Verknüpfungen zu bearbeiten (Abbildung 8.30).

Abbildung 8.30:
Datei nicht
gefunden



Verknüpfungen bearbeiten... öffnet das im Hintergrund sichtbare Dialogfeld. Klicken Sie darin auf *Quelle ändern...*, erscheint das *Dateiauswahl*-Dialogfeld. Darin selektieren Sie die Mappe, auf die sich die externen Bezüge beziehen und geben Excel auf diese Weise ihren neuen Namen bzw. das Verzeichnis an, in das sie verschoben wurde.

8.8 Arbeitsmappen als Vorlagen verwenden und automatisch öffnen

8.8.1 Startordner

XLSTART

In dem Ordner, in dem Sie Excel bzw. das Office-Paket installierten, befindet sich unter anderem der Startordner XLSTART. Alle Dateien, die Sie darin speichern, werden beim Start von Excel *automatisch geöffnet*!

Im Eingabefeld *Beim Start alle Dateien in diesem Ordner laden* des Registers *Allgemein* von EXTRAS|OPTIONEN... können Sie einen zusätzlichen Ordner angeben, der die gleiche Aufgabe besitzen soll.

Anschließend werden beim Start von Excel alle Dateien automatisch geöffnet, die sich entweder in XLSTART befinden oder aber in jenem zusätzlich angegebenen Startordner.

Speichern Sie eine Datei im Startordner, die ein Makro namens AUTO_ÖFFNEN enthält (siehe Kapitel 31.2, »Auto-Makros«), so wird dem Anwender jegliche Arbeit abgenommen: Nach dem Start von Excel wird die Mappe geladen und automatisch dieses darin enthaltene Makro ausgeführt, das beispielsweise ein selbstdefiniertes Dialogfeld öffnet oder ein VBA-Programm startet und den Anwender beispielsweise nach einer einzugebenden Adresse fragt oder nach den Daten für eine auszuführende Kalkulation fragt, deren Ergebnis ihm anschließend präsentiert wird.

Speichern Sie Dateien im Startordner im Dateityp als Mustervorlage (Endung XLT), werden sie zwar nicht automatisch geöffnet, dafür jedoch wie im folgenden Kapitel beschrieben als Vorlagen verwendet.



8.8.2 Mustervorlagen

Um eine neue Arbeitsmappe zu erzeugen, klicken Sie entweder auf das abgebildete Symbol oder wählen DATEI|NEU... und klicken danach im Aufgabenbereich auf *Leere Arbeitsmappe*. In beiden Fällen erzeugt Excel eine neue Arbeitsmappe, die genauso aufgebaut ist wie seine Standardvorlage.



Möglicherweise passt Ihnen jedoch die dadurch vorgegebene Spaltenbreite nicht oder Sie wollen prinzipiell keine Gitternetzlinien sehen. Dann erzeugen Sie einfach Ihre eigene Standardvorlage, damit neue Mappen automatisch die von Ihnen gewünschte Struktur erhalten.

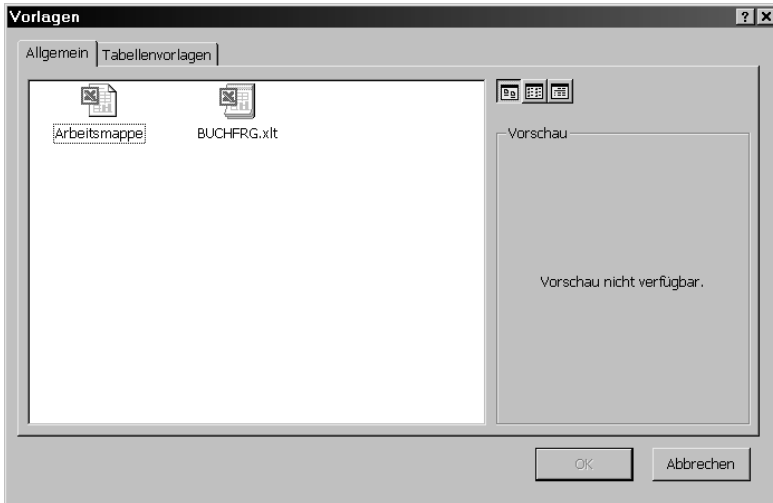
Sie können sogar zusätzliche Vorlagen erzeugen, was immer dann sinnvoll ist, wenn Sie öfters absolut identisch aufgebaute Arbeitsmappen erzeugen müssen.

Ein Beispiel: Besitzen Sie eine Arbeitsmappe für die Buchführung eines Jahres namens BUCHFRG.XLS, so müssen Sie jedes Jahr eine gleichartig aufgebaute Mappe erzeugen. Speichern Sie die Mappe als Vorlage, so können Sie diesen Vorgang vereinfachen und per Mausklick jedes Jahr eine neue Mappe anlegen, die auf dieser Vorlage basiert und exakt den gleichen Aufbau besitzt.

Dazu wählen Sie DATEI|SPEICHERN UNTER... und unter *Dateityp* die Option *Mustervorlage*. Dadurch wird automatisch der Ordner VORLAGEN geöffnet und die Datei erhält die Erweiterung .XLT und heißt somit nach dem Speichern in diesem Vorlagenordner BUCHFRG.XLT.

Auf diese Weise im Vorlagenordner als Mustervorlagen gespeicherte Mappen werden im Register *Allgemein* des *Vorlagen*-Dialogfelds aufgelistet, das erscheint, wenn Sie im Aufgabenbereich unter *Mit Vorlage beginnen* auf *Allgemeine Vorlagen...* klicken. Außer dem Eintrag »Arbeitsmappe« (Excels Standardvorlage) enthält dieses Register nun also den Eintrag »Buch-frg.xlt«, die von Ihnen gespeicherte Mustervorlage (Abbildung 8.31).

Abbildung 8.31:
Mustervorlage
öffnen



Wenn Sie darauf klicken, legt Excel eine neue Mappe namens »Buchfrg1« an, die genauso aufgebaut ist wie Ihre Mustervorlage, also eine Mappe, die nicht nur alle Werte enthält, die möglicherweise im Original BUCHFRG.XLT enthalten sind, sondern auch die gleichen Formatierungen, Zeilenhöhen, Spaltenbreiten etc. besitzt und die somit in jeder Beziehung mit der Vorlage identisch ist.

INFO

Zusätzlich listet Excel im Abschnitt *Mit Vorlage beginnen* des Aufgabenbereichs die zuletzt verwendeten Mustervorlagen auf, damit diese noch schneller zugänglich sind. Legten Sie eine auf der Mustervorlage »Buchfrg.xlt« basierende Mappe an, so befindet sich dort nach dem nächsten Start von Excel dieser Eintrag. Um erneut eine darauf basierende Mappe anzulegen, müssen Sie daher nicht mehr wie zuvor das Vorlagen-Dialogfeld öffnen, sondern nur noch unter *Mit Vorlage beginnen* auf »Buchfrg.xlt« klicken.

INFO

Sie können die Vorlagendatei BUCHFRG.XLT jederzeit ganz normal öffnen, um sie zu verändern und wieder zu speichern. Erzeugen Sie anschließend eine neue Mappe, die auf der Vorlage basiert, so wird sie den gleichen Inhalt und die gleiche Struktur wie die inzwischen veränderte Vorlage besitzen.

INFO

Klicken Sie im Aufgabenbereich unter *Neu aus vorhandener Arbeitsmappe* auf *Arbeitsmappe wählen...*, dann können Sie im folgenden Dateiauswahl-Dialogfeld eine beliebige Excel-Datei als zu verwendende Vorlage auswählen (für Vorlagen, die Sie immer wieder verwenden, ist die zuvor beschriebene Vorgehensweise jedoch empfehlenswerter).

8.8.3 Automatische Mustervorlagen

Statt im Vorlagenordner können Sie Mustervorlagen alternativ auch im Ordner XLSTART speichern.

Noch einen Schritt weiter geht die Möglichkeit, eine Mustervorlage in XLSTART als »automatische« Mustervorlage zu speichern. Wie erläutert ist der Eintrag »Arbeitsmappe« im Register *Allgemein* des *Vorlagen*-Dialogfelds die Standardvorlage von Excel beim Erzeugen neuer Arbeitsmappen.

Speichern Sie eine Arbeitsmappe im Verzeichnis XLSTART unter dem Namen MAPPE.XLT, so verwendet Excel ab sofort diese Mappe als neue Standardvorlage.

Wenn Sie Excel am nächsten Tag starten, wird die leere Arbeitsmappe, die automatisch immer angelegt wird, daher exakt Ihrer Vorlage MAPPE.XLT gleichen. Auch jede weitere mit dem abgebildeten Symbol oder mit *Leere Arbeitsmappe* erzeugte Mappe wird eine exakte Kopie von MAPPE.XLT sein.



Diese Technik sollten Sie vorwiegend für individuelle Formatierungs-Standards einsetzen: Sie verwenden eine leere Mappe, ändern darin die Formatvorlage »Standard«, weisen ihr beispielsweise den Schriftgrad 8 zu und speichern die Tabelle im Verzeichnis XLSTART als Mustervorlage MAPPE.XLT.

Analog dazu definiert eine im Verzeichnis XLSTART gespeicherte Datei TABELLE.XLT die Tabellen-Autovorlage. Speichern Sie eine Arbeitsmappe, die nur eine Tabelle enthält, unter diesem Namen, so wird diese Tabelle als Vorlage für jede neue Tabelle verwendet, die Sie mit EINFÜGEN|TABELLENBLATT in eine Arbeitsmappe einfügen. Enthält TABELLE.XLT mehrere Tabellen, so werden all diese Tabellen eingefügt.



8.8.4 Mitgelieferte Vorlagen

Das Register *Tabellenvorlagen* des *Vorlagen*-Dialogfelds enthält mehrere Vorlagen (Abbildung 8.32).

Diese Vorlagen sind fertige, teilweise äußerst komplexe Excel-Tabellen, die für das Ausfüllen von Bestellungen, das Anfertigen von Kostenvoranschlägen und andere »täglich« vorkommende Probleme gedacht sind (Abbildung 8.33).

Schauen Sie sich die verschiedenen Anwendungen einfach der Reihe nach an und prüfen Sie, welche für Ihre Zwecke brauchbar sind.

Abbildung 8.32:
Vorbereitete
Vorlagen

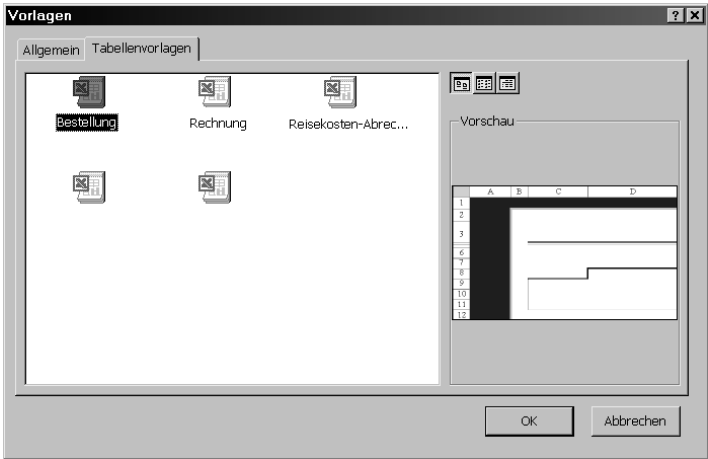


Abbildung 8.33:
Bestellformular

Anzahl	Einheit	Beschreibung	Preis/Einheit	TOTAL



Die beiden letzten Vorlagenbefehle im Aufgabenbereich ermöglichen Ihnen zusätzlich, Vorlagen zu verwenden, die auf Websites gespeichert sind, die Sie zuvor den Webordnern hinzugefügt, oder aber auf der Microsoft-Website, die weitere vorbereitete Vorlagen bereitstellt.

9 Namen

Excel ermöglicht Ihnen, bestimmten Werten, aber auch Zellen und ganzen Bereichen beliebige Namen zu geben. Anschließend können Sie in Formeln statt in unhandlichen Bezügen wie C35 die zugehörigen Namen verwenden, beispielsweise »MWSt-Prozentsatz«.

Ich zeige Ihnen nun, wie Sie derartige Namen definieren und anwenden. Anwenden bedeutet nicht nur, dass Sie diese Namen in Ihren Formeln manuell eingeben können: Excel kann die in Ihren Formeln enthaltenen Bezüge bei Bedarf selbstständig durch die entsprechenden Namen ersetzen, was die Anwendung dieser Technik bei bereits vorhandenen komplexen Tabellen ungemein erleichtert.

Excel kann Namen übrigens auch selbstständig vergeben – vorausgesetzt, neben den betreffenden Zellen befinden sich passende Texte, die Excel zur Benennung der Zellen verwenden kann!

9.1 Definieren und Editieren

Angenommen, Ihre Tabelle enthält Formeln, die irgendwelche Umsatzsteuerbeträge berechnen, ausgehend vom aktuellen Umsatzsteuersatz 16%, dann können Sie diesen Satz in eine Zelle des Arbeitsblatts eintragen und sich in den Formeln auf diese Zelle beziehen.

Namen festlegen

Mit EINFÜGEN|NAMEN|DEFINIEREN... geht es aber auch ohne den Umweg über eine Zelle (Abbildung 9.1).

Das Listenfeld enthält alle für das aktive Arbeitsblatt definierten »Namen« und ist momentan leer.

Konstanten

Geben Sie im Eingabefeld »MWSt« ein und unter *Bezieht sich auf* entsprechend der Abbildung die Formel »=16%« ein (das Gleichheitszeichen ist nicht unbedingt notwendig und kann entfallen). Wählen Sie *Hinzufügen* (oder OK, um das Dialogfeld zusätzlich wieder zu schließen).

Für Excel ist der Name »MWSt« nun gleichbedeutend mit 16% und die Formel =250*MWSt wird wie =250*16% berechnet (Abbildung 9.2).

Abbildung 9.1:
Konstanten
benennen

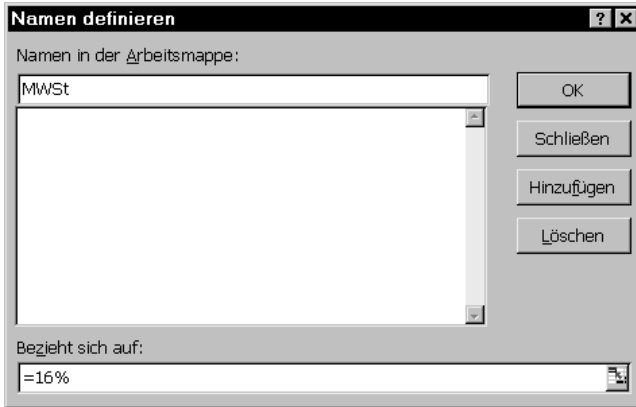
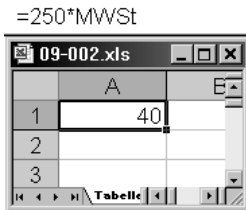


Abbildung 9.2:
Namen anwenden



Löschen und Ändern

Nach Selektion eines Eintrags der Namensliste können Sie den betreffenden Namen mit der Schaltfläche *Löschen* entfernen.

Sie können die Definition aber auch ändern, indem Sie in der Liste einen Namen wie »MWSt« selektieren und unter *Bezieht sich auf* zum Beispiel den aktuellen Eintrag *=16%* durch *=20%* ersetzen, wenn sich der Umsatzsteuersatz entsprechend erhöht.

Daraus resultiert, dass alle Ihre Formeln, die den Namen »MWSt« verwenden, nun automatisch mit dem aktualisierten Satz von 20% rechnen.

Sie ersparen sich dadurch, jede Formel einzeln aktualisieren zu müssen, wie das bei Formeln der Art *=250*16%*, in die die Konstanten direkt eingetragen werden, der Fall wäre.



Nach dem Löschen eines Namens kann es passieren, dass in einer oder mehreren Formelzellen #NAME? angezeigt wird. Diese Fehlermeldung weist Sie darauf hin, dass in der betreffenden Formel ein nicht (mehr) existierender Name verwendet wird.

Bei der Anwendung von Namen können Sie sich das Eintippen ersparen. Sie geben eine Formel nur bis zu jenem Punkt ein, an dem ein Name eingesetzt werden soll und wählen EINFÜGEN|NAMEN|EINFÜGEN... (ABBILDUNG 9.3).

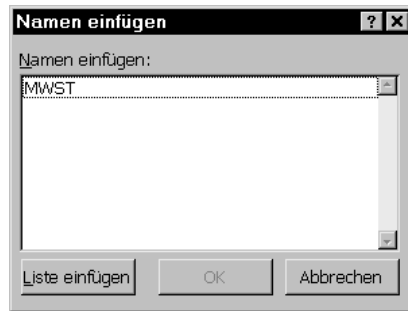


Abbildung 9.3:
Name einfügen

Selektieren Sie im Listenfeld einen Namen und schließen Sie das Dialogfeld, so wird der Name an der aktuellen Cursorposition in Ihre Formel eingefügt.

Sie können beliebig komplexe Formeln benennen, die sogar selbst wiederum zuvor definierte Namen enthalten dürfen.

Formeln

Nehmen wir an, Sie definierten einen Namen »Zins«, dem =8% zugewiesen ist. Zusätzlich benötigen Sie häufig die Hälfte des normalerweise verwendeten Zinssatzes, zum Beispiel weil von den Zinsen nach der Versteuerung nur die Hälfte übrig bleibt.

Dann können Sie einen weiteren Namen »Zinshalbe« festlegen, unter *Bezieht sich auf* =Zins/2 eingeben, und diese zweite Definition mit *Hinzufügen* der Namensliste hinzufügen (Abbildung 9.4).

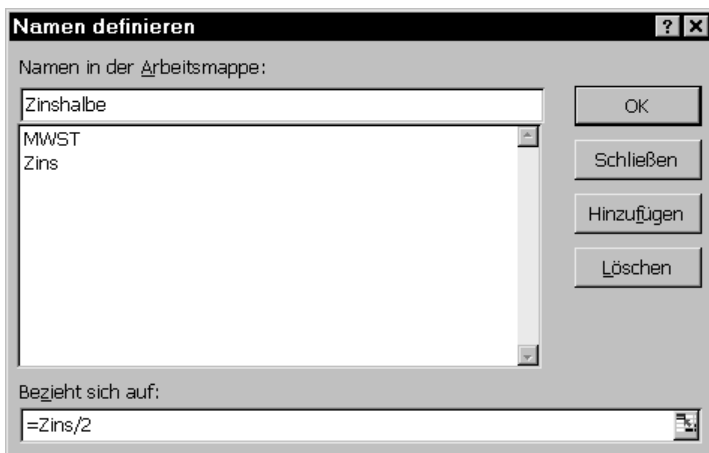


Abbildung 9.4:
Formeln benennen

»Zinshalbe« entspricht der Formel =Zins/2. Daher würde nun beispielsweise der Ausdruck =10000*Zinshalbe berechnet, indem Excel zuerst den Wert dieser Formel ermittelt, der wiederum vom Wert des Namens »Zins« abhängt, und ihn anschließend durch 2 dividiert.

Für die Namensgebung gelten folgende Regeln:

- ➔ Ein Name darf aus bis zu 255 Zeichen bestehen.
- ➔ Zulässige Zeichen sind Buchstaben, Ziffern, Punkte und der Unterstrich »_«. Das erste Zeichen muss ein Buchstabe sein.
- ➔ Da weder Leerzeichen noch Bindestriche zulässig sind, sollte zum Trennen mehrerer Wörter der Unterstrich oder ein Punkt verwendet werden (»Zins_halbe« oder »Zins.halbe«).
- ➔ Zwischen Groß- und Kleinschreibung wird nicht unterschieden. Für Excel ist »Test«, »test« und »TEST« ein und derselbe Name.
- ➔ Ein Name darf nicht mit einer Zellreferenz verwechselt werden können. Namen wie A1 oder Z(2)S(3) sind daher nicht zulässig.
- ➔ Die zugehörige Referenz sollte mit einem Gleichheitszeichen »=« beginnen.

**Zellen und
Bereiche**

Je größer und komplexer ein Arbeitsblatt wird, um so größer wird die Rolle von Zellnamen und Bereichsnamen.

Normalerweise sprechen Sie eine Zelle über ihre absoluten oder relativen Koordinaten an. Stattdessen können Sie jedoch jeder Zelle und jedem Bereich einen eigenen Namen wie »Einnahmen« oder »Ausgaben« geben und diese Namen anschließend in Formeln der Art

=Einnahmen-Ausgaben

verwenden. Diese Formel ist zweifellos erheblich verständlicher als etwa

=A5-D8.

Als Beispiel dafür verwende ich eine Monats-Tabelle aus der in den vorhergehenden Kapiteln vorgestellten Buchführung. Darin enthält beispielsweise die Zelle F16 die Formel

=SUMME(D9:I9).

Ebenso wie einer einzelnen Zelle können Sie auch Bereichen Namen geben und den Bereich D9:I9 beispielsweise »MWSt_Ausgaben« nennen.

**Namen und die
Bearbeitungs-
leiste**

Diese Definition können Sie im gewohnten Dialogfeld vornehmen und den Zellbezug eintippen oder darauf zeigen (und dazu gegebenenfalls das Dialogfeld mit dem bunten Symbol verkleinern).

Einfacher: Selektieren Sie die ausgewählte Zelle bzw. den Bereich und tippen Sie *im linken Teil der Bearbeitungsleiste*, in dem normalerweise der aktuelle Zellbezug angezeigt wird, den gewünschten Namen ein. Angewandt auf das Beispiel selektieren Sie D9:I9 und geben »MWSt_Ausgaben« ein (siehe Bild 9.5, ganz oben links).

MWSt_Ausgaben 0

09-005.xls

	A	B	C	D	E	F	G	H	I
1			Einnahmen				Ausgaben		
2			Bücher Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12				Netto	MWSt				
13				- Steuerpflichtige Umsätze mit 16%	3327,16	532,34			
14				- Steuerpflichtige Umsätze mit 7%	1745,40	122,18			
15	Gesamt				5072,56	654,52			
16	Vorsteuerbeträge					462,95			
17	Umsatzsteuervorauszahlung					191,57			

Abbildung 9.5:
Bereich benennen

Nach dem Beenden der Eingabe mit  können Sie die in F16 enthaltene Formel durch

=SUMME(MWSt_Ausgaben)

ersetzen und alle Berechnungen sind weiterhin korrekt (Abbildung 9.6).

F16 =SUMME(MWSt_Ausgaben)

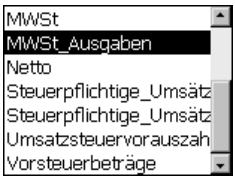
09-005.xls

	A	B	C	D	E	F	G	H	I
1			Einnahmen				Ausgaben		
2			Bücher Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12				Netto	MWSt				
13				- Steuerpflichtige Umsätze mit 16%	3327,16	532,34			
14				- Steuerpflichtige Umsätze mit 7%	1745,40	122,18			
15	Gesamt				5072,56	654,52			
16	Vorsteuerbeträge					462,95			
17	Umsatzsteuervorauszahlung					191,57			

Abbildung 9.6:
Name anwenden

Sie können auf diese Weise beliebig viele Bereiche benennen. Klicken Sie in der Bearbeitungsleiste den DropDown-Pfeil an, so wird das zugehörige Namenslistenfeld geöffnet (Abbildung 9.7).

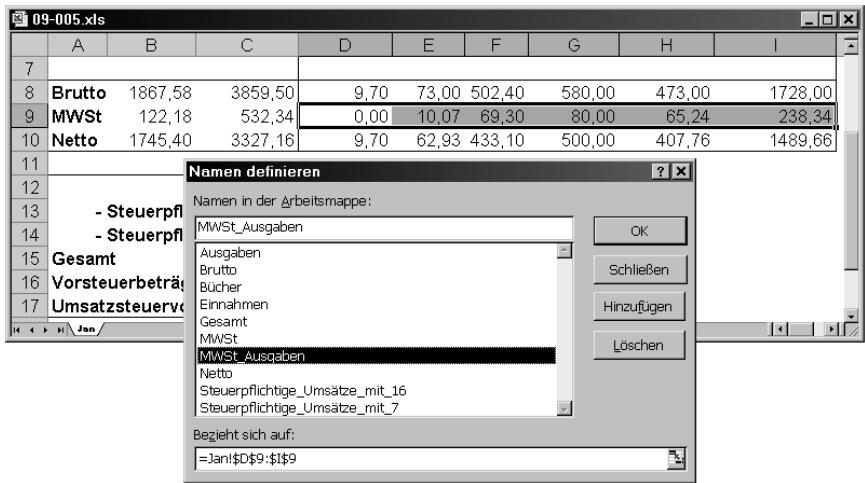
Abbildung 9.7:
Namenslistenfeld



Selektieren Sie darin einen Namen, markiert Excel sofort den zugehörigen Bereich in der Tabelle.

Wenn Sie darauf bestehen, können Sie Bereichsnamen natürlich auch wie zuvor Konstanten mit EINFÜGEN|NAMEN|DEFINIEREN... definieren (Abbildung 9.8).

Abbildung 9.8:
Bereich benennen



Unter *Bezieht sich auf* ist der zuvor selektierte Bereich bereits in Form absoluter Koordinaten vorgegeben. Um ihm einen Namen zuzuordnen, geben Sie im Eingabefeld »MWSt_Ausgaben« ein und aktivieren *Hinzufügen*. Anschließend ist dieser Name für Excel mit dem zugehörigen Bereich identisch.

Sie können auch umgekehrt vorgehen: ohne vorhergehende Selektion das Dialogfeld öffnen und einen Namen eingeben, dann *Bezieht sich auf* aktivieren und erst jetzt die gewünschte Zellreferenz eingeben bzw. mit der Maus auf die Zelle oder den Bereich zeigen, um Excel zu veranlassen, die betreffende Referenz einzufügen.

Um eine 3D-Referenz zu benennen, tippen Sie ebenfalls zunächst den gewünschten Namen ein und aktivieren danach *Bezieht sich auf*. Nun selektieren Sie die betreffenden Tabellen, beispielsweise »Jan« bis »Dez«.

**3D-Referenzen
benennen**

Anschließend selektieren Sie in einer dieser Tabellen die betroffene Zelle (beziehungsweise einen Bereich), beispielsweise C5. Excel fügt daraufhin ein Gleichheitszeichen und die zugehörige Referenz ein: `=Jan:Dez!$C$5`.

Normalerweise ist ein Name wie »MWSt« für die gesamte Mappe gültig, in der er definiert wird, und Sie können ihn in jeder Tabelle dieser Mappe anwenden.

Lokale Namen

Soll ein Name jedoch ausnahmsweise nur für die momentan aktive Tabelle der Mappe Gültigkeit besitzen, so geben Sie im Namensfeld den Namen der Mappe ein, gefolgt von einem Ausrufezeichen und dem eigentlichen Namen.

Ist beispielsweise momentan die Tabelle »Summe« aktiv und definieren Sie für `=16%` den Namen »Summe!MWSt«, dann können Sie den Namen »MWSt« in dieser Tabelle anschließend zwar wie gewohnt verwenden (`=213*MWSt` oder ähnlich), aber in keiner anderen Tabelle der Mappe.

Dafür können Sie dort jedoch identische lokale Namen definieren: beispielsweise in der Tabelle »Jan« mit »Jan!MWSt« und der Referenz `=20%` einen nur für diese Tabelle gültigen Namen »MWSt«, der darin mit 20% statt mit 16% gleichbedeutend ist.

Um sich eine Übersicht über alle momentan in der aktiven Mappe definierten Namen und die zugehörigen Referenzen zu verschaffen, wählen Sie **EINFÜGEN|NAMEN|EINFÜGEN...** und aktivieren im zugehörigen Dialogfeld *Liste einfügen*.

	G	H
29	Ausgaben	=Jan!\$D\$9:\$I\$9
30	Brutto	=Jan!\$D\$9:\$I\$9
31	Bücher	=Jan!\$D\$9:\$I\$9
32	Einnahmen	=Jan!\$D\$9:\$I\$9
33	Gesamt	=Jan!\$D\$9:\$I\$9
34	MWST	=Jan!\$D\$9:\$I\$9
35	MWST_Ausgaben	=Jan!\$D\$9:\$I\$9
36	Netto	=Jan!\$D\$9:\$I\$9
37	Steuerpflichtige_Umsätze_mit_16	=Jan!\$D\$9:\$I\$9
38	Steuerpflichtige_Umsätze_mit_7	=Jan!\$D\$9:\$I\$9
39	Umsatzsteuervorauszahlung	=Jan!\$G\$29
40	Vorsteuerbeträge	=Jan!\$G\$29
41		

Abbildung 9.9:
Namensliste

Excel fügt eine Liste aller definierten Namen als zweispaltigen Bereich in die aktive Tabelle ein, wobei die momentan aktive Zelle die linke obere Bereichsecke bildet.

Die linke Spalte dieses Bereichs enthält die definierten Namen, die rechte die zugehörigen Referenzen (Abbildung 9.9).

9.2 Namen erstellen

Die Verwendung von Namen vereinfacht den Überblick über eine Tabelle erheblich. Allerdings ist es mit hohem Aufwand verbunden, viele separate Bereiche einer Tabelle zu benennen und diese Namen anschließend anzuwenden, das heißt Bezüge in Formeln durch die nun definierten Namen für diese Bezüge zu ersetzen.

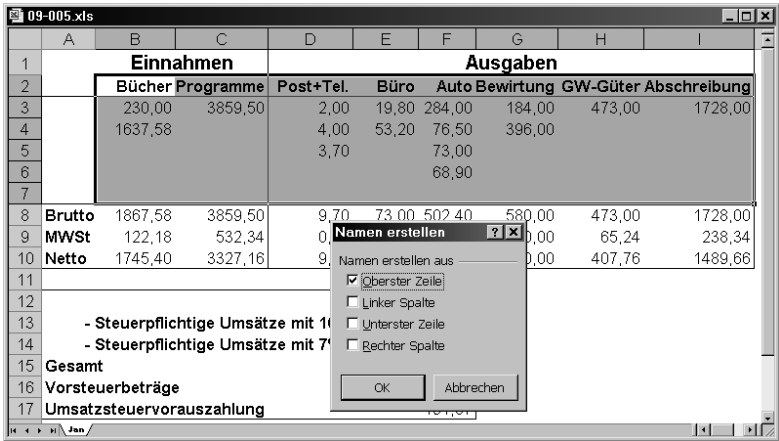
Excel erleichtert Ihnen diese Strukturierung mit zwei speziellen Befehlen. Der Befehl EINFÜGEN|NAMEN|ERSTELLEN... vergibt vollautomatisch passende Namen für eine beliebig große Anzahl an Bereichen.

Voraussetzung ist, dass an jeden dieser Bereiche passende Texte angrenzen, die Excel als Bereichsnamen verwenden kann. Wo diese Texte angrenzen, rechts, links, oben oder unten, ist gleichgültig.

Eine weitere Voraussetzung ist, dass alle zu verwendenden Texte den Regeln für gültige Namen entsprechen: Keine der Überschriften darf Leerzeichen oder andere in Namen unzulässige Zeichen enthalten!

Zur Praxis: Sie selektieren den zu benennenden Bereich *inklusive der daran angrenzenden Texte*, die als Namen verwendet werden sollen, und wählen EINFÜGEN|NAMEN|ERSTELLEN... (Abbildung 9.10).

Abbildung 9.10:
Übernahme von
Bereichsnamen



Sie müssen angeben, wo sich die zu übernehmenden Bereichsnamen befinden: ob in der obersten Zeile der selektierten Bereiche, in der linken Spalte, der untersten Zeile oder der rechten Spalte.

Da sich die Texte im Beispiel in der obersten Zeile befinden, gibt Excel entsprechend der Abbildung die Option *Oberste Zeile* vor, was Sie mit *OK* bestätigen.

Schauen Sie sich anschließend alle momentan definierten Namen im Listenfeld von EINFÜGEN|NAMEN|DEFINIEREN... an. Neu hinzugekommen sind die Namen »Bücher«, »Programme«, »Post_Tel« und so weiter. Wie »Post_Tel« zeigt, ersetzt Excel dabei nicht zulässige Zeichen wie »+« durch einen Unterstrich.

Den automatisch vergebenen Namen sind tatsächlich die gewünschten Bereiche zugeordnet: Selektieren Sie in der Liste den Namen »Post_Tel« (Abbildung 9.11), erscheint unter *Bezieht sich auf:* die Referenz

=Jan!\$D\$3:\$D\$7

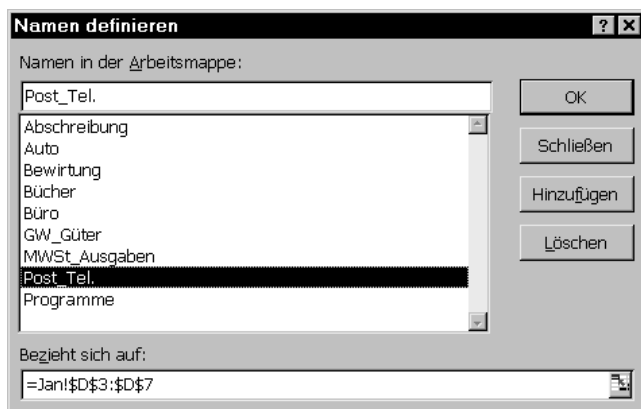


Abbildung 9.11:
Vergebene
Bereichsnamen

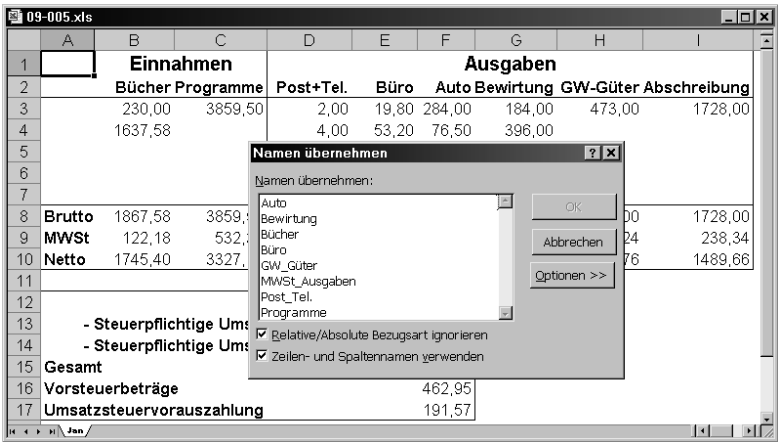
9.3 Namen übernehmen

Der Befehl EINFÜGEN|NAMEN|ÜBERNEHMEN... erspart es uns, nach der Definition von Bereichsnamen in allen Formeln die Bereichsangaben per Hand durch die zugehörigen Bereichsnamen zu ersetzen und nimmt diese Ersetzung vollautomatisch vor.

Ist vor Anwahl dieses Befehls ein Bereich selektiert, werden nur in den darin enthaltenen Formeln Bezüge durch zugehörige Namen ersetzt. Ist kein Bereich selektiert (nur eine einzelne Zelle ist aktiv), wird diese vollautomatische Ersetzung in der gesamten Tabelle durchgeführt.

Angewandt auf das vorhergehende Beispiel (Abbildung 9.12) sieht das so aus.

Abbildung 9.12:
Namen anwenden



Selektieren Sie die zu übernehmenden Namen im Listefeld durch Anklicken (erneutes Anklicken hebt die Selektion wieder auf) und klicken Sie auf OK.

Schauen Sie sich anschließend die in der Tabelle enthaltenen Formeln der Reihe nach an: Überall, wo zu einem Bezug ein Name existierte, ersetzte Excel den Bezug durch den betreffenden Namen.

Beispielsweise enthält B8 nun statt

=SUMME(B3:B7)

die wesentlich aussagekräftigere Formel

=SUMME(Bücher).

Der Name »Bücher« ersetzt somit den zugehörigen Bezug B3:B7. Insgesamt wurden folgende Bezüge durch Namen ersetzt:

Tabelle 9.1:
Angewendete
Namen

Alte Formel	Neue Formel
=SUMME(B3:B7)	=SUMME(Bücher)
=SUMME(C3:C7)	=SUMME(Programme)
=SUMME(D3:D7)	=SUMME(Post_Tel)
=SUMME(E3:E7)	=SUMME(Büro)
=SUMME(F3:F7)	=SUMME(Auto)

Alte Formel	Neue Formel
=SUMME(G3:G7)	=SUMME(Bewirtung)
=SUMME(H3:H7)	=SUMME(GW_Güter)
=SUMME(I3:I7)	=SUMME(Abschreibung)

Tabelle 9.1:
Angewendete
Namen
(Forts.)

Hätten Sie im Listenfeld nur einen einzigen Namen wie »Bücher« selektiert, würde Excel nur jene Bezüge ersetzen, denen der Name »Bücher« zugeordnet ist.



9.4 Schnittpunkt-Operator und impliziter Schnittpunkt

Wenn Sie *Optionen* >> aktivieren, wird das *Namen übernehmen*-Dialogfeld erweitert (Abbildung 9.13).

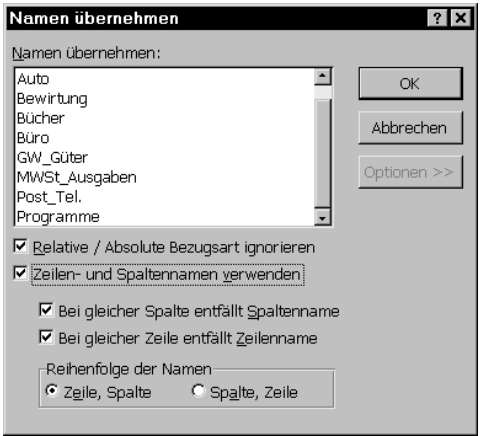


Abbildung 9.13:
Zusätzliche
Optionen

Zunächst sind darin alle Optionen aktiviert. Im Falle von *Relative/Absolute Bezugsart ignorieren* bedeutet das: Excel wird alle Bezüge durch zugehörige Namen ersetzen, soweit welche vorhanden sind, unabhängig vom verwendeten Bezugsformat.

*Bezugsart
ignorieren*

Deaktivieren Sie diese Option, so beachtet Excel beim Ersetzen die Bezugsart in folgender Art und Weise: Ist ein Bezug absolut, wird er nur dann durch den zugehörigen Namen ersetzt, wenn auch dieser mit absolutem Bezug festgelegt wurde! Das Gleiche gilt für relative Bezüge.

Angenommen, Sie haben A5 den Namen »Test« zugeordnet und dabei die von Excel unter *Bezieht sich auf* vorgegebenen absoluten Koordinaten übernommen, also \$A\$5. »Test« wurde also mit absolutem Bezug definiert.

Befindet sich bei dieser Annahme in A10 die Formel `=A$5`, wird `NAMEN|ÜBERNEHMEN...` darin den absoluten Bezug `$A$5` in jedem Fall durch den ebenfalls mit absolutem Bezug definierten Namen »Test« ersetzen, egal ob *Relative/Absolute Bezugsart ignorieren* aktiviert ist oder nicht.

Befindet sich in A10 jedoch die Formel `=A5`, wird dieser relative Bezug nur dann durch den mit absolutem Bezug definierten Namen »Test« ersetzt, wenn *Relative/Absolute Bezugsart ignorieren* aktiviert war!

**Zeilen-/Spalten-
namen
verwenden**

Zeilen- und Spaltennamen verwenden verfolgt ebenso wie die beiden folgenden Optionen den Zweck, auch Bezügen auf *einzelne Zellen von Bereichen* Namen zu geben. Obwohl nur für die kompletten Bereiche Namen definiert wurden und nicht für jede einzelne darin enthaltene Zelle!

Voraussetzung dazu ist jedoch, dass der betreffende Bereich zeilen- und spaltenorientiert ist (Abbildung 9.14).

Abbildung 9.14:
Übernahme aus
spalten- und zeilen-
orientiertem Bereich

	A	B	C
1		Einnahmen	Ausgaben
2	JAN	126	73
3	FEB	372	48
4	MÄR	712	391
5	APR	163	173
6	MAI	89	283
7	JUN	75	82
8	JUL	263	28
9	AUG	596	123
10	SEP	710	328
11	OKT	106	192
12	NOV	462	129
13	DEZ	284	193
14	Summen	3958	2043
15			
16	Quartalsüberschüsse		698
17			-211
18			1090
19			338

Selektieren Sie den Bereich A1:C13, also den Zahlenbereich inklusive der Zeilen- und der Spaltenüberschriften, wählen Sie `EINFÜGEN|NAMEN|ERSTELLEN...` und aktivieren Sie im zugehörigen Dialogfeld *Oberster Zeile* und(!) *Linker Spalte*.

Excel definiert genau 15 Namen, einen pro Spalten- bzw. Zeilenüberschrift:

Name	Zugehöriger Bereich
Summen	\$B\$14:\$C\$14
Einnahmen	\$B\$2:\$B\$13
Ausgaben	\$C\$2:\$C\$13
JAN	\$B\$2:\$C\$2
FEB	\$B\$3:\$C\$3
MÄR	\$B\$4:\$C\$4
APR	\$B\$5:\$C\$5
MAI	\$B\$6:\$C\$6
JUN	\$B\$7:\$C\$7
JUL	\$B\$8:\$C\$8
AUG	\$B\$9:\$C\$9
SEP	\$B\$10:\$C\$10
OKT	\$B\$11:\$C\$11
NOV	\$B\$12:\$C\$13
DEZ	\$B\$13:\$C\$13

Tabelle 9.2:
Spalten- und
zeilenorientierte
Bereichsbenennung

Allgemein: EINFÜGEN|NAMEN|ERSTELLEN... vergibt für einen Bereich mehrere Namen, je einen pro Unterbereich, wobei jede Spalte bzw. Zeile des selektierten Bereichs einen solchen Unterbereich bildet und entsprechend einen eigenen Namen erhält.



Je nachdem, ob nur Spalten- oder Zeilennamen oder beide gewünscht werden, müssen sich am betreffenden Bereichsrand (rechts, links, oben, unten) geeignete Überschriften befinden, die als Namen für die Unterbereiche verwendet werden.

Excel kann solche Unterbereichsnamen verwenden, um Formeln noch aussagekräftiger zu gestalten. Sehen Sie sich dazu das untere Ende der Tabelle an: Mehrere Zahlen werden dort zu Quartalsüberschüssen addiert. Zum Beispiel enthält C16 die Formel

$$=(B2-C2)+(B3-C3)+(B4-C4).$$

$(B2-C2)$ subtrahiert die Ausgaben im Januar von den Einnahmen in diesem Monat, ermittelt also der Einnahmeüberschuss dieses Monats.

**Schnittpunkt-
Operator**

Auf die gleiche Weise ermitteln (B3-C3) und (B4-C4) den Überschuss der Monate Februar und März. Die Addition dieser drei Zahlen führt schließlich zum Gesamtüberschuss im ersten Quartal. Die drei anderen Quartalsüberschüsse entstehen analog.

In diesen Formeln ist kein einziger Bezug enthalten, für den ein Name definiert wurde (siehe vorhergehende Tabelle). Dennoch kann Excel die definierten Bereichsnamen anwenden!

Wenn Sie C16:C19 selektieren, EINFÜGEN|NAMEN|ÜBERNEHMEN... wählen und alle Einstellungen unverändert übernehmen, enthält zum Beispiel C16 anschließend statt

$$=(B2-C2)+(B3-C3)+(B4-C4)$$

eine erheblich aussagekräftigere Formel (Abbildung 9.15):

$$=(\text{JAN Einnahmen}-\text{JAN})+(\text{FEB Einnahmen}-\text{FEB})+(\text{MÄR Einnahmen}-\text{MÄR})$$

Abbildung 9.15:
Kombination aus
Spalten- und
Zeilenname

$$=(\text{JAN Einnahmen}-\text{JAN})+(\text{FEB Einnahmen}-\text{FEB})+(\text{MÄR Einnahmen}-\text{MÄR})$$

	A	B	C	D
1		Einnahmen	Ausgaben	
2	JAN	126	73	
3	FEB	372	48	
4	MÄR	712	391	
5	APR	163	173	
6	MAI	89	283	
7	JUN	75	82	
8	JUL	263	28	
9	AUG	596	123	
10	SEP	710	328	
11	OKT	106	192	
12	NOV	462	129	
13	DEZ	284	193	
14	Summen	3958	2043	
15				
16	Quartalsüberschüsse		698	
17			-211	
18			1090	
19			338	
20				

Zum Beispiel wurde der Bezug B2 durch den Namen »JAN Einnahmen« ersetzt. Dabei wurde eine uns bisher unbekannte Bezugsform verwendet, der Schnittpunkt-Operator, ein Leerzeichen zwischen zwei Bezugssteilen, die für sich genommen beide nicht eindeutig sind.

Der Name »JAN« steht für den zeilenorientierten Bereich \$B\$2:\$C\$2 und der Name »Einnahmen« für den spaltenorientierten Bereich \$B\$2:\$B\$13. Die durch ein Leerzeichen getrennte Kombination »JAN Einnahmen« beider Namen referenziert jene Zelle, die sich im Schnittpunkt beider Bereiche befindet, also \$B\$2.

Dank der Option *Zeilen- und Spaltennamen verwenden* ersetzt Excel somit den fehlenden Namen für den Bezug \$B\$2 einfach durch die Kombination zweier Bereichsnamen, eines Zeilennamens und eines Spaltennamens, die zusammen B2 eindeutig identifizieren!

Eigentlich sollte man nach diesen Erläuterungen folgende Formel erwarten:

`=(JAN Einnahmen-JAN Ausgaben)+(FEB Einnahmen-FEB Ausgaben)+(MÄR Einnahmen-MÄR Ausgaben)`

*Impliziter
Schnittpunkt*

Seltsamerweise wurde jedoch beispielsweise der Bezug C2 nicht durch die Namenskombination »JAN Ausgaben« ersetzt, sondern einfach durch den Namen »JAN«, obwohl dieser für den Bereich \$B\$2:\$C\$2 steht und nicht wie gefordert für die Zelle C2!

Ursache dafür ist die aktivierte Option *Bei gleicher Spalte entfällt Spaltenname*. Sie bewirkt, dass Excel nicht unbedingt notwendige Spaltenangaben (Spaltennamen) weglässt und einen Bezug mit implizitem Schnittpunkt verwendet, eine bisher noch nicht erläuterte Bezugsform.

Bei dieser Bezugsart kann die Spaltenangabe entfallen, wenn sich die referenzierte Zelle *in der gleichen Spalte wie die Formelzelle selbst* befindet.

Der Bezug JAN in der in C16 enthaltenen Formel bezieht sich einfach auf jene Zelle des durch diesen Namen identifizierten Bereichs \$B\$2:\$C\$2, die sich in der gleichen Spalte C wie die Formelzelle befindet, also auf die Zelle C2.

Ist *Bei gleicher Zeile entfällt Zeilenname* aktiviert, entfallen analog dazu überflüssige Zeilenamen bei Bezügen, die sich *in der gleichen Zeile wie die Formelzelle* befinden.

Als Letztes können Sie noch die Reihenfolge der Spalten- und Zeilenamen festlegen: Ist die Option *Zeile, Spalte* aktiviert, nennt Excel zuerst den Zeilen- und dann den Spaltennamen, zum Beispiel »JAN Einnahmen«. Bei Aktivierung von *Spalte, Zeile* würde Excel den Bezug stattdessen »Einnahmen JAN« nennen.

*Bezugs-
reihenfolge*

10 Gliederungen

Gliederungen verbessern den Überblick über umfangreiche Tabellen. Ich erläutere nun, wie Sie nebensächliche Details auf Tastendruck bzw. per Mausklick aus- oder wieder einblenden. Dabei handelt es sich immer um komplette Zeilen oder Spalten, die – zumindest vorübergehend – uninteressant sind und verschwinden sollen; sie werden zu Gruppen zusammengefasst und versteckt, indem ihnen Excel die Zeilenhöhe bzw. Spaltenbreite 0 zuweist.

Excel kann derartige Gliederungen selbst erstellen, wenn – wie in den meisten Tabellen – die Priorität der einzelnen Bereiche offensichtlich ist, sich also oben einzelne Werte befinden, diese Werte weiter unten addiert oder sonstige Zwischenergebnisse berechnet werden und darunter wiederum aus diesen Zwischenergebnissen das benötigte Endresultat ermittelt wird.

Egal, ob Sie Gliederungen vollautomatisch von Excel erstellen lassen oder manuell erstellen, anschließend ist es in jedem Fall möglich, auf Knopfdruck ausgewählte Gliederungselemente aus- oder einzublenden und damit unterschiedliche Ansichten ein und derselben Tabelle zu erzeugen.

Gliederungen verbessern den Überblick über umfangreiche Tabellen durch das Aus- und Wiedereinblenden nebensächlicher Details auf Tastendruck bzw. Mausklick.

10.1 Automatische Gliederungserstellung

In den meisten Tabellen ist die Priorität der einzelnen Bereiche offensichtlich: Oben befinden sich einzelne Werte, weiter unten werden diese Werte addiert oder sonstige Zwischenergebnisse berechnet. Darunter wiederum wird aus diesen Zwischenergebnissen das interessierende Endresultat ermittelt.

Möglicherweise ist die Tabelle stattdessen in prinzipiell gleicher Weise von links nach rechts statt von oben nach unten strukturiert.

Nur wenn eine solche durchgehende Ordnung vorhanden ist, kann Excel automatisch vernünftige Gliederungen erstellen. Es ist jedoch nicht möglich, wenn sich zum Beispiel in den Zeilen 5 bis 10 einzelne Werte befinden, in

Zeile 15 daraus ein Zwischenergebnis berechnet und inkonsequenterweise am oberen Tabellenrand in Zeile 1 daraus wiederum das Endergebnis ermittelt wird. Eine halbwegs durchdachte Tabelle kann Excel jedoch meist vollautomatisch gliedern.

Probieren wir das an unserer Buchführung aus. Ist bei Anwahl von DATEN|GRUPPIERUNG UND GLIEDERUNG|AUTOGLIEDERUNG kein Bereich selektiert, erstellt Excel die Gliederung für die gesamte Tabelle. Wäre ein Bereich selektiert (komplette Zeilen oder Spalten), würde sich die Gliederung auf diesen Bereich beschränken.

Eine Gesamtgliederung für die Tabelle »Jan« von BUCHFRG.XLS führt zu folgendem Resultat (Abbildung 10.1):

Abbildung 10.1:
AutoGliederung

10-001.xls											
1	2	3	A	B	C	D	E	F	G	H	I
1	Einnahmen					Ausgaben					
2	Bücher Programme					Post+Tel.	Büro	Auto	Bewirtung	3W-Güter	Abschreibung
3	230,00 3859,50					2,00	19,80	284,00	184,00	473,00	1728,00
4	1637,58					4,00	53,20	76,50	396,00		
5						3,70		73,00			
6								68,90			
7											
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00		
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34		
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66		
11											
12						Netto	MWSt				
13						- Steuerpflichtige Umsätze mit 16%	3327,16	532,34			
14						- Steuerpflichtige Umsätze mit 7%	1745,40	122,18			
15	Gesamt						5072,56	654,52			
16	Vorsteuerbeträge							462,95			
17	Umsatzsteuervorauszahlung							191,57			
18											
Jan Feb Mar Apr Jun Jul Aug Sep Okt Nov Dez											

Hauptzeilen Ziel der Gliederungsfunktion ist es, wichtige Hauptzeilen hervorzuheben, unwichtige Detaildaten jedoch auf tieferen Gliederungsebenen zu verbergen.

Jede Gliederungsebene besitzt eine Nummer. Auf der Ebene mit der höchsten Nummer, im Beispiel ist das die momentan aktive Ebene 3, ist die gesamte Tabelle sichtbar. Auf Ebene 2 werden die unwichtigsten Detaildaten ausgeblendet. Auf Ebene 1 werden bereits relativ wichtige Daten ausgeblendet, sodass auf dieser tiefsten Ebene nur noch die absolut unverzichtbaren Teile der Tabelle sichtbar sind.

Die unwichtigsten Detaildaten sind im Beispiel zweifellos die Werte in den Zeilen 3 bis 7. Sie sind eindeutig weniger interessant als die zugehörigen Summen in den Zeilen darunter und werden bereits auf Ebene 2 ausgeblendet. Aktivieren Sie diese Gliederungsebene, indem Sie das Symbol 2 anklicken (Abbildung 10.2).

	A	B	C	D	E	F	G	H	I
1	Einnahmen			Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	3W-Güter	Abschreibung
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
12				Netto	MWSt				
13				- Steuerpflichtige Umsätze mit 16%	3327,16	532,34			
14				- Steuerpflichtige Umsätze mit 7%	1745,40	122,18			
15	Gesamt				5072,56	654,52			
16	Vorsteuerbeträge					462,95			
17	Umsatzsteuervorauszahlung					191,57			

Abbildung 10.2:
Eine Ebene tiefer

Die Zeilen 3 bis 7 sind nun nicht mehr sichtbar. Klicken Sie das Symbol 1 an, so bewegen Sie sich erneut nach unten, auf die Gliederungsebene 1 (Abbildung 10.3).

	A	B	C	D	E	F	G	H	I
1	Einnahmen			Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	3W-Güter	Abschreibung
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
12				Netto	MWSt				
17	Umsatzsteuervorauszahlung					191,57			

Abbildung 10.3:
Die tiefste Gliederungsebene

Auf dieser Ebene werden zusätzlich zu den Detaildaten auch die Zeilen 8 und 9 ausgeblendet, die Zwischenergebnisse enthalten.

Zeile 8 enthält Formeln wie `=SUMME(C3:C7)` (in Zelle C8), die sich auf den bereits zuvor ausgeblendeten Detailbereich beziehen und die darin enthaltenen Werte addieren.

Zeile 9 wiederum enthält Formeln wie `=C8/116*16` (in Zelle C9), die die Umsatzsteuer berechnen, die in den auf diese Weise in Zeile 8 ermittelten Summen enthalten ist.

Beide Zwischenergebnisse sind unwichtig im Vergleich zu Zeile 10, die Formeln wie `=C8-C9` enthält (in Zelle C10), die also Ergebnisse liefern, die auf den beiden Zeilen 8 und 9 basieren. Zeile 10 ist daher höherwertiger als die Zeilen 8 und 9 und bleibt eingeblendet, wenn die beiden Zwischenergebnisse 8 und 9 ausgeblendet werden.

Ähnliches gilt für die Zeilen 13 bis 17: In den Zeilen 13 bis 16 werden Zwischenergebnisse errechnet, auf denen das in Zeile 17 ermittelte Endergebnis basiert, die Umsatzsteuervorauszahlung. Also können auf dieser Ebene zwar die Zeilen 13 bis 16 ausgeblendet werden, die Zeile 17 mit dem darauf basierenden Endergebnis muss jedoch erhalten bleiben.

Um wieder auf die höchste Gliederungsebene zu gelangen, auf der die gesamte Tabelle sichtbar ist, klicken Sie einfach auf das Symbol 3.



Merken Sie sich bitte das Prinzip: Sie aktivieren eine Gliederungsebene durch Anklicken der zugehörigen Ebenennummer. Auf der betreffenden Ebene sind alle Zeilen ausgeblendet, die von den senkrechten Strichen unterhalb dieser Nummer umklammert werden.

Zusätzlich zeigen die Punkte (und die »-«-Symbole) neben den Zeilennummern an, ab welcher Ebene die betreffende Zeile sichtbar ist: ab jener, unter deren Ebenennummer sich der zugehörige Punkt befindet (und natürlich auch auf allen höheren Ebenen mit größeren Ebenennummern).

Gliederungen können auch spaltenorientiert sein. Die Hauptdaten befinden sich dann nicht unterhalb, sondern rechts von den Detaildaten und statt Zeilen werden entsprechend Spalten ausgeblendet (Abbildung 10.4).

Abbildung 10.4:
Spaltenorientierte
Gliederung

	A	B	C	D
1				Summe
2	Büro	295	233	528
3	Post	68	152	220
4	Auto	690	424	1114
5				

Klicken Sie 1 an, so werden die Spalten B und C mit den Detaildaten ausgeblendet und es sind nur noch die Spalten A (die Spaltenüberschriften) und D (die Summen der ausgeblendeten Detailwerte) sichtbar.

Gliederungsart
beeinflussen

Mit DATEN|GRUPPIERUNG UND GLIEDERUNG|EINSTELLUNGEN... können Sie die Art und Weise beeinflussen, in der Excel eine automatische Gliederung erzeugt (Abbildung 10.5).

Abbildung 10.5:
Das Einstellungs-
Dialogfeld

Einstellungen

Richtung

☒ Hauptzeilen unter Detaildaten

☒ Hauptspalten rechts von Detaildaten

☐ Automatische Formatierung

Erstellen

Formatvorlage übernehmen

OK

Abbrechen

Erstellen erzeugt ebenso wie DATEN|GRUPPIERUNG UND GLIEDERUNG|AUTO-GLIEDERUNG eine automatische Gliederung, ausgehend von den aktuellen Einstellungen.

Automatische Formatierung formatiert Hauptzeilen, beispielsweise kursiv oder fett, und hebt sie dadurch auch auf der höchsten Ebene, auf der die gesamte Tabelle sichtbar ist, von den Detaildaten ab.

Sie können eine solche Formatierung auch nachträglich durchführen, indem Sie den zu formatierenden Tabellenbereich selektieren, DATEN|GRUPPIERUNG UND GLIEDERUNG|EINSTELLUNGEN... wählen und auf *Formatvorlage übernehmen* klicken.

Mit *Hauptzeilen unter Detaildaten* und *Hauptspalten rechts von Detaildaten* können Sie Excel angeben, wo sich die Hauptzeilen bzw. -spalten befinden, wenn eine Tabelle nicht eindeutig aufgebaut ist.

Im Normalfall lassen Sie beide vorselektierten Optionen unverändert, sodass sich Excel den zutreffenden Fall selbst aussuchen kann. Bei fehlerhaften Resultaten aktivieren Sie stattdessen nur die tatsächlich zutreffende Option, bei der zeilenweise angeordneten Buchführungs-Tabelle beispielsweise *Hauptzeilen unter Detaildaten*.

Mit DATEN|GRUPPIERUNG UND GLIEDERUNG|GLIEDERUNG ENTFERNEN können Sie jederzeit die gesamte Gliederung entfernen.

*Gliederung
entfernen*

Deaktivieren Sie im Register *Ansicht* des Befehls EXTRAS|OPTIONEN... das Kontrollkästchen *Gliederungs-Symbole*, werden die Gliederungs-Symbole vorübergehend ausgeblendet. Die Gliederung selbst bleibt dabei jedoch erhalten und kann durch Aktivierung des Kontrollkästchens jederzeit wieder eingeblendet werden.

*Gliederung
ausblenden*

10.2 Manuelle Gliederungen

Sie können Gliederungen vollständig manuell erstellen oder automatisch erzeugte Gliederungen gezielt nachbearbeiten.

Der Befehl DATEN|GRUPPIERUNG UND GLIEDERUNG|GRUPPIERUNG... (**Alt** +  + ) bzw. das zugehörige Symbol stuft (in einem Dialogfeld zu selektierende) Zeilen/Spalten herab, sodass sie nur noch auf der höchsten Gliederungsebene angezeigt werden. Stufen Sie die betreffenden Zeilen/Spalten erneut herab, werden sie bereits auf der zweithöchsten Ebene nicht mehr angezeigt und so weiter.

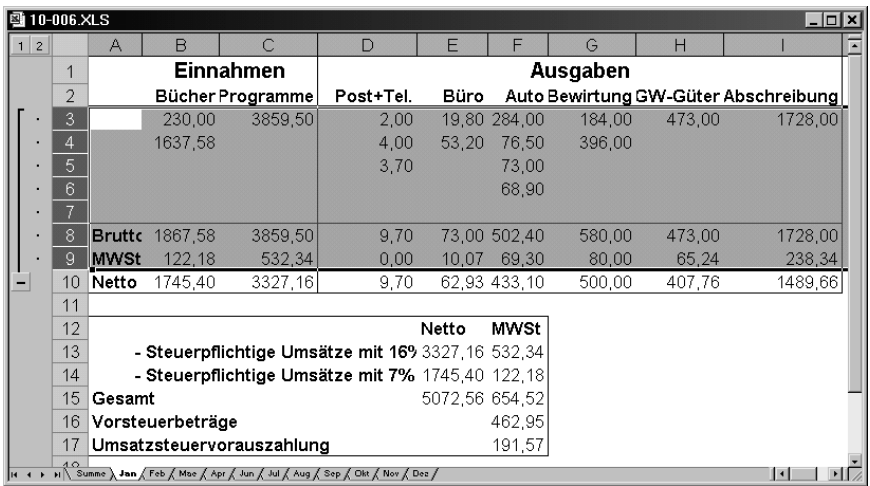




DATEN|GRUPPIERUNG UND GLIEDERUNG|GRUPPIERUNG AUFHEBEN... (Alt +  + ) bzw. das zugehörige Symbol stuft selektierte Zeilen/Spalten genau umgekehrt bei jedem Anklicken eine Ebene herauf, sodass sie anschließend auch auf tieferen Ebenen als zuvor angezeigt werden.

Wird das auf die Tabelle »Jan« in BUCHFRG.XLS (nach Entfernung der vorhergehenden AutoGliederung mit DATEN|GRUPPIERUNG UND GLIEDERUNG|GLIEDERUNG ENTFERNEN) angewandt, dann sollen (im Gegensatz zur vorhergehenden automatischen Gliederung) bereits auf der zweithöchsten Ebene nicht nur die Zeilen 3 bis 7 mit den Detaildaten, sondern zusätzlich auch die Zwischenergebniszeilen 8 und 9 ausgeblendet werden. Selektieren Sie all diese Zeilen, also 3 bis 9, und wählen Sie DATEN|GRUPPIERUNG UND GLIEDERUNG|GRUPPIERUNG... (ABBILDUNG 10.6).

Abbildung 10.6:
Herabstufen



	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher Programme		Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutt	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12					Netto	MWSt			
13					- Steuerpflichtige Umsätze mit 16%	3327,16	532,34		
14					- Steuerpflichtige Umsätze mit 7%	1745,40	122,18		
15	Gesamt				5072,56	654,52			
16	Vorsteuerbeträge					462,95			
17	Umsatzsteuervorauszahlung					191,57			

Sollen auf dieser Ebene zusätzlich auch noch die Zeilen 13 und 14 ausgeblendet werden, dann selektieren Sie diese anschließend und wählen erneut DATEN|GRUPPIERUNG UND GLIEDERUNG|GRUPPIERUNG..., um auch diese beiden Zeilen herabzustufen (Abbildung 10.7).

Soll noch eine Ebene tiefer der gesamte Bereich von Zeile 3 bis Zeile 16 ausgeblendet werden, dann selektieren Sie ihn und wählen diesen Befehl erneut (Abbildung 10.8).

Der selektierte Bereich enthält die beiden vorhergehenden Zeilengruppen als Untermenge. Da das Gliederungsprinzip lautet, dass auf jeder tieferen Ebene weniger sichtbar sein soll als auf der Ebene zuvor, verleiht Excel der umschließenden Gruppe eine höhere Priorität als den davon umschlossenen Gruppen und legt eine weitere Gliederungsebene an.

	A	B	C	D	E	F	G	H	I
1	Einnahmen			Ausgaben					
2	Bücher Programme			Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3	230,00	3859,50		2,00	19,80	284,00	184,00	473,00	1728,00
4	1637,58			4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12				Netto	MWSt				
13	-	Steuerpflichtige Umsätze mit 16%	3327,16	532,34					
14	-	Steuerpflichtige Umsätze mit 7%	1745,40	122,18					
15	Gesamt			5072,56	654,52				
16	Vorsteuerbeträge				462,95				
17	Umsatzsteuervorauszahlung				191,57				

Abbildung 10.7:

Zweiten Bereich
herabstufen

	A	B	C	D	E	F	G	H	I
1	Einnahmen			Ausgaben					
2	Bücher Programme			Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3	230,00	3859,50		2,00	19,80	284,00	184,00	473,00	1728,00
4	1637,58			4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12				Netto	MWSt				
13	-	Steuerpflichtige Umsätze mit 16%	3327,16	532,34					
14	-	Steuerpflichtige Umsätze mit 7%	1745,40	122,18					
15	Gesamt			5072,56	654,52				
16	Vorsteuerbeträge				462,95				
17	Umsatzsteuervorauszahlung				191,57				

Abbildung 10.8:

Umfassenderen
Bereich herabstufen

Vielleicht überlegen Sie nachträglich, dass der von Zeile 3 bis 7 reichende Detailbereich doch unwichtiger ist als die Zwischenergebnisse in den Zeilen 13 und 14 und noch vor diesen ausgeblendet werden soll.

Kein Problem: Sie selektieren die betreffenden Zeilen 3 bis 7 und wählen wieder DATEN|GRUPPIERUNG UND GLIEDERUNG|GRUPPIERUNG... (Abbildung 10.9).

Da der selektierte Bereich bereits von einer Gruppe umschlossen ist, besitzt er eine niedrigere Priorität als diese und wird eine Ebene zuvor ausgeblendet.

Stellen Sie nun fest, dass Sie die Zeilen 13 und 14 doch für ebenso unwichtig halten, wollen Sie sie also auf die gleiche Ebene herabstufen, dann selektieren Sie sie und wählen den Befehl erneut (Abbildung 10.10).

Abbildung 10.9:
Umfassten Bereich
herabstufen

	A	B	C	D	E	F	G	H	I
1	Einnahmen			Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
11									
12									
13									
14									
15									
16									
17									

Abbildung 10.10:
Von einer Ebene zur
anderen
verschieben

	A	B	C	D	E	F	G	H	I
1	Einnahmen			Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreib
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	172
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	172
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	23
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	148
11									
12									
13									
14									
15	Gesamt								
16	Vorsteuerbeträge								
17	Umsatzsteuervorauszahlung								

Diesmal müssen Sie sehr genau hinschauen, um den Unterschied zur vorhergehenden Abbildung zu erkennen: Die Klammer um die Zeilengruppe verändert ihre Position zwar nicht, die beiden Punkte werden jedoch nach rechts verschoben, zur höchsten Ebene.

Das bedeutet, dass diese beiden Zeilen nun ebenso wie der Detailbereich nur noch auf dieser höchsten Ebene 4 sichtbar und bereits auf Ebene 3 ausgeblendet sind.

Statt kompletter Zeilen/Spalten können Sie auch einzelne Zellen selektieren, beispielsweise den Bereich B4:B7. Da Excel in diesem Fall jedoch nicht weiß, ob nun die Zeilen 4 bis 7 oder aber die Spalte B herab- oder heraufgestuft werden soll, erscheint folgende Frage (Abbildung 10.11):



Abbildung 10.11:
Spalten- oder
Zeilengruppierung?

Zeilen würde die Zeilen 4 bis 7 behandeln, *Spalten* die Spalte B.

10.3 Selektives Aus- und Einblenden

Statt gleich die Gliederungsebene zu wechseln, können Sie auch einzelne Gruppen ein- oder ausblenden.

Klicken Sie einen der mit »-« beschrifteten Knöpfe an, so wird die zugehörige Gruppe ausgeblendet und zwar unabhängig von der Gliederungsebene, in der Sie sich befinden!

Wird das auf die vorletzte Abbildung angewandt, so können Sie beispielsweise das »-«-Symbol anklicken, das zur Zeilengruppe 3 bis 9 gehört, um diese Gruppe jederzeit auszublenden, auch wenn Sie sich gerade auf der höchsten Ebene befinden (Abbildung 10.12).

10-012.XLS												
1	2	3	4	A	B	C	D	E	F	G	H	I
	1	Einnahmen					Ausgaben					
	2	Bücher			Programme		Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschre
	10	Netto	1745,40	3327,16			9,70	62,93	433,10	500,00	407,76	14
	11											
	12	Netto MWSt										
	13	- Steuerpflichtige Umsätze mit 16%					3327,16	532,34				
	14	- Steuerpflichtige Umsätze mit 7%					1745,40	122,18				
	15	Gesamt					5072,56	654,52				
	16	Vorsteuerbeträge						462,95				
	17	Umsatzsteuervorauszahlung						191,57				
	18											
	19											
	20											
	21											
	22											
	23											
	24											
	25											
	26											
	27											
	28											
	29											
	30											
	31											
	32											
	33											
	34											
	35											
	36											
	37											
	38											
	39											
	40											
	41											
	42											
	43											
	44											
	45											
	46											
	47											
	48											
	49											
	50											
	51											
	52											
	53											
	54											
	55											
	56											
	57											
	58											
	59											
	60											
	61											
	62											
	63											
	64											
	65											
	66											
	67											
	68											
	69											
	70											
	71											
	72											
	73											
	74											
	75											
	76											
	77											
	78											
	79											
	80											
	81											
	82											
	83											
	84											
	85											
	86											
	87											
	88											
	89											
	90											
	91											
	92											
	93											
	94											
	95											
	96											
	97											
	98											
	99											
	100											
	101											
	102											
	103											
	104											
	105											
	106											
	107											
	108											
	109											
	110											
	111											
	112											
	113											
	114											
	115											
	116											
	117											
	118											
	119											
	120											
	121											
	122											
	123											
	124											
	125											
	126											
	127											
	128											
	129											
	130											
	131											
	132											
	133											
	134											
	135											
	136											
	137											
	138											
	139											
	140											
	141											
	142											
	143											
	144											
	145											
	146											
	147											
	148											
	149											
	150											
	151											
	152											
	153											
	154											
	155											
	156											
	157											
	158											
	159											
	160											
	161											
	162											
	163											
	164											
	165											
	166											
	167											
	168											
	169											
	170											
	171											
	172											
	173											
	174											
	175											
	176											
	177											
	178											
	179											
	180											
	181											
	182											
	183											
	184											
	185											
	186											
	187											
	188											
	189											
	190											
	191											
	192											
	193											
	194											
	195											
	196											
	197											
	198											
	199											
	200											
	201											
	202											
	203											
	204											
	205											
	206											
	207											
	208											
	209											
	210											
	211											
	212											
	213											
	214											
	215											
	216											
	217											
	218											
	219											
	220											
	221											
	222											
	223											
	224											
	225											
	226											
	227											
	228											
	229											
	230											
	231											
	232											
	233											
	234											
	235											
	236											
	237											
	238											
	239											
	240											
	241											
	242											
	243											
	244											
	245											
	246											
	247											
	248											
	249											
	250											
	251											
	252											
	253											
	254											
	255											
	256											
	257											
	258											
	259											
	260											
	261											
	262											
	263											
	264											
	265											
	266											
	267											
	268											
	269											
	270											
	271											
	272											
	273											
	274											
	275											
	276											
	277											
	278											
	279											
	280											
	281											
	282											
	283											
	284											
	285											
	286											
	287											
	288											
	289											
	290											
	291											
	292											
	293											
	294											
	295											
	296											
	297											
	298											
	299											
	300											
	301											
	302											
	303											
	304											
	305											
	306											
	307											
	308											
	309											
	310											
	311											
	312											
	313											
	314											
	315											
	316											
	317											
	318											
	319											
	320											
	321											
	322											
	323											
	324											
	325											
	326											
	327											
	328											
	329											
	330											
	331											
	332											
	333											
	334											
	335											
	336											
	337											
	338											
	339											
	340											
	341											
	342											
	343											
	344											
	345											
	346											
	347											
	348											
	349											
	350											
	351											
	352											
	353											
	354											
	355											
	356											
	357											
	358											
	359											
	360											
	361											
	362											
	363											
	364											
	365											
	366											
	367											
	368											
	369											
	370											
	371											
	372											
	373											
	374											
	375											
	376											
	377											
	378											
	379											
	380											
	381											
	382											
	383											
	384											
	385											
	386											
	387											
	388											
	389											
	390											
	391											
	392											
	393											
	394											
	395											
	396											
	397											
	398											
	399											
	400											
	401											
	402											
	403											
	404											
	405											
	406											
	407											
	408											
	409											
	410											
	411											
	412											
	413											
	414											
	415											
	416											
	417											
	418											
	419											
	420											
	421											
	422											
	423											
	424											
	425											
	426											
	427											
	428											
	429											
	430											
	431											
	432											
	433											
	434											
	435											
	436											
	437											
	438											
	439											
	440											
	441											
	442											
	443											
	444											
	445											
	446											
	447											
	448											
	449											
	450											
	451											
	452											
	453											
	454											
	455											
	456											
	457											
	458											
	459											
	460											
	461											
	462											
	463											
	464											
	465											
	466											
	467											
	468											
	469											
	470											
	471											
	472											
	473											
	474											
	475											
	476											
	477											
	478											
	479											
	480											
	481											
	482											
	483											
	484											
	485											
	486											
	487											
	488											
	489											
	490											
	491											
	492											
	493											
	494											
	495											
	496											
	497											
	498											
	499											
	500											
	501											
	502											
	503											
	504											
	505											
	506											
	507											
	508											
	509											

Abbildung 10.12:
Selektives
Ausblenden

Umgekehrt blendet Klicken auf das zugehörige »+«-Symbol eine Gruppe ein, egal auf welcher Gliederungsebene Sie sich befinden.

Eine verwandte Aufgabe hat der Befehl DATEN|GRUPPIERUNG UND GLIEDERUNG|DETAIL AUSBLENDEN. Ist momentan ein Bereich selektiert, blendet er ebenso wie das zugehörige Symbol alle darin enthaltenen Gruppen aus (Abbildung 10.13).



Der selektierte Bereich umfasst die beiden von Zeile 3 bis 7 und die von Zeile 3 bis 10 reichenden Gruppen, die entsprechend ausgeblendet werden (Abbildung 10.14).

Abbildung 10.13:
Detailbereich
ausblenden
vorbereiten

10-013.XLS

	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschrei
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	17
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	17
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	2
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	14
11									
12						Netto	MWSt		
13						- Steu	- Steu		
14						pflichtige Umsätze mit 16%	pflichtige Umsätze mit 7%		
15	Gesamt					3327,16	532,34		
16	Vorsteuerbeträge					1745,40	122,18		
17	Umsatzsteuervorauszahlung					5072,56	654,52		
							462,95		
							191,57		

Summe Jan Feb Mar Apr Jun Jul Aug Sep Okt Nov Dez

Abbildung 10.14:
Nach dem
Ausblenden

10-014.XLS

	A	B	C	D	E	F	G	H	I
1		Einnahmen		Ausgaben					
2		Bücher	Programme	Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschre
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	14
11									
12						Netto	MWSt		
13						- Steu	- Steu		
14						pflichtige Umsätze mit 16%	pflichtige Umsätze mit 7%		
15	Gesamt					3327,16	532,34		
16	Vorsteuerbeträge					1745,40	122,18		
17	Umsatzsteuervorauszahlung					5072,56	654,52		
							462,95		
							191,57		

Summe Jan Feb Mar Apr Jun Jul Aug Sep Okt Nov Dez



Umgekehrt blendet DATEN|GRUPPIERUNG UND GLIEDERUNG|DETAIL ANZEIGEN ebenso wie das zugehörige Symbol alle in einem selektierten Bereich enthaltenen Gruppen wieder ein.

11 Funktionen

Grundsätzlich kennen Sie Funktionen zwar bereits, das folgende Kapitel geht jedoch weitaus detaillierter auf Funktionen ein, als das bisher der Fall war.

Sie erfahren, welche Funktionstypen es gibt (numerische Funktionen, Stringfunktionen etc.), wie diesen Funktionen Argumente übergeben werden und wie Sie Funktionen verschachteln, das heißt beim Aufruf einer Funktion als Argument wiederum eine andere Funktion angeben.

Nach dieser prinzipiellen Erläuterung gehe ich anhand von Beispielen auf die verschiedenen Funktionsgruppen ein, also auf mathematische Funktionen, Statistikfunktionen und so weiter.

Sehen Sie es mir bitte nach, dass ich mir aus jeder Gruppe nur einige wenige Funktionen herauspicke, die stellvertretend für alle anderen Funktionen der betreffenden Gruppe herhalten müssen.

Da Excel Hunderte von Funktionen kennt (über 400!), wäre ein eigenes Buch notwendig, um jede einzelne davon auch nur grob zu erläutern. Allerdings ist das nicht notwendig, da die Online-Hilfe den Zweck und die Anwendung jeder Tabellenfunktion sehr ausführlich erläutert.

Daher werde ich mich nun auf die Beschreibung der Grundprinzipien und der verschiedenen Funktionskategorien beschränken, die ich anhand einiger ausgewählter Funktionen erläutere.

11.1 Funktionsvorschrift und Argument

Eine Funktion ist eine Rechenvorschrift, die bestimmt, welche Operation mit einem oder mehreren Werten ausgeführt wird, den Funktionsargumenten. Das Ergebnis dieser Operation ist der Funktionswert.

Zum Beispiel erwartet die *SUMME*-Funktion als Argument die Angabe eines Bereichs. Da Argumente immer in Klammern übergeben werden, lautet der Aufruf der *SUMME*-Funktion

`SUMME(Bereich)`

Oft können oder müssen einer Funktion mehrere Argumente übergeben werden, *SUMME* beispielsweise bis zu 14. Jedes Argument kann eine Konstante, ein Zell- oder ein Bereichsbezug sein und muss vom vorhergehenden durch ein Semikolon getrennt werden! Die exakte Syntax lautet

```
SUMME(Zahl1;Zahl2;...)
```

Die Pünktchen bedeuten, dass das letzte Argument noch mehrfach wiederholt werden kann. Ein möglicher Aufruf von *SUMME* wäre daher

```
SUMME(1;2;3)
```

Dieser Aufruf übergibt *SUMME* drei Konstanten als Argumente. *SUMME* addiert die drei Zahlen und übergibt als Funktionswert das Ergebnis 6.

Einen Funktionswert können Sie wie jeden anderen Wert auch weiterverwenden, z.B. in folgender Formel:

```
=2*SUMME(1;2;3)
```

Sie ruft *SUMME* auf, um die drei Zahlen 1, 2 und 3 zu addieren. Anschließend wird der resultierende Funktionswert 6 mit 2 multipliziert und das Ergebnis 12 angezeigt.

Sie könnten auch eine Formel verwenden, in der *SUMME* die Zellen zweier Bereiche summiert und zusätzlich die Inhalte zweier separater Zellen und einer Konstanten addiert:

```
=SUMME(A1:A5;B1:B10;C25;X4;100)
```

SUMME addiert die Inhalte der Zellen A1 bis A5, B1 bis B10, C25 und X4 und die Konstante 100.

11.2 Verschachtelte Funktionsaufrufe

Sie können beim Aufruf einer Funktion als Argument einen anderen Funktionsaufruf verwenden:

```
SUMME(A1:C1;SUMME(A5:D5))
```

- ➔ Das erste Argument ist ein Bereich.
- ➔ Das zweite Argument ist selbst ein Funktionsaufruf. Dieser Aufruf wird nun ausgeführt, *SUMME* aufgerufen und als Argument der Bereich A5:D5 übergeben.

Nehmen wir an, dieser Bereich enthält die Zahlen 1, 2, 3 und 4. Dann übergibt der innere Aufruf von *SUMME* als Funktionswert die Zahl 10.

➔ Dem äußeren Aufruf von *SUMME* werden nun zwei Argumente übergeben: der Bereich A1:C1 bzw. die Inhalte der darin enthaltenen Zellen. Und die Zahl 10, also der Funktionswert des inneren Aufrufs von *SUMME*.

Eine solche Verschachtelung von Funktionsaufrufen setzt voraus, dass der resultierende Funktionswert der aufgerufenen inneren Funktion jenen Datentyp besitzt, der von der äußeren Funktion als Argument gefordert wird!

11.3 Datentypen

Ein Argument kann eine Konstante, ein Zellbezug oder ein Bereich sein.

In allen Fällen kann auch ein Name verwendet werden, wenn die Konstante, der Zellbezug oder der Bereich benannt wurde.

Auch eine Matrix kann ein Argument sein, zum Beispiel die Matrixkonstante {12;15;18}.

Welche Argumente Sie verwenden können, hängt von der betreffenden Funktion ab. Nicht jede Funktion akzeptiert jede Argumentart. Einige Funktionen erwarten sogar überhaupt keine Argumente. Vor allem aber akzeptiert nicht jede Funktion jeden Datentyp! Die gebräuchlichsten Datentypen sind Zahlen, Texte, Zeit, Datum, Matrizen und Bezüge.

Weitere wichtige Typen sind Fehlerwerte und Wahrheitswerte. Fehlerwerte sind Ausdrücke wie *#ZAHL!* oder *#DIV/0*, mit denen Excel signalisiert, dass ein Fehler auftrat.

Fehlerwerte

#DIV/0 zeigt beispielsweise an, dass die Formel in der betreffenden Zelle eine Division durch Null durchführt, die bekanntlich nicht definiert ist.

Es gibt genau zwei Wahrheitswerte: *WAHR* und *FALSCH*. Beide spielen vor allem bei den noch zu besprechenden logischen Funktionen eine große Rolle. Dabei entspricht der Wahrheitswert *FALSCH* intern der Zahl 0 und *WAHR* einer beliebigen Zahl ungleich 0.

Wahrheitswerte

Ist die Übergabe eines solchen Wahrheitswerts gefordert, können Sie somit zwar einfach 0 bzw. irgendeine andere Zahl übergeben. Im Interesse der Kompatibilität zu zukünftigen Excel-Versionen sollten Sie jedoch stattdessen die offiziellen Werte *WAHR* und *FALSCH* verwenden, zum Beispiel so:

Schlecht: =TREND(A11:F11;A11:F11;G11:I11;1)

Besser: =TREND(A11:F11;A11:F11;G11:I11;Wahr)

Beide Aufrufe der Funktion *TREND* funktionieren in der aktuellen Version von Excel. Es ist jedoch nicht gewährleistet, dass die erste Version, die nicht den Wahrheitswert selbst, sondern die zugehörige Zahl übergibt, auch in zukünftigen Versionen funktioniert, da die Microsoft-Programmierer den Ausdrücken *WAHR* und *FALSCH* in der nächsten Excel-Version ebenso gut andere interne Werte zuordnen könnten!

Argumente Sie können in einer Funktion als Argument jeden beliebigen Ausdruck verwenden, solange er dem verlangten Datentyp entspricht. Zum Beispiel erwartet die Funktion *SUMME* als Argumente Zahlen. Sie können jeden Ausdruck als Argument verwenden, der eine Zahl ergibt:

```
=SUMME(12+5;100*2-10;A1)
```

Die Auswertung dieser Formel erfolgt in vier Schritten:

- ➔ Excel wertet den Ausdruck $12+5$ aus; das Ergebnis ist die Zahl 17.
- ➔ Anschließend wird der Ausdruck $100*2-10$ mit 190 als Resultat ausgewertet.
- ➔ Der dritte Ausdruck ist ein Bezug. Er wird beim Aufruf von *SUMME* durch die in A1 enthaltene Zahl ersetzt, sagen wir durch 20.
- ➔ Zum Schluss addiert *SUMME* die vier Argumente und übergibt als Funktionswert 227.

Das einzige problematische Argument ist der Bezug A1, bei dem nicht offensichtlich ist, worum es sich dabei handelt.

Enthält A1 den Text »Ein Test«, so erhalten Sie garantiert eine Fehlermeldung, denn in diesem Fall ist das *SUMME* übergebene Argument nicht vom Typ Zahl, den diese Funktion fordert, sondern vom Typ Zeichenkette.

Es gibt alle möglichen Arten von Funktionswerten. Zum Beispiel wandelt die Funktion *GROSS* alle Zeichen eines Textes in Großbuchstaben um. Der Funktionswert ist also keine Zahl, sondern wie im folgenden Beispiel eine Zeichenkette, ein so genannter String:

```
=GROSS("Test")
```

Hier wird *GROSS* die Zeichenkette »Test« übergeben. Beachten Sie, dass als Argumente übergebene Zeichenketten immer in Anführungszeichen eingeschlossen werden!

Die Formel bewirkt, dass als Inhalt des Feldes, in dem sie sich befindet, der String »TEST« angezeigt wird. Die *GROSS*-Funktion erwartet somit als Argument einen String und übergibt als Funktionswert selbst wieder einen String.

Ein weiteres Beispiel: Da *SUMME* als Argumente Zahlen enthält, wäre ein Aufruf wie

```
=SUMME(3; 6; GROSS("Test"))
```

offensichtlich fehlerhaft. Denn das dritte Argument *GROSS*(»Test«) ist ja eine Zeichenkette, eben die von *GROSS* übergebene Zeichenkette »TEST«. Folgender Aufruf ist dagegen korrekt:

```
=SUMME(3; 6; LÄNGE("Test"))
```

Die Funktion *LÄNGE* erwartet als Argument eine Zeichenkette und liefert als Funktionswert eine Zahl, nämlich die Länge der Zeichenkette, in diesem Fall also 4.

Diese Zahl ist ein völlig korrektes Argument für *SUMME*, sodass dieser Aufruf die Zahlen 3, 6 und 4 addiert und als Ergebnis den resultierenden Funktionswert 13 als Feldinhalt anzeigt.

Übrigens könnten Sie der *LÄNGE*-Funktion als Argument auch einen Bezug übergeben, etwa in der Art $2 * \text{LÄNGE}(A1)$. Vorausgesetzt, A1 enthält eine Zeichenkette, denn dann ist ja die Bedingung, dass an *LÄNGE* ein String übergeben wird, erfüllt. Enthält A1 zum Beispiel den Text »Dies ist ein Test«, liefert *LÄNGE* als Funktionswert 17.

Manchmal sind Argumente optional: Sie können sie angeben, müssen es jedoch nicht.

*Optionale
Argumente*

Ein gutes Beispiel dafür ist wieder die *SUMME*-Funktion. Sie müssen ihr mindestens ein Argument übergeben, können jedoch auch weitere Argumente angeben.

Ein weiteres Beispiel ist die Funktion *FINDEN*:

```
=FINDEN(Suchtext; Text; Erstes_Zeichen)
```

Sie prüft, ob in der Zeichenkette »Text« eine zweite Zeichenkette »Suchtext« enthalten ist. Das optionale Argument »Erstes_Zeichen« bestimmt, ab dem wievielten Zeichen die Suche in »Text« beginnt. Zum Beispiel durchsucht

```
=FINDEN("u"; "Bauer"; 1)
```

die Zeichenkette »Bauer« ab dem ersten Zeichen »B« nach der gesuchten Zeichenkette (hier nur ein einzelnes Zeichen) »u«. Als Funktionswert liefert *FINDEN* die Zahl 3, die Position, an der die gesuchte Zeichenkette gefunden wurde (bzw. den Fehlerwert #WERT!, wenn sie nicht darin vorhanden ist).

Ein weiteres Beispiel: Der Aufruf

```
=FINDEN("a"; "Hans Maier"; 6)
```

liefert den Funktionswert 7: Da für das Argument *Beginn* der Wert 6 übergeben wurde, beginnt Excel das Absuchen von »Hans Maier« nach einem »a« erst beim sechsten Zeichen. Von da an gibt es erst ab Zeichen Nummer 7 wieder ein »a«.

Ein Argument, das wie *Beginn* in der Syntaxangabe kursiv ausgezeichnet ist, ist optional. Geben Sie es nicht an, verwendet Excel den meistverwendeten Standardwert, bei *FINDEN* beispielsweise eine 1 (= Durchsuchen des Textes ab dem ersten Zeichen).

Das heißt, die Funktionsaufrufe

```
=FINDEN("a"; "Hans Maier"; 1)
```

und

```
=FINDEN("a"; "Hans Maier")
```

sind absolut äquivalent. Beide suchen in der Zeichenkette »Hans Maier« ab dem ersten Zeichen nach einem »a«.

Eine Funktion wird somit genau von drei Merkmale gekennzeichnet:

- ➔ Anzahl und Art der übergebenen Funktionswerte
- ➔ Anzahl und Art des resultierenden Funktionswerts (der Funktionswerte)
- ➔ und natürlich die Art und Weise, in der die übergebenen Argumente behandelt werden

Einige Funktionen besitzen im Gegensatz zu *SUMME* oder *GROSS* mehrere Funktionswerte und zwar die Matrixfunktionen, die gesondert besprochen werden. Davon abgesehen liefert jedoch jede Funktion genau einen Funktionswert, unabhängig davon, wie viele Argumente ihr übergeben werden.

11.4 Funktionsgruppen

11.4.1 Mathematische Funktionen

Ein einfaches Beispiel für eine mathematische Funktion ist *PRODUKT*. Sie erwartet wie *SUMME* als Argument maximal 14 Zahlen (Konstanten, Bereiche, Bezüge auf Zellen, die Zahlen enthalten etc.). Ein Beispiel:

=PRODUKT(A1;15;B10:B15)

Diese Zahlen werden jedoch nicht addiert, sondern miteinander multipliziert.

Weitere interessante mathematische Funktionen sind *ABS*, *GANZZAHL* und *WURZEL*. Alle drei erwarten als Argument jeweils eine Zahl und liefern als Funktionswert ebenfalls eine Zahl.

ABS ermittelt den absoluten Wert einer Zahl, unabhängig von ihrem Vorzeichen: *ABS*(20) liefert den Funktionswert 20 und *ABS*(-20) ebenfalls das Ergebnis 20. *ABS* ersetzt somit einfach ein eventuell vorhandenes negatives Vorzeichen durch ein positives.

Die Funktion *GANZZAHL* kennt jeder Programmierer unter dem zugehörigen englischen Namen *INT*. Sie liefert den ganzzahligen Anteil einer Zahl.

Dies geschieht oft, indem einfach alle Nachkommastellen entfernt werden. Excel geht jedoch anders vor: Bei positiven Zahlen werden die Nachkommastellen entfernt, bei negativen Zahlen jedoch auf die nächstkleinere Zahl aufgerundet.

Allgemein: *GANZZAHL* liefert die größte ganze Zahl, die kleiner oder gleich dem Argument ist. Einige Beispiele dafür:

➔ *GANZZAHL*(10,34) ergibt 10

➔ *GANZZAHL*(5) ergibt 5

➔ *GANZZAHL*(-7,28) ergibt -8

WURZEL liefert die Quadratwurzel einer Zahl. Das heißt *WURZEL*(16) ergibt 4 und *WURZEL*(2) ergibt 1,41.

Sehr interessant ist auch *ZUFALLSZAHL*. Dieser Funktion werden keinerlei Argumente übergeben, dennoch liefert sie irgendeine Zufallszahl (genauer: Pseudozufallszahl) zwischen 0 und 1, genauer: eine Zahl, die größer oder gleich 0 und kleiner als 1 ist. 1 wird also niemals ganz erreicht.

Bei jeder Neuberechnung der Tabelle wird eine neue Zufallszahl generiert, eine praktische Sache beispielsweise für Lotto-Zahlen. Ich spiele zwar selbst nicht, aber soweit ich weiß, benötigt man dafür sechs Zahlen zwischen 1 und 49 (Abbildung 11.1).

Die Zellen B3 bis B8 enthalten alle die Formel

=GANZZAHL(ZUFALLSZAHL()*49)+1.

Abbildung 11.1:
Funktion.xls,
»Lotto«

=GANZZAHL(ZUFALLSZAHL()*49)+1

	A	B
	Lottozahlen generieren (Neue "Auslosung" mit F9)	
1		
2		
3	1.Zahl	28
4	2.Zahl	8
5	3.Zahl	3
6	4.Zahl	11
7	5.Zahl	32
8	6.Zahl	31

Sie multipliziert den von **ZUFALLSZAHL** gelieferten Wert mit 49, schneidet anschließend mit **GANZZAHL** alle Nachkommastellen ab und addiert zum Schluss eine 1.

ZUFALLSZAHL liefert als kleinste Zahl 0. Multipliziert mit 49 bleibt das immer noch 0 und auch nach dem Abschneiden der Nachkommastellen ändert sich nichts daran. Nach Addition der 1 ergibt sich daher als kleinstmögliches Resultat die niedrigste Lottozahl 1.

Umgekehrt liefert **GANZZAHL** höchstens eine Zahl, die minimal kleiner als 1 ist, zum Beispiel 0,9999. Multipliziert mit 49 wird daraus 48,9999. **GANZZAHL** macht daraus eine glatte 48 und nach Addition der 1 ergibt sich die größtmögliche Zufallszahl von 49.

Allgemein: Um eine ganzzahlige Zufallszahl in einem Bereich von MIN bis einschließlich MAX zu erzeugen, verwenden Sie die Formel

=GANZZAHL(ZUFALLSZAHL()*MAX)+MIN



Mit **[F9]** erhalten Sie eine neue Auslosung, da **ZUFALLSZAHL** bei jeder Neukalkulation des Arbeitsblatts neue Zufallszahlen liefert.

11.4.2 Textfunktionen

Textfunktionen (Stringfunktionen) operieren mit Texten, nicht mit Zahlen. Ein Beispiel dafür ist die Funktion **LINKS**, die aus einer Zeichenkette einen Teil heraustrennt:

=LINKS(Text;Anzahl_Zeichen)

Der Funktionswert von **LINKS** ist eine Zeichenkette, die die ersten *Anzahl_Zeichen* von *Text* enthält. Wird der optionale Parameter *Anzahl_Zeichen* nicht angegeben, übergibt **LINKS** die komplette Zeichenkette.

Nehmen wir an, A1 enthält irgendeinen Text. Sie benötigen in A2 eine Kopie dieses Textes, allerdings nicht in der vollen Länge, sondern um fünf Zeichen gekürzt. Also geben Sie in A2 folgende Formel ein:

```
=LINKS(A1;LÄNGE(A1)-5)
```

LÄNGE ermittelt die Länge der in A1 enthaltenen Zeichenkette, die – um fünf vermindert – der *LINKS*-Funktion als Anzahl der zu übergebenden Zeichen angegeben wird. Enthält A1 zum Beispiel den Text »Dies ist ein Test«, enthält A2 anschließend »Dies ist ein«.

Auf nur eine Zelle angewandt, erscheint das noch nicht unbedingt sinnvoll. Aber stellen Sie sich eine Tabelle vor, die A1 bis A12 Monatsnamen enthält, »Januar«, »Februar« bis »Dezember«.

Nehmen wir weiter an, ganz rechts ist als letzte Spalte J im Arbeitsfenster zu sehen. Sie benötigen auch in dieser Spalte die Monatsnamen, allerdings nur die jeweils ersten drei Buchstaben, da Sie die Breite dieser Spalte stark verringert haben und die Namen sonst nicht vollständig sichtbar werden. Also geben Sie in J1 die Formel

```
=LINKS(A1;3)
```

ein, die als Inhalt von J1 die ersten drei Buchstaben des in A1 enthaltenen Textes »Januar« definiert, und kopieren diese Formel nach J2 bis J12 – das ist einfacher als die erneute Eingabe per Hand, nicht wahr?

Wollen Sie die ersten drei Buchstaben groß darstellen (»JAN«, »FEB«, »MÄR« etc.) verwenden Sie in J1:J12 stattdessen die Formel

```
=GROSS(LINKS(A1;3))
```

Sie übergibt nicht nur die jeweils ersten drei Zeichen, sondern wandelt sie zusätzlich in Großbuchstaben um.

11.4.3 Datum- und Uhrzeitfunktionen

Sowohl ein Datum als auch eine Uhrzeit wird von Excel als serielle Zahl gespeichert, als Zahl, für die als Ausgangsbasis 1 benutzt wird und die ganzzahlig weitergezählt wird.

Serielle Zahlen

Der Ausgangsbasis 1 entspricht das Datum 1.1.1900. Entsprechend speichert Excel das Datum 2.1.1900 intern als eine 2, den 3.1.1900 als 3 und so weiter.

Geben Sie ein Datum in eine Zelle ein und formatieren diese danach mit einem echten Zahlenformat wie »0,00«, wird Excel statt des Datums die zugehörige serielle Zahl anzeigen.

Als Datum wird diese Zahl nur dargestellt, wenn Sie die betreffende Zelle mit dem Standard- oder einem der Datumsformate formatieren.

Ähnliches gilt für die Uhrzeit, für die ich im Folgenden das Format »hh:mm:ss« verwende. Für Excel ist eine Uhrzeit eine serielle Zahl im Bereich 0 bis 0,999. Dieser Bereich entspricht dem Zeitbereich von 00:00:00 Uhr bis 23:59:59 Uhr.

Zum Beispiel wird 01:00:00 als serielle Zahl 0,0416 dargestellt, eben als 1/24. Und 12:00:00 als 1/2, also 0,5.

Allgemein: Die Uhrzeit X entspricht der seriellen Zahl $1/(24/X)$.

Die Eingabe einer Zeit mit den Zusätzen »AM« oder »PM« ändert übrigens nichts an der zugehörigen seriellen Zahl: Die Eingabe »13:00:00« wird als serielle Zahl 0,5416 gespeichert. Und »1:00:00 PM« (Nachmittag) ebenfalls, da Excel die 1 gemäß der erwähnten Formel umwandelt und dank des Zusatzes »PM« einen halben Tag dazu addiert, also die serielle Zahl 0,5.

Beispiele für Datumsfunktionen sind die Funktionen von *DATUM*, die ein Datum in die zugehörige serielle Zahl umwandeln, *DATWERT*, die einen Datumstext in eine serielle Zahl wandelt, und *WOCHENTAG*, die aus einer seriellen Zahl, die ein Datum darstellt, den zugehörigen Wochentag ermittelt.

Datumsdifferenz

Nehmen wir an, Sie interessieren sich dafür, wie viele Tage zwischen dem 1.3.2008 und dem 21.6.2009 liegen. Jeder zusätzliche Tag erhöht eine serielle Datumszahl um 1. Also wandeln Sie beide Daten mit *DATUM* in serielle Zahlen um und subtrahieren sie voneinander.

DATUM besitzt die Syntax

=DATUM(Jahr;Monat;Tag)

Die Funktion erwartet drei Argumente: zuerst das Jahr, dann den Monat und zuletzt den Tag.

Für das Datum dürfen Sie wahlweise die Eingabe »2008« oder einfach »08« verwenden. Kommen Sie jedoch keinesfalls auf die Idee, den Monat im Klartext anzugeben (»Jun«). Die Funktion erwartet keinen Datumstext, sondern eine echte Zahl!

»Monat« und »Tag« können Sie bei überflüssigen Nullen einstellig eingeben, also »6« statt »06«.

Die benötigte Formel lautet somit

=DATUM(9;6;21)-DATUM(8;3;1)

Sie führt zur Anzeige 477, da die Differenz zwischen beiden seriellen Zahlen und damit auch die Differenz in Tagen 477 ist.

Häufiger als *DATUM* werden Sie wahrscheinlich *DATWERT* einsetzen. Diese Funktion besitzt zwar auch die Aufgabe, die zu einem Datum zugehörige serielle Zahl zu ermitteln, allerdings müssen Sie für das Datum nicht drei Argumente in recht ungewohnter Reihenfolge übergeben, sondern nur ein Argument, einen Datumstext in der üblicheren Form »TT.MM.JJ«.

Datumstext

Zum Beispiel ergibt der Aufruf

```
=DATWERT("1.1.2009")
```

die Zahl 39814. Dieses Datum entspricht somit dem 39.814. Tag ab dem Beginn der Excel-Zeitrechnung, dem 1.1.1900.

DATWERT können Sie hervorragend verwenden, um die Differenz zwischen zwei Datumsangaben zu ermitteln, die sich als *Datumstexte* in irgendwelchen Feldern befinden.

Zum Beispiel ermittelt

```
=DATWERT(A2)-DATWERT(A1)
```

die Differenz in Tagen zwischen den in A2 und A1 enthaltenen Datumsangaben. Enthält A2 »21.6.2009« und A1 den Text »1.3.2008«, ist das Resultat dieser Formel der Wert 477.

DATWERT kann ausschließlich auf Datumstexte angewendet werden! Wenn Sie ein Datum in der Form »1.1.2009« eingeben, speichert Excel intern jedoch keinen Text, sondern die zugehörige serielle Zahl!



!!
STOP

Excel speichert dieses Datum nur dann als echten Text und nicht als Datum, wenn Sie es in der folgenden Form eingeben:

```
= "1.1.2009"
```

Auf ein Datum, das als Text eingegeben wurde und somit für Excel eigentlich gar kein Datum, sondern einen Text wie »Hallo« oder »Test« darstellt, müssen Sie daher die Funktion *DATWERT* anwenden, die diesen Text als Datum interpretieren kann.

Wurde das Datum jedoch nicht als Text, sondern ganz normal eingegeben, müssen Sie *DATUM* verwenden, um die Differenz zwischen den Datumsangaben in zwei Zellen zu ermitteln.

Zusätzlich müssen Sie vor dem Aufruf dieser Funktion mit den Funktionen *JAHR*, *MONAT* und *TAG* die von *DATUM* benötigten drei Argumente ermitteln, zum Beispiel mit der Formel

```
=DATUM(JAHR(A2);MONAT(A2);TAG(A2))-DATUM(JAHR(A1);MONAT(A1);TAG(A1))
```

Wochentage Eine weitere nützliche Datumsfunktion ist *WOCHENTAG*. Sie verwenden sie am besten gemeinsam mit *DATUM* oder *DATWERT*.

Ausgangspunkt für die Ermittlung eines Wochentags ist normalerweise ein bestimmtes Datum, zum Beispiel der 22.6.2009. *WOCHENTAG* erwartet als Argument jedoch kein Datum, sondern die zugehörige serielle Zahl. Also übergeben Sie das Datum zunächst der Funktion *DATUM* als Argument, die die zugehörige serielle Zahl liefert. Das Ergebnis übergeben Sie wiederum an *WOCHENTAG*:

```
=WOCHENTAG(DATUM(2009;6;22))
```

Diese Formel ergibt als Resultat 2. Da Excel die Tage von 1 bis 7 durchnummeriert, wobei der 1 der Sonntag und der 7 der Samstag entspricht, ist der 22.6.2009 offenbar ein Montag.

Bei Datumstexten verwenden Sie *DATWERT*, um das Argument für *WOCHENTAG* zu ermitteln:

```
=WOCHENTAG(DATWERT("22.6.09"))
```

Ein schönes Beispiel für die Anwendung der Zeitfunktionen liefert *ZEIT*. Diese Funktion erwartet als Argument eine Uhrzeit, zum Beispiel 14:42:00.

Die Syntax:

```
=ZEIT(Stunde;Minute;Sekunde)
```

Zeitdifferenz Als Funktionswert liefert *ZEIT* die zugehörige serielle Zahl. Die Angabe *Sekunde* ist optional. Entfällt sie, verwendet Excel dafür 0.

Um die Differenz zwischen zwei Uhrzeiten herauszubekommen, wandeln Sie beide mit *ZEIT* in serielle Zahlen um und subtrahieren sie voneinander. Multiplizieren Sie das Ergebnis mit 24, dann erhalten Sie die Differenz in Stunden. Zum Beispiel ergibt

```
=24*(ZEIT(14;12;00)-ZEIT(13;10;00))
```

als Resultat 1,03. Zwischen den beiden Uhrzeiten liegen also 1,03 Stunden.

Befände sich die Uhrzeit 14:12:00 in A1 und die Zeit 13:10:00 in A2, könnten Sie die Differenz noch einfacher ermitteln. Mit $24 * (A1-A2)$ subtrahieren Sie einfach die beiden seriellen Zahlen und multiplizieren die Differenz wieder mit 24.

Es gibt nicht nur ganze, sondern auch ungerade serielle Zahlen, nämlich Kombinationen aus einem Datum und einer Uhrzeit (siehe Kapitel 6.4, »Zahlen-, Datums- und Zeitformate«).

*Gemischte serielle
Zahlen*

Excel speichert diese beide Informationen in einer kombinierten seriellen Zahl der Art 12345,678. Dabei entspricht der Teil vor dem Komma dem Datum und der Nachkommateil der seriellen Zahl der Uhrzeit.

Um diese interne Speicherung müssen Sie sich jedoch nicht kümmern, da uns die entsprechenden Datums- und Zeitfunktionen die Auftrennung dieser Informationen abnehmen.

11.4.4 Statistikfunktionen

Einfachere statistische Funktionen sind auch für Statistiklaien interessant, zum Beispiel **ANZAHL**:

`=ANZAHL(Wert1;Wert2;...)`

Diese Funktion liefert die Anzahl der angegebenen Werte. Zum Beispiel ergibt

`=ANZAHL(A1:A10;B1:B5)`

das Resultat 15.

MAX ermittelt die größte in der angegebenen Werteliste enthaltene Zahl und **MIN** entsprechend die kleinste:

`=MAX(Zahl1;Zahl2;...)`

MITTELWERT kennen Sie alle. Diese Funktion liefert den arithmetischen Mittelwert der aufgeführten Zahlen:

`=MITTELWERT(Zahl1;Zahl2;...)`

Zum Beispiel liefert der Ausdruck **MITTELWERT(A1:A5)** den Funktionswert 3, wenn A1 bis A5 der Reihe nach die Zahlen 1 bis 5 enthalten.

Für Profis interessanter sind Grundfunktionen der deskriptiven Statistik zur Ermittlung der Standardabweichung oder der Varianz.

Trendprognosen

In der Praxis oft nützlich ist die Funktion *TREND*, die mehrfaktorielle lineare Mini-Regressionsanalysen durchführt:

=TREND(y_Werte;x_Werte;Neue_x_Werte;Konstante)

Eine Regression ermittelt durch Interpolation die Beziehung zwischen einer unabhängigen und einer abhängigen Variablen.

Ein einfaches Beispiel: Sie kennen Ihr Einkommen in den ersten sechs Monaten dieses Jahres. Die unabhängige Variable ist in diesem Fall der Monat, Januar, Februar bis Juni. Davon abhängig ist die Variable »Einkommen«. *TREND* versucht die Beziehung zwischen beiden Variablen durch eine lineare Funktion der Form

$$y = m * x + b$$

zu beschreiben. Mit dem Resultat, dass Sie dieser Funktion anschließend neue Werte der unabhängigen Variablen übergeben können, zum Beispiel die Monate Juli, August und September, und als Funktionswerte das voraussichtliche Einkommen in diesen Monaten erhalten.

Einigermaßen zuverlässig sind diese Prognosen jedoch nur, wenn dem Problem eine lineare Regression auch tatsächlich angemessen ist, wenn Ihr Einkommen also wirklich von Monat zu Monat um einen halbwegs konstanten Betrag steigt (oder etwa fällt?).

Besteht zwischen den Monaten und dem zugehörigen Einkommen kein linearer Zusammenhang, müssen Sie andere Funktionen verwenden, zum Beispiel *VARIATION* oder *RKP*, die von einem exponentiellen Zusammenhang ausgehen.

Bis auf *Konstante* sind alle Argumente von *TREND* Matrizen. Als *y_Werte* können Sie *TREND* beispielsweise eine Matrixkonstante mit den Einkommenswerten während der sechs Monate übergeben (zum Beispiel {5386;5800;5974;5864;6012;6753}) oder aber einen Bereich, der diese Werte enthält (A3:F3).

Für *x_Werte*, also die unabhängige Variable, können Sie natürlich keine Texte wie »Januar« oder »Februar« verwenden – es muss sich schon um Zahlen handeln.

In Fällen wie diesem bietet es sich an, die Monatsnamen umzukodieren und durch die Zahlen 1 bis 6 zu ersetzen, also die Matrixkonstante {1;2;3;4;5;6} anzugeben, oder einen Bereich, der diese Zahlen enthält.

Neue_x_Werte sind die Werte der unabhängigen Variablen, für die Sie eine Prognose wünschen, in unserem Fall also die Monate Juli, August und Sep-

tember. Beziehungsweise – in einer *TREND*-gerechten Form – die Matrixkonstante {7;8;9}.

In der Praxis sieht die Vorbereitung dieser Regression so aus (Abbildung 11.2):

	A	B	C	D	E	F	G	H	I
1	Monatliche Einnahmen								
2	1	2	3	4	5	6	7	8	9
3	5386	5800	5974	5864	6012	6753			
4									
5									
6									
7									
8									
9	Monatliche Ausgaben								
10	1	2	3	4	5	6	7	8	9
11	2831	2695	2964	3625	3275	3386			
12									
13									
14									
15									
16									
17	Monatlicher Überschuß								
18	2555	3105	3010	2239	2737	3367	0	0	0
19									

Abbildung 11.2:
Funktion.xls,
»Trend«

In Zeile 3 ist das bekannte Einkommen in den Monaten Januar bis Juni angeordnet, also die bekannten Werte der abhängigen Variablen.

Zeile 2 enthält die zugehörigen Monatsnummern, also sowohl die bekannten Werte der unabhängigen Variablen (A2:F2) als auch die neuen Werte (G2:I2), für die darunter die zugehörigen Werte der abhängigen Variablen »Einkommen« ermittelt werden sollen.

TREND liefert als Funktionswert eine Matrix! Sie besitzt den gleichen Umfang wie die Matrix *Neue_x_Werte* und ist in unserem Fall somit eine 3x1-Matrix (drei Werte sollen prognostiziert werden).

Entsprechend müssen Sie den Aufruf von *TREND* als Matrixformel (Eingabe mit **Strg** + **↕** + **↵** beenden!) in jenen zuvor selektierten Bereich eingeben, der die Resultate aufnehmen soll, also in G3:I3:

```
=TREND(A3:F3;A2:F2;G2:I2)
```

Die Matrix A3:F3 enthält die bekannten Werte der abhängigen Variablen »Einkommen«, A2:F2 die zugehörigen Werte der unabhängigen Variablen »Monat« und G2:I2 die interessierenden Werte dieser Variablen.

Im markierten Bereich G3:I3 soll *TREND* die zugehörigen prognostizierten Werte der abhängigen Variablen anzeigen.

Damit es etwas interessanter wird, erstellt die Tabelle zusätzlich eine Prognose für die Ausgaben in den drei Monaten Juli bis September (Codes 7 bis 9), basierend auf den bekannten Werten von Januar bis Juni. Die entsprechende Matrixformel lautet

```
=TREND(A11:F11:A11:F11;G11:I11)
```

Weiter unten werden die Differenzen zwischen dem jeweiligen Einkommen und den zugehörigen Ausgaben ermittelt. Geben Sie entsprechend in A18 die Formel

```
=A3-A11
```

ein und kopieren Sie diese Formel nach B18:I18.

Die resultierende Trendanalyse sehen Sie hier (Abbildung 11.3):

Abbildung 11.3:
Funktion.xls,
»Trendres«

	A	B	C	D	E	F	G	H	I
1	Monatliche Einnahmen								
2	1	2	3	4	5	6	7	8	9
3	5386	5800	5974	5864	6012	6753	6700,9	6911,2	7121,6
4									
5									
6									
7									
8									
9	Monatliche Ausgaben								
10	1	2	3	4	5	6	7	8	9
11	2831	2695	2964	3625	3275	3386	3646,9	3794,8	3942,7
12									
13									
14									
15									
16									
17	Monatlicher Überschuß								
18	2555	3105	3010	2239	2737	3367	3054	3116,4	3178,9
19									

Soviel zu *TREND*, abgesehen von einigen Erläuterungen bezüglich der optionalen Angaben *x_Werte*, *Neue_x_Werte* und *Konstante*.

Geben Sie *Neue_x_Werte* nicht an, so berechnet Excel den Trend für die bekannten Werte: Sie erhalten keine Prognose, sondern eine Glättung der bereits bekannten Werte, entsprechend der erstellten Trendgeraden.

Entfernen Sie zum Beispiel in *TREND2* aus der ersten Matrixformel die letzte Angabe G2:I2, wird G3 statt wie zuvor 6700,9 nun den Wert 5439 enthalten, H3 den Wert 5649,4 und I3 den Wert 5859,7.

Statt wie zuvor einer Prognose für die Folgemonate Juli bis September entsprechen diese drei Werte nun einer Glättung der Monate Januar bis März.

Zur Interpretation: Regression bedeutet, dass Excel die von Ihnen angegebenen Werte in ein gedachtes Koordinatensystem einträgt. Dabei entspricht die Monatsnummer der X- und die Einkommenshöhe der Y-Achse. Die sechs Monats-/Einkommenspaare entsprechen in diesem Koordinatensystem sechs X-/Y-Wertepaaren, also sechs einzelnen Punkten.

Bei der Regression versucht Excel nun eine Gerade so durch diese Punkte zu legen, dass der Abstand der Geraden von jedem Punkt möglichst klein ist.

Tatsächlich ist bei der linearen Regression jedoch praktisch immer eine Abweichung der Geraden von den Punkten gegeben, die Punkte streuen um die Gerade – außer, Ihr Einkommen würde tatsächlich Monat für Monat um einen konstanten Betrag steigen oder fallen, was wohl kaum der Fall ist.

Mit den rückwirkenden Prognosen liefert Excel nicht wie zuvor Prognosen, sondern zeigt Ihnen stattdessen in G3:I3, wo die Punkte für die Monate Januar bis März auf der Geraden tatsächlich lägen, wenn das der Fall wäre. Zum Beispiel hätte Ihr Einkommen im Januar eben nicht 6.700 DM betragen, sondern nur 5.439 DM.

Wie stark diese rückwirkend prognostizierten Werte von den tatsächlichen abweichen, ermöglicht Ihnen eine Rückschluss auf die Güte der Regression, darauf, wie gut oder schlecht sich Ihre Daten überhaupt durch eine lineare Regression vorhersagen lassen.

Sind die Abweichungen groß, ist der Zusammenhang zwischen der unabhängigen Variablen »Monat« und der abhängigen Variablen »Einkommen« eben nicht linear und kann daher auch nicht zuverlässig durch eine lineare Regression erfasst werden.

Sie müssten sich nun überlegen, wie denn der Zusammenhang tatsächlich ist und entsprechend zum Beispiel eine quadratische oder logarithmische Regression auf die Daten loslassen.

Übrigens könnte für die rückwirkende Prognose auch das Argument *Bekannte_x_Werte* entfallen, da Excel dann als Standardwert eine Matrixkonstante der Form {1;2;3;...} verwenden würde, die ja exakt unseren Monatsnummern entspräche.

Das optionale Argument *Konstante* ist ein Wahrheitswert, Sie setzen also an der betreffenden Stelle in den Funktionsaufruf entweder *WAHR* oder *FALSCH* ein.

WAHR führt dazu, dass Excel den konstanten Ausdruck b der linearen Funktion

$$y = m * x + b$$

schätzt. Mit *FALSCH* wird stattdessen für diesen Ausdruck der Wert 0 verwendet.

**Mehrfaktorielle
Regression**

Zwei Tipps zum Schluss: Excel kann auch eine mehrfaktorielle lineare Regression durchführen, die benötigt wird, wenn eine abhängige Variable wie »Einkommen« nicht nur von einer unabhängigen Variablen abhängt, sondern von mehreren, zum Beispiel von der aktuellen Monatsnummer und der Betriebszugehörigkeit.

Dazu ordnen Sie einfach die verschiedenen unabhängigen Variablen parallel neben- oder untereinander an und geben als Argument *Bekannte_x_Werte* statt einer eindimensionalen eine mehrdimensionale Matrix an.

Eng mit *TREND* verwandt sind die Funktionen *VARIATION*, *RGP* und *RKP*. *VARIATION* geht jedoch von einem exponentiellen Zusammenhang zwischen den X- und den Y-Werten aus, führt also eine Exponentialregression durch, als deren Ergebnis ebenfalls eine Matrix aus Y-Werten übergeben wird.

RGP entspricht *TREND* und führt ebenfalls eine lineare Regression durch, übergibt jedoch statt einer Matrix aus Y-Werten die Werte, die für die Terme m und b der linearen Funktion ermittelt wurden.

Die gleiche Aufgabe besitzt *RKP*, allerdings nicht für eine lineare, sondern wie *VARIATION* für eine Exponentialfunktion.

11.4.5 Suchfunktionen

Fast alle Suchfunktionen wie *VERGLEICH*, *VERWEIS* oder *INDEX* durchsuchen einen Bereich nach einem Wert, der einem angegebenen Suchkriterium entspricht.

Diese Funktionen sind hervorragend geeignet, um Tabellen mit zusammengehörenden Werten zu durchsuchen. Zum Beispiel eine Steuertabelle mit dem zu versteuernden Einkommen und der jeweiligen Steuerschuld.

Eine solche Tabelle kann zweiseitig aufgebaut sein. In der linken Spalte befinden sich verschiedene Einkommensbeträge und in der Spalte rechts daneben die Höhe der zugehörigen Steuerschuld, etwa so:

Einkommen	Steuerschuld
10.000,00	2.200,00
12.500,00	2.700,00
15.000,00	3.300,00
...	
...	

Mit *VERGLEICH* können Sie Excel in dieser Tabelle die Höhe der Steuerschuld bei einem bestimmten Einkommen selbstständig ermitteln lassen:

VERGLEICH

=VERGLEICH(Suchkriterium;Suchmatrix;Vergleichstyp)

Suchkriterium ist der Wert, nach dem der Bereich *Suchmatrix* durchsucht wird.

In unserem Fall ist *Suchmatrix* die Spalte »Einkommen« und *Suchkriterium* der fragliche Einkommensbetrag, zum Beispiel 12.500,00 DM.

Findet *VERGLEICH* den gesuchten Betrag, gibt die Funktion als Resultat die Position – den Index – des Wertes innerhalb des durchsuchten Bereichs zurück, im Beispiel eine 2, da sich 12.500 an zweiter Stelle des durchsuchten Bereichs befindet.

Sie können anschließend nachschauen, welcher Betrag sich an der entsprechenden Position 2 in der Tabelle »Steuerschuld« befindet – oder gar Excel (siehe unten) mithilfe der Suchfunktion *INDEX* nachschauen lassen.

In der Einkommens-Tabelle sind natürlich nicht alle möglichen Einkommensbeträge enthalten, sondern zum Beispiel nur eine grobe Abstufung in 2.500-DM-Schritten (10.000 DM, 12.500 DM, 15.000 DM usw.).

VERGLEICH wird daher einen krummen Einkommensbetrag wie 13.417,74 DM mit Sicherheit nicht finden, sondern nur einen Wert, der leicht darunter, und einen zweiten, der ein wenig darüber liegt. Was dann passiert, darüber entscheidet das Argument *Vergleichstyp*, eine 1, -1 oder 0:

- 1: Excel nimmt den größten Wert der Tabelle, der noch kleiner oder gleich dem Suchkriterium ist.

Voraussetzung: die Tabelle muss in aufsteigender Reihenfolge geordnet sein.

- 1: Excel nimmt den kleinsten Wert der Tabelle, der größer oder gleich dem Suchkriterium ist.

Voraussetzung: die Tabelle muss in absteigender Reihenfolge geordnet sein.

- 0: Excel nimmt nur Werte, die dem Suchkriterium exakt entsprechen.

Geben Sie *Vergleichstyp* nicht an, verwendet Excel automatisch den Wert 1.

Enthält die Tabelle zum Beispiel als nächsten Betrag nach 12.500 DM die Zahl 15.000 DM und Sie suchen 13.417,74 DM, so übergibt *VERGLEICH* die Position des Betrags 12500 DM, der größten Zahl, die noch kleiner oder gleich 13.417,74 DM ist.

Im Beispiel wird die relative Positionsangabe 2 übergeben, da sich 12.500 an zweiter Stelle im Bereich befindet.

INDEX Diese relative Positionsangabe, diesen Index, können Sie benutzen, um mit der Funktion *INDEX* den zugehörigen Betrag in der Steuerschuldtabelle zu suchen:

=INDEX(Matrix;Zeilenindex;Spaltenindex)

INDEX übergibt den Wert, der sich an der Position *Zeilenindex/Spaltenindex* im Bereich *Matrix* befindet.

Zeilenindex und *Spaltenindex* beziehen sich nicht auf die Tabelle selbst, sondern sind immer relativ zum angegebenen Bereich gemeint, der sich über beliebig viele Zeilen und Spalten erstrecken kann.

Im Beispiel erstreckt sich der Bereich über mehrere Zeilen einer einzigen Spalte. Daher muss als Argument *Spaltenindex* auf jeden Fall eine 1 angegeben werden, damit die erste (und einzige) Spalte des Bereichs durchsucht wird.

Zeilenindex, die relative Zeilenposition innerhalb des Bereichs, wird uns von *VERGLEICH* übergeben. Damit ergibt sich folgende Kombination beider Funktionen (Abbildung 11.4).

Diese Abbildung zeigt die beiden Spalten »Betrag« und »Steuern«. In C19 befindet sich das im Beispiel gewählte Einkommen von 71.312,58 DM und in C20 eine relativ komplizierte Formel, die die zugehörige Steuerschuld ermittelt:

=INDEX(C2:C18;VERGLEICH(C19;A2:A18);1)

Werten Sie die Formel bitte wie Excel von innen nach außen aus: Zunächst wird *VERGLEICH* aufgerufen, um den Bereich A2:A18 (die Suchmatrix) nach dem kleinsten Betrag zu durchsuchen, der größer oder gleich jenem in C19 ist (das Suchkriterium).

VERGLEICH findet 70.000 DM und liefert als relative Position dieses Betrags im durchsuchten Bereich den Index 5, da 70.000 DM der fünfte Wert im Bereich A2:A18 ist.

```
=INDEX(C2:C18;VERGLEICH(C19;A2:A18);1)
```

	A	B	C
1	Betrag		Steuern
2	60000,00		14400,00
3	62500,00		15300,00
4	65000,00		16200,00
5	67500,00		17100,00
6	70000,00		18000,00
7	72500,00		19000,00
8	75000,00		20000,00
9	77500,00		21000,00
10	80000,00		22000,00
11	82500,00		23000,00
12	85000,00		24000,00
13	87500,00		25100,00
14	90000,00		26200,00
15	92500,00		27300,00
16	95000,00		28500,00
17	97500,00		29600,00
18	100000,00		30700,00
19	Zu verst. Einkommen		71312,58
20	Steuerschuld		18000,00
21			

Abbildung 11.4:

Funktion.xls,
»Indvergl«

Nun wird *INDEX* aufgerufen, um den Wert zu übergeben, der sich in der fünften Zeile (Argument *Zeilenindex*) der ersten Spalte (Argument *Spaltenindex*) des Bereichs C2:C18 (*Matrix*) befindet. Das Resultat sehen Sie in C20: die zugehörige Steuerschuld von 18.000,00 DM bei einem zu versteuernden Einkommen von 71.312,58 DM.

Ich will Ihnen noch eine weitere Suchfunktion vorstellen, die derartige Aufgaben erheblich vereinfacht, da sie eine Art Kombination aus *VERGLEICH* und *INDEX* darstellt:

SVERWEIS

```
=SVERWEIS(Suchkriterium;Matrix;Spaltenindex)
```

Lassen Sie sich nicht durch den Begriff *Matrix* verwirren. Es muss sich keineswegs um einen so genannten Mehrfachoperationsbereich handeln, der durch den Befehl DATEN|TABELLE... entstand!

Matrix ist einfach ein rechteckiger Bereich mit allen Tabellen, die beim betreffenden Problem von Bedeutung sind, in unserem Beispiel also A2:C18.

SVERWEIS sucht in der ersten Spalte dieses Bereichs nach *Suchkriterium* und liefert jenen Wert, der sich in der gleichen Zeile, aber in der Spalte *Spaltenindex* des Bereichs befindet.

Die Funktion soll nach dem in C19 enthaltenen Wert suchen (Suchkriterium) und zwar in der ersten Spalte des Bereichs A2:C18 (*Matrix*).

Anschließend soll sie jenen Wert übergeben, der sich in der gleichen Zeile, aber der dritten Spalte des Bereichs befindet (Spaltenindex). Damit ergibt sich die Formel

```
=SVERWEIS(C19;A2:C18;3)
```

Sie ersetzt die zuvor erläuterte deutlich komplexere Kombination aus *VERGLEICH* und *INDEX* vollständig.

SVERWEIS wird übrigens kein Argument *Vergleichstyp* übergeben. Die Funktion sucht immer den kleinsten Wert, der *in einer aufsteigend sortierten Tabelle* größer oder gleich dem Suchkriterium ist. Sie verhält sich also ebenso wie *VERGLEICH* bei Angabe des Vergleichstyps 1.

Sowohl *INDEX* als auch *SVERWEIS* können Tabellen mit wesentlich mehr als nur einer Spalte durchsuchen. Zum Beispiel könnte sich rechts neben der Spalte C, »Steuern«, in Spalte D eine weitere Tabelle »Spitzensteuersatz« befinden, die die zugehörigen Spitzensteuersätze enthält, jenen Prozentsatz, mit dem praktisch die letzte verdiente Mark versteuert wird.

Nehmen wir weiter an, C20 soll, ausgehend von einem bestimmten Einkommensbetrag in C19, nicht die zugehörige Steuerschuld, sondern den zugehörigen Spitzensteuersatz anzeigen, dann verwenden Sie die Formel

```
=SVERWEIS(C19;A2:D18;4)
```

Der gesamte Bereich mit allen Tabellen reicht nun von A2:D18. A2:A18 enthält die Tabelle »Betrag«, C2:C18 die zugehörigen »Steuern« und D2:D18 die entsprechenden Spitzensteuersätze.

Gesucht wird in der ersten Bereichsspalte nach dem Betrag, der sich in C19 befindet und ausgegeben wird der zugehörige Betrag in der vierten Spalte des Bereichs, der Tabelle mit den Spitzensteuersätzen.

WVERWEIS *WVERWEIS* verhält sich analog zu *SVERWEIS*, geht allerdings von einer zeilen- statt spaltenweisen Anordnung der Tabellen aus. Durchsucht wird die erste Bereichs*zeile* und geliefert wird der zugehörige Wert in der angegebenen Bereichs*spalte*.

Texte suchen Alle Suchfunktionen suchen und liefern nicht nur Zahlenwerte, sondern können auch mit anderen Datentypen verwendet werden.

Zum Beispiel können Sie mit Texten verwendet werden! Die folgende Abbildung zeigt ein Arbeitsblatt, das ausgehend von einem bestimmten KFZ-Typ die zugehörigen Versicherungsprämien ermittelt (Abbildung 11.5).

=SVERWEIS(B13;A2:E10;3)

	A	B	C	D	E
1	KFZ-Typ		Haftpflicht	Teilkasko	Vollkasko
2	BMW 735i, 211 PS		1496,00	128,00	3975,00
3	BMW M3, 205 PS		1496,00	123,00	3864,00
4	Ferrari F40, 486 PS		1496,00	875,00	10746,00
5	Kadett GSI, 150 PS		1496,00	94,00	1365,00
6	Kadett, 60 PS		562,00	74,00	964,00
7	Porsche 911, 250 PS		1496,00	135,00	4374,00
8	Renault Alpine, 185 PS		1496,00	112,00	3726,00
9	Toyota Supra, 235 PS		1496,00	132,00	4284,00
10	VW Käfer, 44 PS		430,00	53,00	653,00
11					
12					
13	KFZ-Prämie	Toyota Supra, 235 PS			
14	Haftpflicht	1496,00			
15	Teilkasko	132,00			
16	Vollkasko	4284,00			
17					

Abbildung 11.5:

Funktion.xls,
»Prämie«

Der Bereich A2:E10 enthält alle vier benötigten Tabellen, den KFZ-Typ und die zugehörigen – ziemlich frei erfundenen – Prämiensätze in drei Versicherungssparten.

Die Anwendung dieses Blatts ist einfach: Sie tragen in B13 den Namen eines der neun Fahrzeuge *exakt* ein. Daraufhin erscheinen in den drei Feldern darunter die zugehörigen Prämiensätze.

Alle drei Zellen enthalten prinzipiell die gleiche Formel, die *SVERWEIS* aufruft, als Suchkriterium den in B13 enthaltenen Text und als Bereich A2:E10 angibt.

Nur werden einmal die Resultate der dritten, einmal der vierten und beim letzten Mal (Ermittlung der Vollkaskoprämie) der fünften Spalte dieses Bereichs entnommen.

Entsprechend lautet die Formel zur Ermittlung der Vollkaskoprämie

=SVERWEIS(B13;A2:E10;5)

Diese Formel entnimmt den gewünschten Wert der fünften Bereichsspalte und ermittelt damit die Vollkaskoprämie.

Achten Sie beim Suchen nach Texten mit *SVERWEIS* darauf, dass diese alphabetisch aufsteigend sortiert sind!

Statt Zahlen können Sie sich als Resultate natürlich ebenfalls Texte liefern lassen, die in den Spalten 3, 4 und 5 des Bereichs abgelegt sind.

Beim Suchen nach Texten können Sie zusätzlich eine sehr nette Eigenschaft von Excel benutzen, das ungenaue Suchen: Wissen Sie nicht mehr, ob der gesuchte Name »Maier« oder »Meier« lautet, verwenden Sie als Suchkriterium einfach »M*«.

Angenommen, Sie durchsuchen eine Tabelle mit *SVERWEIS*. Dann verwenden Sie nicht eine Formel wie

```
=SVERWEIS("Maier";A2:E10;3)
```

sondern besser

```
=SVERWEIS("M*";A2:E10;3)
```

Excel akzeptiert daraufhin den ersten Namen, der mit »M« beginnt, egal, wie er weitergeht.

Allerdings wird auf diese Weise eventuell ein »Müller« statt des gewünschten »Maier« oder »Meier« gefunden. Daher ist in diesem Fall eher das Ausmaskieren einzelner Zeichen mit »?« angebracht: Taucht in einem Suchstring das Zeichen »?« auf, akzeptiert Excel im untersuchten Text an dieser Position beliebige Zeichen.

Die Formel

```
=SVERWEIS("M?ier";A2:E10;3)
```

findet den ersten in der Tabelle enthaltenen »Maier« oder »Meier«, aber auf keinen Fall einen »Müller«.

Beide Methoden, das Abkürzen und das Maskieren, können Sie miteinander kombinieren. So akzeptiert die Formel

```
=SVERWEIS("M?ier*";A2:E10;3)
```

nicht nur »Meier« und »Maier«, sondern auch einen »Meierbach« oder »Maiersdorf«.

11.4.6 Logische Funktionen

Aussagen Alle logischen Funktionen basieren auf der Prüfung von Aussagen. Eine Aussage ist ein Vergleich zweier Daten vom gleichen Typ, zum Beispiel zweier Zahlen oder zweier Texte, der mithilfe eines Vergleichsoperators durchgeführt wird.

Ein solcher Vergleich besitzt folgende Form:

[Datum1] [Vergleichsoperator] [Datum2]

Excel kennt folgende Vergleichsoperatoren:

Operator	Funktion
=	prüft, ob <i>Datum1</i> mit <i>Datum2</i> identisch ist
>	prüft, ob <i>Datum1</i> größer als <i>Datum2</i> ist
<	prüft, ob <i>Datum1</i> kleiner als <i>Datum2</i> ist
>=	prüft, ob <i>Datum1</i> größer oder gleich <i>Datum2</i> ist
<=	prüft, ob <i>Datum1</i> kleiner oder gleich <i>Datum2</i> ist
<>	prüft, ob <i>Datum1</i> ungleich <i>Datum2</i> ist

Tabelle 11.1:
Vergleichs-
operatoren

Datum1 und *Datum2* müssen vom gleichen Datentyp sein.

Das Resultat des Vergleichs ist einer der beiden Wahrheitswerte *WAHR* und *FALSCH*, je nachdem, ob der Wert wahr oder falsch ist. Einige Beispiele:

- ➔ $1=5$ liefert den Wahrheitswert *FALSCH*, da die Aussage nicht zutrifft; 1 ist zweifellos nicht gleich 5.
- ➔ $2*10=100/5$ liefert *WAHR*, da $2*10$ (20) gleich $100/5$ (auch 20) ist.
- ➔ $30>40/2$ ergibt *WAHR*, da 30 größer als $40/2$ ist.
- ➔ »Hallo" <=10 ist ein dummer Vergleich, da man keinen Text mit einer Zahl vergleichen kann.
- ➔ »Maier">"Bauer« ergibt *WAHR*, da »Maier« – alphabetisch gesehen, da es sich um Zeichenketten handelt – tatsächlich größer ist als »Bauer«.

An den numerischen Vergleichen ist interessant, dass die verglichenen Daten beliebig komplex sein können.

Sie können eine einzige numerische Konstante in den Vergleich einsetzen, eine Berechnung wie $2*10$ mit mehreren Konstanten oder auch eine Mischung aus Konstanten und Bezügen. Beispielsweise prüft der Vergleich » $2*10>=5*A1$ «, ob 20 größer oder gleich dem Fünffachen des in A1 enthaltenen Werts ist.

Das fünfte Beispiel zeigt, dass Sie auch Zeichenketten miteinander vergleichen können. Ein solcher Vergleich verläuft alphabetisch. Excel prüft, welche der beiden Zeichenketten alphabetisch größer und welche kleiner ist.

Entscheidungen

Eingesetzt wird eine Aussagenprüfung immer als Teil der wichtigsten logischen Funktion, der *WENN*-Funktion. Sie ermöglicht Ihnen, Entscheidungen zu treffen. Sie können in einer Zelle einen bestimmten Wert ermitteln, wenn eine angegebene Bedingung erfüllt ist, und einen anderen Wert, wenn diese Bedingung nicht erfüllt ist:

`=WENN(Prüfung;Dann_Wert;Sonst_Wert)`

Ist die Aussage *Prüfung* wahr, übergibt *WENN* als Funktionswert *Dann_Wert*. Ist sie falsch, übergibt *WENN* den *Sonst_Wert*. *WENN* könnte man übersetzen mit: »wenn die Aussage wahr ist, tue dies, sonst jenes«.

Beide Werte können jeden beliebigen Typ besitzen. Es kann sich zum Beispiel um Zahlen oder um Texte handeln (und natürlich um Bezüge auf Zellen oder Bereiche, die Zahlen oder Texte enthalten).

Nehmen wir an, Sie geben in A1 folgende Formel ein:

`=WENN(5>10;"Hallo";"Noch mal Hallo")`

Prüfung ist die Aussage »5>10«, *Dann_Wert* die Zeichenkette »Hallo« und *Sonst_Wert* die Zeichenkette »Noch mal Hallo«.

Was wird Excel wohl in A1 anzeigen? Natürlich »Noch mal Hallo«. Denn die getestete Aussage ist falsch, 5 ist wohl kaum größer als 10. *WENN* übergibt daher *Sonst_Wert*, die Zeichenkette »Noch mal Hallo«.

Ändern Sie die Aussage dagegen so ab, dass sie wahr ist, zum Beispiel in »10>5« oder in »10<>5«, wird in A1 stattdessen »Hallo« angezeigt, der *Dann_Wert*.

Ein Beispiel: Nehmen wir eine Tabelle, die Ihre Einnahmen und Ausgaben während der vergangenen zwölf Monate addiert. Rein theoretisch könnten Sie irgendwo versehentlich eine Null zu viel eintippen und aus der Ausgabe 1.363 DM auf diese Weise 13.630 DM machen.

Mit der *WENN*-Funktion können Sie solche offensichtlich unsinnigen Eingaben zwar nicht verhindern, aber anschließend darauf hinweisen. Unsinnig könnte heißen, dass die Summe aller Einnahmen 100.000 DM übersteigt oder die Summe aller Ausgaben 80.000 DM (Abbildung 11.6).

```
=WENN(SUMME(C2:C13)<80000;SUMME(C2:C13);"Irgend 'was stimmt nicht!")
```

	A	B	C	D	E
1		Einnahmen	Ausgaben		
2	Jan	3267	8563		
3	Feb	3934	3056		
4	Mae	7361	5946		
5	Apr	9473	4527		
6	Mai	7265	4614		
7	Jun	2826	4382		
8	Jul	5628	4967		
9	Aug	6384	3826		
10	Sep	9837	48260		
11	Okt	6352	3967		
12	Nov	8462	3725		
13	Dez	8563	3715		
14	Summen	79352	Irgend 'was stimmt nicht!		
15					

Abbildung 11.6:
Funktion.xls,
»Wenn«

Summiert werden die Zahlen in den Bereichen B2:B13 und C2:C13 wie üblich mit der *SUMME*-Funktion. Während in B14 jedoch ganz normal die Summe aller Zahlen der Tabelle darüber ausgegeben wird, zeigt Excel in C14 »Irgendwas stimmt nicht!« an.

C14 enthält die Formel

```
=WENN(SUMME(C2:C13)<80000;SUMME(C2:C13);"Irgend 'was stimmt nicht!").
```

Sie ruft *WENN* mit *SUMME(C2:C13)<80000* als Argument *Wahrheitsprüfung* auf: Excel prüft, ob die Summe der Zahlen im Bereich C2:C13 kleiner als 80000 ist.

Ist das der Fall (die Prüfung ergibt *WAHR*), dann übergibt *WENN* den *Dann_Wert*, also das Resultat des Aufrufs *SUMME(C2:C13)*: In C14 erscheint die Summe der addierten Zahlen.

War die Aussage dagegen falsch, wird *Sonst_Wert* übergeben – der Wert der Formel ist die Zeichenkette »Irgendwas stimmt nicht!«.

Genau das passiert im Beispiel, da die Summe tatsächlich größer ist als 80.000 und zwar, weil in C10 versehentlich statt 4.826 eine Null zu viel und somit 48.260 eingetippt wurde.

Diesen Eingabefehler hätten Sie normalerweise sicher nicht sofort entdeckt. Dank der *WENN*-Funktion werden Sie jedoch unmissverständlich darauf hingewiesen.

Übrigens befindet sich in B14 die Formel

```
=WENN(SUMME(B2:B13)<100000;SUMME(B2:B13);"Irgend 'was stimmt nicht!")
```

Die zu addierenden Einnahmen werden nur akzeptiert, wenn die Summe kleiner als 10.0000 ist. Ist das nicht der Fall, erhalten Sie auch in diesem Fall eine entsprechende Warnung.

Sehr häufig wird die *WENN*-Funktion eingesetzt, wenn – abhängig von einer Bedingung – der Wert eines Feldes entweder durch Formel A oder aber durch Formel B gebildet wird.

Nehmen wir an, Sie wollen abschätzen, welchen Betrag Sie nächstes Jahr jeden Monat längerfristig anlegen können: Immer, wenn in einem Monat mehr als 500 DM übrig bleiben, wollen Sie die Hälfte davon nicht einfach auf dem Girokonto liegen lassen, sondern längerfristig investieren, zum Beispiel in Aktien; nur die Hälfte, damit auf dem Giro- oder Sparkonto jederzeit liquide Mittel für Notfälle zur Verfügung stehen.

Setzen wir voraus, Sie sind Angestellter und beziehen ein festes Gehalt, sagen wir 3.200 DM netto. Sie wissen, dass Sie jeden Monat feste Ausgaben in Höhe von etwa 1.800 DM haben, für Miete, Einkaufen, Kleidung, Essen gehen und so weiter.

Dazu kommen unregelmäßige Ausgaben wie jährlich fällige Versicherungsprämien, der bereits für den Juli gebuchte Jahresurlaub, die im März geplante Anschaffung eines neuen Autos und Ähnliches.

Steht all das bereits halbwegs fest, können Sie auch Ihre monatlichen Ausgaben grob schätzen und Excel für jeden Monat prüfen lassen, ob wahrscheinlich mehr als 500 DM übrig bleiben und, falls ja, wie viel die Hälfte dieses Betrags ist, den Sie langfristig anlegen wollen (Abbildung 11.7).

Spalte B enthält die monatlichen Einnahmen, also Ihr Gehalt, das jeweils 3.200 DM netto beträgt. Eventuell ist die Novemberzahl wegen des hinzukommenden Weihnachtsgelds zu ändern.

Spalte C enthält die konstanten Ausgaben und D die außergewöhnlichen, die meist auch eine außergewöhnliche Höhe besitzen.

Spalte E soll für jeden Monat das für längerfristige Investitionen verfügbare Kapital ausweisen. E2 enthält die Formel

```
=WENN(B2-(C2+D2)>500;(B2-(C2+D2))/2;0)
```

$B2-(C2+D2)$ entspricht den Einnahmen minus den Ausgaben (konstante und unregelmäßige), also dem monatlichen Überschuss.

=WENN(B2-(C2+D2)>500;(B2-(C2+D2))/2;0)

	A	B	C	D	E
		Einnahmen	Feste Ausgaben	Unregelm. Ausgaben	Investitionskapital
1					
2	Jan	3200	1800	570	415
3	Feb	3200	1800	0	700
4	Mär	3200	1800	22000	0
5	Apr	3200	1800	0	700
6	Mai	3200	1800	0	700
7	Jun	3200	1800	0	700
8	Jul	3200	1800	3400	0
9	Aug	3200	1800	0	700
10	Sep	3200	1800	0	700
11	Okt	3200	1800	800	300
12	Nov	3200	1800	0	700
13	Dez	3200	1800	0	700
14	Summen	38400	21600	26770	6315

Abbildung 11.7:

Funktion.xls,
»Wenn1«

Ist dieser Überschuss größer als 500 DM, ergibt sich der Wert der Formel durch den Ausdruck $(B2-(C2+D2))/2$, der die Hälfte dieses Betrags ermittelt. Sonst ist der Formelwert gleich 0 – in diesem Monat steht kein Investitionskapital zur Verfügung.

Diese Formel wurde nach E3 bis E13 kopiert und in E14 die Summe gebildet. Sie sehen, dass Ihnen übers Jahr immerhin gut 6.000 DM für Aktien oder Ähnliches zur Verfügung stehen.

Übrigens habe ich Ihnen gleich die komplizierte Form der WENN-Funktion gezeigt. Das Argument *Sonst_Wert* ist nämlich optional. Geben Sie es nicht an, liefert WENN einfach den Funktionswert *FALSCH*, wenn die Aussage nicht wahr ist.

Entfernen Sie im Beispiel aus den Formeln in E2:E13 das letzte Argument 0, gibt Excel statt einer 0 in der betreffenden Zelle den Text *FALSCH* aus, wenn die Aussage nicht wahr ist – was allerdings nicht sehr schön aussieht.

Oft genügt es für eine Entscheidung nicht, nur eine Bedingung zu prüfen, zum Beispiel, wenn Sie entscheiden sollen, ob es Sommer ist. Die Aussage »wenn die Sonne scheint, ist es Sommer« ist nicht unbedingt richtig. Auch im Winter scheint die Sonne ab und zu.

Besser wäre eine Aussage, die zwei miteinander verknüpfte Bedingungen enthält: »Wenn die Sonne scheint und es warm ist, ist es Sommer.«

*Aussagen-
verknüpfung*

Dazu werden zwei Teilaussagen mithilfe des logischen Operators UND zu einer Gesamtaussage verknüpft, die nur dann wahr ist, wenn *beide* Teilaussagen wahr sind.

UND Derartige Verknüpfungen von Aussagen sind mithilfe der *UND*-Funktion möglich:

=UND(Wahrheitswert1;Wahrheitswert2;...)

Wahrheitswert ist eine beliebige Aussage. *UND* übergibt genau dann den Wahrheitswert *WAHR*, wenn beide Aussagen wahr sind. Ist eine oder sind beide falsch, ist auch die Gesamtaussage falsch und *UND* übergibt entsprechend *FALSCH*.

Angewendet wird *UND* innerhalb der *WENN*-Funktion, zum Beispiel so:

=WENN(UND(A1<100;A2<100);"Beide kleiner 100";"Nicht beide kleiner 100")

Sind beide Aussagen wahr, ist also sowohl der Inhalt von A1 als auch der von A2 kleiner als 100, liefert *UND* den Funktionswert *WAHR*. Die in der *WENN*-Funktion geprüfte Aussage ist dann wahr und der Funktionswert entsprechend der Text »Beide kleiner 100«.

Enthält dagegen auch nur eine der beiden Zellen eine Zahl, die größer ist als 100, übergibt *UND* den Funktionswert *FALSCH* und die *WENN*-Funktion entsprechend »Nicht beide kleiner 100«.

In der Praxis können Sie zum Beispiel bei einer Versuchsreihe prüfen, wie viele Testergebnisse größer als 40 und gleichzeitig kleiner als 60 sind. Nehmen wir an, die Testwerte sind spaltenweise in A1, A2, A3 und so fort untereinander angeordnet. Dann verwenden Sie in B1 die Formel

=WENN(UND(A1>40;A1<60);1;0).

Ist der Wert in A1 größer als 40 und kleiner als 60, wird in B1 eine 1 als Ergebnis dieser Formel eingetragen, sonst eine 0.

Die Formel kopieren Sie nach B2, B3 etc. Jedesmal erscheint eine 1, wenn der Wert links daneben die beiden Bedingungen erfüllt, sonst eine 0. Nun müssen Sie nur noch mit *SUMME* die Einsen addieren und wissen, wie viele Testwerte sich im Bereich zwischen 40 und 60 bewegen.

UND können Sie bis zu 14 Aussagen übergeben. Nur wenn alle wahr sind, liefert *UND* für die Gesamtaussage *WAHR*.

ODER Die Funktion *ODER* übergibt entsprechend *WAHR*, wenn *wenigstens* eine von ebenfalls bis zu 14 Aussagen wahr ist. Weitere logische Funktionen finden Sie im Anhang.

11.4.7 Informationsfunktionen

Alle Funktionen, deren Name mit *IST...* beginnt, prüfen, ob eine Zelle einen bestimmten Datentyp enthält. Falls ja, liefern sie den Funktionswert *WAHR*, sonst *FALSCH*.

ISTTEXT liefert *WAHR*, wenn *Wert* ein Text ist:

=ISTTEXT(Wert)

ISTTEXT(»Hallo«) liefert demnach auf alle Fälle *WAHR*, *ISTTEXT*(A1) nur dann, wenn A1 einen Text enthält.

Analog dazu prüft *ISTZAHL*, ob der angegebene Wert oder Bezug eine Zahl ist, und *ISTLEER*, ob es sich um eine leere Zelle handelt.

11.4.8 Finanzmathematische Funktionen

Die finanzmathematischen Funktionen drehen sich um die Finanzierung von Darlehen oder Hypotheken, den Ertrag einer Anleihe, eines Sparvertrags etc. Die verschiedenen Funktionen erfordern die Übergabe folgender Argumente:

- ➔ **Bw (Barwert):** Angenommen, Sie leisten oder erhalten eine Reihe von monatlichen Zahlungen. Dann ist der Barwert die Summe dieser Zahlungen, zum Beispiel der Darlehensbetrag, den Sie monatlich in kleinen Raten tilgen.
- ➔ **Rmz (Regelmäßige Zahlung):** Das sind regelmäßig zu leistende oder erhaltene Zahlungen, zum Beispiel die monatlichen Raten zur Tilgung eines Kredits und der Kreditzinsen oder umgekehrt die regelmäßigen Zinszahlungen, die Sie aus einem Sparvertrag erhalten.
- ➔ **Zins (Zinssatz):** Zinssatz pro Zahlungsperiode
- ➔ **Zw (Zukünftiger Wert):** Das ist ein optionales Argument, für das Excel den Standardwert 0 verwendet. Es wird bei Aufgaben der Art »Wie viel muss ich monatlich zurücklegen, um in X Jahren Y DM anzusparen?« als Y angegeben, als Sparziel.
- ➔ **Zzr (Zahlungszeiträume):** Gesamtzahl der Zahlungszeiträume bei Jahreszahlungen

Zum Beispiel gibt es bei einem Darlehen mit einer Laufzeit von zehn Jahren und *monatlicher* Ratenzahlung insgesamt $10 \cdot 12 = 120$ Zahlungszeiträume. Bei den *jährlichen* Zinszahlungen aus einer fünf Jahre laufenden Anleihe gibt es entsprechend fünf Zahlungszeiträume.

Ausgangspunkt der finanzmathematischen Funktionen ist fast immer die gleiche Grundformel, die, nach den verschiedenen Variablen aufgelöst, den unterschiedlichen Funktionen entspricht.

Zum Beispiel ermittelt *ZINS* den gleichnamigen Parameter, in Abhängigkeit von den restlichen Variablen. *ZW* ermittelt ebenfalls den gleichnamigen Parameter, wiederum in Abhängigkeit von den anderen Variablen.

Gerade *ZW* ist für finanzmathematische Laien interessant, da sich diese Funktion eignet, um zum Beispiel das Endkapital zu ermitteln, das sich bei einmaliger Investition in eine Anlage und daraus resultierenden regelmäßigen Zinszahlungen ergibt.

Zum Beispiel aus dem Kauf einer Anleihe mit einem Zinssatz von 6,4% und einer Laufzeit von acht Jahren für insgesamt 15.000 DM:

=ZW(Zins;Zzr;Rmz;Bw;F)

In diesem Beispiel benötigen Sie nur die Angaben *Zins*, *Zzr* und *Bw*.

- ➔ *Zins* ist bei jährlichen Zinszahlungen genau ein Jahreszinssatz von 6,4%.
- ➔ *Zzr*, Zahlungszeiträume, gibt es bei einer Laufzeit von acht Jahren und jährlichen Zinszahlungen genau acht.
- ➔ *Bw* ist Ihr Investitionskapital von 15.000 DM, genauer: -15.000, da sonst ein negativer Endbetrag resultiert.
- ➔ Für *Rmz* geben Sie 0 an, da Sie im Gegensatz zu einem Darlehen keinerlei regelmäßige Zahlungen leisten.

Damit ergibt sich folgende Formel:

=ZW(6,4%;8;0;-15.000)

Sie ergibt als Endkapital nach Fälligkeit der Anleihe inklusive der Zinsen und Zinseszinsen 24.639 DM.

11.4.9 Trigonometrische Funktionen

Das beste Beispiel für die trigonometrischen Funktionen ist die allseits bekannte Sinus-Funktion:

=SIN(Zahl)

SIN liefert den Sinus des im Bogenmaß angegebenen Winkels. Eine Winkelangabe in Grad müssen Sie zuvor durch Multiplikation mit der Kreiszahl Pi (3,14...) und anschließender Division durch 180 ins Bogenmaß konvertieren.

Den ungefähren Wert von Pi müssen Sie nicht per Hand eingeben. Die Funktion *PI*, der keinerlei Argumente übergeben werden, liefert eine Annäherung an Pi. Zum Beispiel liefert der Ausdruck

```
=SIN(30*PI()/180)
```

das Resultat 0,5, den Sinus von 30 Grad.

11.4.10 Matrixfunktionen

Mit Matrixfunktionen können Sie zwei Matrizen miteinander multiplizieren, die Determinante oder Inverse einer Matrix bestimmen oder eine Matrix transponieren.

Als Beispiel verwende ich die Funktion *MMULT*, die zwei Matrizen miteinander multipliziert:

```
=MMULT(Matrix1;MATRIX2)
```

Matrix2 muss genauso viele Zeilen enthalten, wie *Matrix1* an Spalten enthält, und so viele Spalten, wie *Matrix1* an Zeilen enthält.

Zum Beispiel können Sie eine 2*5-Matrix mit einer 5*2-Matrix multiplizieren, was als Resultat eine 5*5-Matrix ergibt (Abbildung 11.8).

{=MMULT(A1:B5;D1:H2)}

	A	B	C	D	E	F	G	H
1	1	6		11	12	13	14	15
2	2	7		16	17	18	19	20
3	3	8						
4	4	9						
5	5	10						
6								
7								
8								
9								
10								
11								
12								
13								

Resultatmatrix				
107	114	121	128	135
134	143	152	161	170
161	172	183	194	205
188	201	214	227	240
215	230	245	260	275

Abbildung 11.8:
Funktion.xls,
»Matrix«

11.4.11 Datenbankfunktionen

Excel besitzt einige spezielle Datenbankfunktionen, die Sie auf beliebige Bereiche anwenden können, die Listenstruktur besitzen. Das heißt auf alle rechteckigen Bereiche mit Feldnamen in der ersten und Datensätzen in den folgenden Zeilen. Die Datenbankfunktionen führen mit diesen Bereichen statistische Auswertungen durch.

Alle Datenbankfunktionen besitzen Funktionsnamen der Art *DBFunktionsname*. Alle erwarten drei Argumente:

```
=DBFunktionsname(Datenbank;Datenbankfeld;Suchkriterien)
```

Datenbank ist der Sie interessierende Bereich inklusive der Feldüberschriften. Ist ein entsprechender Name definiert, können Sie statt des Bereichsbezugs alternativ auch diesen Namen als Argument einsetzen.

Eine Datenbankfunktion arbeitet immer mit genau einer Spalte des Datenbankbereichs. Diese Spalte geben Sie mit dem Argument *Feld* an. *Datenbankfeld* ist dabei nicht der Feldname, sondern die Feldnummer!

Suchkriterien ist ein Bereich, der Suchkriterien inklusive der Feldüberschriften enthält, bzw. ein zugehöriger Bereichsname.

Interessieren Sie sich nicht nur für ausgewählte Daten, sondern für die gesamte Spalte einer Datenbank, dann geben Sie als Suchkriterienbereich einen Bereich an, der unter der Zeile mit den Feldnamen nur leere Zellen enthält, also ein Suchkriterium, dem alle Datensätze genügen.

Stellvertretend für alle anderen erläutere ich nun den Umgang mit der Funktion *DBMAX*. Sie prüft, welcher unter jenen Sätzen, die die Suchkriterien erfüllen, im angegebenen Feld den größten Wert enthält (Abbildung 11.9).

Abbildung 11.9:
Funktion.xls,
»Datenbank«

Funktion.xls								
	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.09	0,00
3	Maier	Albert	Schwalmweg	67005	Ludwigshafen	0621/826125	21.01.09	0,00
4	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.02.09	123,56
5	Hohenfel	Christin	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.08.09	6254,00
6	Müller	Werner	Bielefelder W	10023	Berlin	0372/735322	30.11.09	573,70
7	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.10.09	0,00
8	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.09.09	875,64
9	Maier	Stefan	Schweriner W	20034	Hamburg	0326/83626	03.02.09	273,90
10	Böhm	Walter	Hans-Pinsel	80042	München	089/836395	18.02.09	0,00
11	Irlbeck	Gustav	Maiersbache	80017	München	089/726384	01.02.09	0,00
12	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.09	3864,00
13	Meier	Franz	Stendaler W	68003	Mannheim	0621/83643	17.11.09	1925,00
14								
15	Mahnung		Datum					
16	WAHR		11.12.09			Größter Fehlbetrag:		6254,00
17								

A16 enthält die Formel

```
=(($C$16-G2)>90
```

Dieses Suchkriterium findet alle Datensätze, bei denen das Rechnungsdatum mehr als 90 Tage gegenüber dem in C16 enthaltenen aktuellen Datum zurückliegt. Die Formel

=DBMAX(A2:H13;8;A15:A16)

in H16 ermittelt den größten Fehlbetrag (»Fehlbetrag« ist Spalte 8 der Datenbank), der sich bei den gefundenen Sätzen ergibt.

Teil 3 Datenver- waltung und Datenanalyse

Kapitel 12: Was-wäre-wenn-Analysen	395
Kapitel 13: Listen und Datenbanken	417
Kapitel 14: Microsoft Query – Zugriff auf externe Datenbanken	445
Kapitel 15: Konsolidieren	473
Kapitel 16: Pivot-Tabellen	485
Kapitel 17: Mehrfachoperationen	519
Kapitel 18: Matrixformeln	529
Kapitel 19: Datenexport und Datenimport	539

Excel besitzt unglaublich vielseitige Möglichkeiten zur Verwaltung und Analyse von Daten.

Datenanalyse meint vor allem das Durchspielen von Modellen, die Fragen beantworten wie »Ab welchem Rohölpreis kommen wir mit unserem Produkt in die Verlustzone, wenn wir den Preis nicht erhöhen?«.

Nachdem ich Excels Möglichkeiten auf diesem Gebiet erläutert habe, zeige ich Ihnen, wie Excel als Datenbank eingesetzt werden kann. Mit echten Datenbanken wie Access kann sich Excel zwar nicht ganz messen, aber immerhin recht komfortabel auch umfangreiche Mengen an Aktien oder Adressen verwalten, inklusive ausgefeilter Such- und Filtermöglichkeiten.

Und wenn das nicht reicht, greifen Sie zur Datenverwaltung einfach auf Query zurück, ein Zusatzprogramm im Lieferumfang von Excel, das eine echte relationale Datenbank darstellt und wunderbar mit Excel zusammenarbeitet.

Weiter geht es mit dem Konsolidieren, dem Zusammenfassen in mehreren Quellbereichen enthaltener Daten für Auswertungen, und mit Pivot-Tabellen, die Kreuz-Tabellen ähneln, jedoch eine weitaus komfortablere Datenanalyse ermöglichen, inklusive dem problemlosen nachträglichen Umstellen und Ändern der erstellten Analysen.

Danach bespreche ich Mehrfachoperationen, die einen oder mehrere Parameter in den von Ihnen angegebenen Schritten variieren und die Auswirkungen auf die interessierenden Formeln im Arbeitsblatt speichern. Damit verwandt sind die Matrixformeln, mit denen Sie sich häufig die wiederholte Eingabe bzw. das Kopieren mehrerer identischer Formeln ersparen können.

Anschließend erläutere ich die umfangreichen Möglichkeiten von Excel, um Daten einer Excel-Tabelle zu exportieren, um sie beispielsweise mit einer Textverarbeitung oder Dateiverwaltung weiterzuverarbeiten oder umgekehrt Daten eines Fremdprogramms in eine Excel-Tabelle zu importieren; und natürlich Excels Internet-Features, angefangen bei Hyperlinks bis hin zum Publizieren von Excel-Tabellen als Webseiten und zur Erstellung von Webabfragen, die Daten aus dem Internet in Excel-Tabellen importieren, beispielsweise die neuesten Aktienkurse.

12 Was-wäre-wenn-Analysen

Excel besitzt verschiedenste Features, die das Durchspielen von Modellen ungemein erleichtern und Fragen beantworten wie »Wie stark darf der Dollar noch steigen, bevor wir mit unserem Dollar-basierten Importgeschäft in die Verlustzone kommen?«.

All diese Features erläutere ich nun der Reihe nach, angefangen beim harmlosesten, der Zielwertsuche. Sie variiert den Inhalt einer bestimmten Zelle solange, bis eine andere angegebene Zelle den von Ihnen gewünschten Inhalt besitzt, den Zielwert.

Die Zielwertsuche kann den Inhalt einer Zelle solange variieren und die davon abhängigen Formeln neu durchrechnen, bis – vielleicht – die Zielzelle das gewünschte Ergebnis enthält.

In komplexen Anwendungen müssen jedoch oft mehrere Parameter gleichzeitig verändert werden, um ein bestimmtes Resultat zu erreichen, also die Inhalte mehrerer Zellen. Das ermöglicht der Solver, der sich auch für weit aus komplexere Modelle eignet. Sie können detailliert festlegen, welche Zellen innerhalb welcher Grenzwerte von Excel verändert werden dürfen, um das gesuchte Resultat zu erhalten. Zum Beispiel kann Excel dazu den Inhalt von B1 und C5 verändern, die Inhalte dieser Zellen dürfen jedoch niemals kleiner als 4% werden.

Der Szenario-Manager ermöglicht sehr detaillierte Analysen, bei denen von Ihnen angegebene Zellen (Zinssatz, Spitzensteuersatz etc.) in zuvor von Ihnen festgelegten Schritten verändert werden. Nach jeder vorprogrammierten Veränderung zeigt Ihnen der Szenario-Manager auf Tastendruck die jeweils resultierenden Auswirkungen.

Der Ansichten-Manager wiederum ermöglicht die Erzeugung unterschiedlicher Ansichten Ihrer Tabelle und das Umschalten zwischen diesen Ansichten auf Tastendruck. Dadurch können Sie beispielsweise blitzschnell zwischen zwei Ansichten umschalten, in der Ihre Tabelle mal mit, mal ohne Gitternetzlinien und Gliederung dargestellt wird.

Der Bericht-Manager ermöglicht Ihnen, beliebige Kombinationen aus zuvor definierten Szenarien und Ansichten auszuwählen und auf Knopfdruck auszudrucken.

12.1 Zielwertsuche

Die Zielwertsuche variiert den Inhalt einer angegebenen veränderbaren Zelle solange, bis eine angegebene Zielzelle den von Ihnen gewünschten Inhalt, den Zielwert, besitzt.

Die Zielwertsuche arbeitet iterativ und nähert sich dem gewünschten Ziel schrittweise, indem der zu variierende Parameter (der Inhalt der veränderbaren Zelle) zuerst in großen und dann in immer kleiner werdenden Schritten verändert wird.

Nach jeder Veränderung wird die gesamte Tabelle neu berechnet und der Inhalt der Zielzelle mit dem gewünschten Zielwert verglichen. Abhängig vom Ergebnis erfolgt dann die nächste Variation der veränderbaren Zelle.

Bei komplexen Tabellen kann diese iterative Annäherung recht lange dauern. Möglicherweise führt sie sogar zu gar keiner Lösung, weil das angegebene Ziel durch Variation der angegebenen Zelle überhaupt nicht erreichbar ist.

Angenommen, Sie kaufen einen amerikanischen Personalcomputer. In Deutschland kostet er 7.000 DM. Beim Direktimport aus Amerika dagegen 3.000 US \$, wobei jedoch noch 100 \$ Frachtkosten hinzukommen und bei der Einfuhr wiederum auf diese 3.100 US \$ weitere 5% Zoll und 16% Einfuhrumsatzsteuer aufgeschlagen werden.

Bei einem angenommenen Dollarkurs von 2,00 DM ergeben sich insgesamt 7.440 DM, der Import lohnt also kaum den hohen Aufwand. Nehmen wir jedoch weiter an, der Dollar fällt gerade und Sie wüssten gern, wie tief er noch fallen müsste, bis der Rechner höchstens 6.000 DM kostet (Abbildung 12.1).

Abbildung 12.1:
Ausgangs-Tabelle
für die Zielwert-
suche

16-007.xls		
	A	B
1	Preis in US\$	3000
2	Fracht in US\$	100
3	Dollarkurs	2
4	DM-Preis	6200
5	- Zoll (5%)	310
6	- MWST (16%)	992
7	Gesamt	7502
8		

Zielwertsuche	
Zielzelle:	B7
Zielwert:	6000
Veränderbare Zelle:	\$B\$3
OK Abbrechen	

EXTRAS|ZIELWERTSUCHE... öffnet das abgebildete Dialogfeld. Selektieren Sie als *Zielzelle* per Zeigen B7. Geben Sie darunter als *Zielwert* 6.000 ein. Selektieren Sie als *Veränderbare Zelle* B3, die den Dollarkurs enthält, der solange variiert werden soll, bis B7 das gewünschte Ergebnis 6.000 enthält, und wählen Sie OK (Abbildung 12.2).

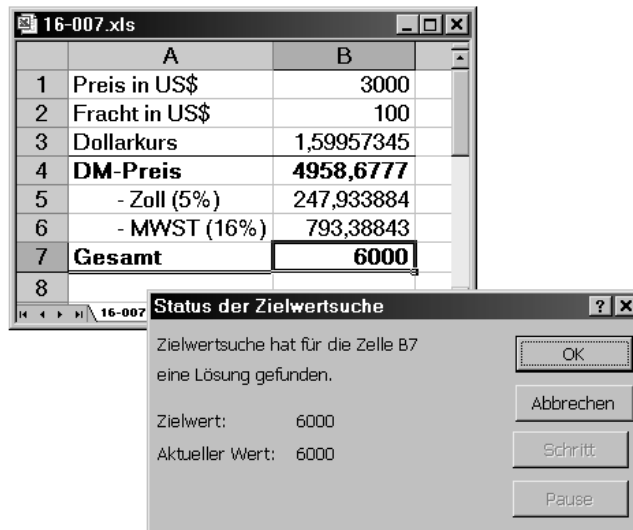


Abbildung 12.2:
Ergebnis der Zielwertsuche

Offensichtlich muss der Dollar bis auf rund 1,60 DM fallen (siehe Zelle B3), bevor sich der Import bei den von Ihnen gestellten Ansprüchen – maximal 6.000 DM – lohnt.

OK schließt das Dialogfeld unter Beibehaltung des ermittelten Werts für die geänderte Zelle. *Abbrechen* würde stattdessen den ursprünglichen Zustand wiederherstellen.

Die Schaltflächen *Pause* und *Schritt* sind nur in komplexen Arbeitsblättern interessant, bei denen die Zielwertsuche längere Zeit dauert.

Pause und Schritt

Mit *Pause* können Sie eine lange Suche unterbrechen und danach mit *Weiter* wieder aufnehmen oder aber die iterative Suche anschließend mit *Schritt* schrittweise fortsetzen lassen, um in aller Ruhe zu beobachten, wie sich Excel – hoffentlich – dem Ergebnis nähert.

12.2 Solver

Die Zielwertsuche kann den Inhalt einer einzigen Zelle solange variieren und die davon abhängigen Formeln neu durchrechnen, bis – vielleicht – die Zielzelle das gewünschte Ergebnis enthält.



In komplexen Anwendungen müssen jedoch oft mehrere Parameter gleichzeitig verändert werden, um ein bestimmtes Resultat zu erreichen. Das ermöglicht der mit EXTRAS|SOLVER... aufgerufene Solver (zur Einbindung siehe Anhang 6, »Add-Ins«) (Abbildung 12.3).

Abbildung 12.3:
Solver.xls, »Solver«

	A	B	C	D	E	F	G	H
1	Zinssatz	5%			Kapital	Zinsen	Zins n.St.	Endkapital
2	Steuersatz	40%		1. Jahr	10000	500	300	10300
3				2. Jahr	10300	515	309	10609
4				3. Jahr	10609	530	318	10927
5				4. Jahr	10927	546	328	11255
6				5. Jahr	11255	563	338	11593
7								

Solver-Parameter	
Zielzelle:	\$H\$6
Zielwert:	☒ Max ☐ Min ☐ Wert: 0
Veränderbare Zellen:	\$B\$1:\$B\$2
Nebenbedingungen:	
Schätzen Hinzu­fügen Ändern Löschen Lösen Schließen Optionen... Zurücksetzen Hilfe	

Die abgebildete Tabelle berechnet das Endkapital einer Anleihe mit einer Laufzeit von fünf Jahren bei Wiederanlage der jährlich ausgeschütteten Zinsen.

Das Endkapital ist von zwei Faktoren abhängig: vom Zinssatz und vom persönlichen Spitzensteuersatz, mit dem Sie Ihre Zinseinnahmen versteuern müssen.

F2 enthält die Formel $=E2 * \$B\$1$, multipliziert also das Startkapital von 10.000 DM mit dem in $\$B\$1$ eingetragenen Zinssatz von 5%, ermittelt also die Zinsen für das erste Jahr.

G2 enthält die Formel $=F2 - F2 * \$B\$2$ und berechnet die Zinseinnahmen nach Steuern, indem vom ausgeschütteten Zinsbetrag (F2) der Finanzamtsanteil abgezogen wird (Zinsbetrag mal persönlichem Spitzensteuersatz).

H2 addiert mit der Formel $=G2 + E2$ das Startkapital und die Zinseinnahmen nach Steuern und ermittelt somit das Endkapital nach einem Jahr.

Da dieser Betrag das Startkapital für das zweite Anlagejahr darstellt, enthält E3 die Formel $=H2$.

All diese Formeln wurden mehrfach kopiert, um die Resultate der folgenden vier Anlagejahre zu ermitteln. Dieses Kopieren machte übrigens die Verwendung absoluter Bezüge für die Zellen B1 mit dem Zinssatz und B2 mit dem Spitzensteuersatz notwendig, da diese Bezüge beim Kopieren keinesfalls angepasst werden dürfen!

Entscheidend ist, dass das in Spalte H nach einem, zwei, drei, vier oder fünf Jahren resultierende Endkapital vom Zinssatz und vom persönlichen Spitzensteuersatz, mit dem Sie Ihre Zinseinnahmen versteuern müssen, abhängt.

Ihr Spitzensteuersatz ist variabel, da Ihr Einkommen während der nächsten Jahre wohl kaum absolut konstant bleiben wird. Wahrscheinlich wird er sich irgendwo zwischen 30% und 50% bewegen.

Nehmen wir zusätzlich an, dass die Anleihe nicht mit einem festen, sondern mit einem variablen Zinssatz ausgestattet ist, der sich am aktuellen Kapitalmarktzins orientiert (so genannter Floater) und der sich Ihrer Ansicht nach in den nächsten Jahren zwischen 4% und 6% bewegen.

Aufgrund dieser beiden variablen Faktoren ist es keineswegs sicher, dass sich wie in der Abbildung nach fünf Jahren ein Endkapital von 11.593 DM ergibt. Stattdessen ist nur eine Aussage folgender Art zulässig: »Das Endkapital wird zwischen X DM und Y DM liegen«.

X und Y können Sie bestimmen, indem Sie für B1 und B2 einmal die niedrigsten möglichen Werte einsetzen (Zinssatz 4%, Spitzensteuersatz 30%) und ein zweites Mal die höchsten vorstellbaren Werte (Zinssatz 6%, Spitzensteuersatz 50%).

Wie die Abbildung zeigt, wird zunächst die Zielzelle angegeben, die das maßgebliche Resultat enthält, im Beispiel H6 mit dem Endkapital nach fünf Jahren.

Zielzelle

Unter dieser Zelle geben Sie an, ob der maximale, der minimale oder aber ein ganz bestimmter Wert dieser Zelle das zu erreichende Ziel ist. In unserem Fall interessieren die Extremwerte, die die Zelle annehmen kann, daher selektierte ich zunächst die Option *Max*.

Zielwert

Anschließend legen Sie die Zellen B1 und B2 fest, deren Wert Excel verändern soll, um das gewünschte Ziel zu erreichen.

Veränderbare Zellen

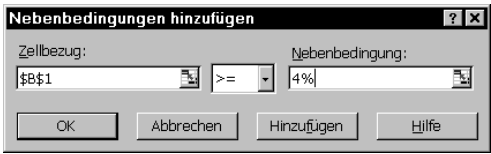
Im Gegensatz zur Zielwertsuche dürfen das beliebig viele sein. Die einzelnen Bezüge werden durch je ein Semikolon voneinander getrennt (nacheinander im Zeigemodus bei gedrückter **[Strg]**-Taste anklicken).

Das Listenfeld *Nebenbedingung* dient der Festlegung von Bedingungen, die Excel bei der Variation der unabhängigen Parameter einhalten muss.

Nebenbedingungen

Zum Beispiel wäre ein Zinssatz von 0% oder 43% realitätsfern, ebenso wie ein Spitzensteuersatz von 70%. Also geben Sie Excel bekannt, innerhalb welcher Grenzen diese beiden Parameter variiert werden dürfen. Zunächst wählen Sie *Hinzufügen* (Abbildung 12.4).

Abbildung 12.4:
Nebenbedingungen
festlegen



Unter *Zellbezug* geben Sie einen Bezug auf jene Zelle ein, für die Sie einen Grenzwert festlegen wollen, unter *Nebenbedingung* den Grenzwert selbst und in der Mitte suchen Sie sich eine von mehreren alternativen Bedingungen aus.

Wie die Abbildung zeigt, habe ich als Bezug den Zinssatz B1 angegeben. Die Bedingung »>=« zusammen mit dem Wert 4% besagt: »Der Inhalt von B1 muss größer oder gleich 4% sein«, darf also niemals kleiner als 4% werden.

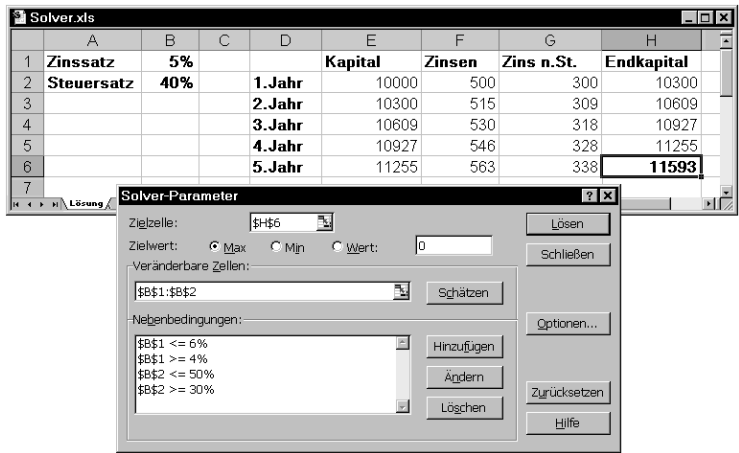


Unter Nebenbedingung können Sie übrigens nicht nur Konstanten, sondern auch Zellbezüge angeben. Ersetzen Sie zum Beispiel den Wert 4% durch den Bezug \$A\$10, so lautet die Bedingung: »Der Inhalt von B1 muss größer oder gleich dem in A10 enthaltenen Wert sein.«

Hinzufügen übernimmt die Bedingung und löscht die Eingabefelder, sodass Sie gleich anschließend eine weitere Nebenbedingung eingeben können.

Die Eingabe der letzten Nebenbedingung schließen Sie mit *OK* statt mit *Hinzufügen* ab. Daraufhin erscheint wieder das übergeordnete Dialogfeld (Abbildung 12.5).

Abbildung 12.5:
Vor dem Start



Die vier von mir definierten Nebenbedingungen besagen, dass B1 kleiner gleich 6% (0,06) und größer gleich 4% (0,04) sein muss und B2 kleiner gleich 50% (0,5) und größer gleich 30% (0,3).

Dadurch wird sichergestellt, dass Excel den Zinssatz nur innerhalb der Grenzen 4% bis 6% und den Spitzensteuersatz in den Grenzen 30% bis 50% variiert.

Leere Nebenbedingungen werden übrigens nicht akzeptiert! Schließen Sie die Eingabe der letzten Nebenbedingung mit Hinzufügen statt mit OK ab (die Eingabefelder sind danach wieder leer) und wählen Sie erst danach OK, so erscheint eine Fehlermeldung. Brechen Sie das Hinzufügen der leeren Nebenbedingung einfach mit Abbrechen ab.

!!
STOP

Nebenbedingungen können im übergeordneten Dialogfeld nachträglich editiert werden: Die Schaltfläche *Ändern* ermöglicht die nachträgliche Änderung der im Listenfeld selektierten Nebenbedingungen, *Löschen* löscht sie und *Zurücksetzen* löscht alle Einstellungen, die Sie im Solver-Dialogfeld getroffen haben, Zellbezüge und Nebenbedingungen.

Lösen startet die iterative Variation der beiden Zellen. Anschließend teilt Ihnen Excel mit, dass es entweder keine Lösung fand, mit der sich für die Zielzelle das gewünschte Resultat ergibt, oder aber wie hier, dass eine Lösung gefunden wurde.

Solver starten

Andere Fehlermeldungen deuten darauf hin, dass die individuell einstellbaren Lösungsoptionen falsch gewählt sind. In einem solchen Fall sollten Sie auf die Schaltfläche Optionen... klicken und diese Optionen entsprechend Bild 12.11 einstellen.

!!
STOP

Wurde eine Lösung gefunden, haben Sie nun die Wahl unter folgenden Alternativen (Abbildung 12.6):

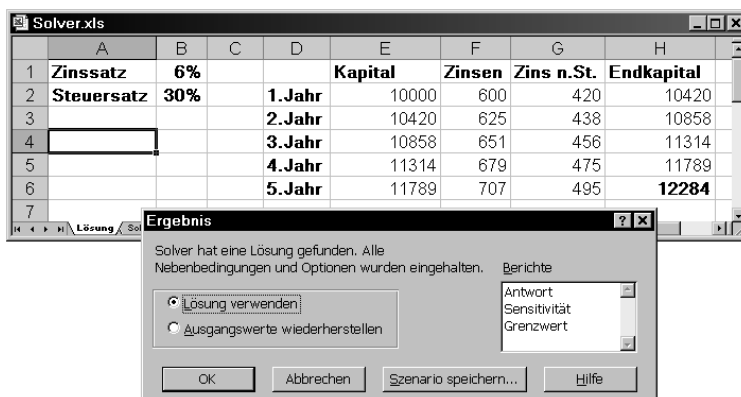


Abbildung 12.6:
Solver.xls, »Lösung«

Wie die Abbildung zeigt, besteht die Lösung in einem Zinssatz von 6% und einem Steuersatz von 30%. Beide Werte zusammen führen zum – bei Einhaltung der vier Nebenbedingungen – höchstmöglichen Endkapital von 12.284 DM.

Sie können nun entweder mit *Ausgangswerte wiederherstellen* die ursprünglichen Werte wiederherstellen oder aber mit *Lösung verwenden* die gefundene Lösung beibehalten.

Szenarien

Wollen Sie die Ausgangswerte wiederherstellen, sollten Sie zuvor mit *Szenario speichern...* den aktuellen Tabellenzustand unter einem frei wählbaren Namen als Szenario speichern (Abbildung 12.7).

Abbildung 12.7:
Als Szenario
speichern



Später können Sie jederzeit den Szenario-Manager aufrufen (siehe Kapitel 12.3, »Szenario-Manager«) und sich mit einem Doppelklick auf das in die Szenarienliste eingefügte »Szenario1« diesen Tabellenzustand und damit die vom Solver gefundene Lösung noch einmal anschauen.

Berichte

Im Listenfeld *Berichte* können Sie mit einem Doppelklick auf einen der Einträge einen der folgenden Berichte erstellen lassen: »Antwort« zeigt die Details der Lösung. »Grenzen« zeigt, welche Resultate sich bei den Grenzwerten der Nebenbedingungen ergeben. »Sensitivität« informiert darüber, wie sensibel die Lösung auf kleine Änderungen in der Zielzelle oder den Nebenbedingungen reagiert.

Sie können eine einzelne oder aber durch Mehrfachselektion zwei bzw. alle drei Auswertungsmöglichkeiten gleichzeitig nutzen.

Antwortbericht

Für jeden Bericht legt Excel ein eigenes Arbeitsblatt an und trägt darin die Analysen ein. Der Antwortbericht sieht zum Beispiel aus wie Abbildung 12.8.

Er enthält alle relevanten Informationen zur gefundenen Lösung: den Bezug der Zielzelle, ihren ursprünglichen und den neuen Inhalt und ihren Namen – sofern in den angrenzenden Zellen eine als Name verwendbare Zeichenkette enthalten ist.

Ebenso detaillierte Informationen erhalten Sie über die variierten Zellen.

Microsoft Excel 8.0 Antwortbericht
Tabelle: [Solver.xls]Lösung
Bericht erstellt am: 06.11.09 20:59:37

Zielzelle (Max)

Zelle	Name	Ausgangswert	Lösungswert
\$H\$6	5. Jahr Endkapital	12284	12284

Veränderbare Zellen

Zelle	Name	Ausgangswert	Lösungswert
\$B\$1	Zinssatz	6%	6%
\$B\$2	Steuersatz	30%	30%

Nebenbedingungen

Zelle	Name	Zellwert	Formel	Status	Differenz
\$B\$1	Zinssatz	6%	\$B\$1>=0.04	Nicht einschränkend	2%
\$B\$1	Zinssatz	6%	\$B\$1<=0.06	Einschränkend	0
\$B\$2	Steuersatz	30%	\$B\$2>=0.3	Einschränkend	0%
\$B\$2	Steuersatz	30%	\$B\$2<=0.5	Nicht einschränkend	0,2

Abbildung 12.8:
Antwortbericht

Zu den Nebenbedingungen gibt Excel an, auf welche Zellen sie sich beziehen, welche Werte diese Zellen momentan besitzen und ob der angegebene Grenzwert erreicht ist (»Einschränkend«) oder ob er die Lösung nicht einschränkte (»Nicht einschränkend«).

Der Grenzenwertbericht erläutert, welche Resultate sich bei den angegebenen Grenzwerten ergeben (Abbildung 12.9).

Grenzenwert-
bericht

Microsoft Excel 8.0 Grenzenwertbericht
Tabelle: [Solver.xls]Lösung
Bericht erstellt am: 06.11.09 21:00:37

Zielzelle

Zelle	Name	Endwert
\$H\$6	5. Jahr Endkapital	12284

Veränderbare Zellen

Zelle	Name	Endwert	Untere Grenze	Zielzelle Ergebnis	Obere Grenze	Zielzelle Ergebnis
\$B\$1	Zinssatz	6%	4%	1148063%	6%	1228397%
\$B\$2	Steuersatz	30%	30%	1228397%	50%	1159274%

Abbildung 12.9:
Grenzenwertbericht

Außer den aktuellen Werten von B1 und B2 zeigt Excel in den Zellen rechts daneben, welche Grenzwerte für diese Zellen gelten und welche Inhalte sich mit ihnen für die Zielzelle H6 ergeben.

Sensitivitäts- bericht

Mit dem Sensitivitätsbericht müssen Sie selbst zurechtkommen, da ich Ihnen mangels entsprechender Spezialkenntnisse keine fundierte Auskunft über Sinn und Zweck eines »reduzierten Gradienten« geben kann (Abbildung 12.10).

Abbildung 12.10:
Sensitivitätsbericht

Microsoft Excel 8.0 Sensitivitätsbericht				
Tabelle: [Solver.xls]Lösung				
Bericht erstellt am: 06.11.09 21:00:34				
Veränderbare Zellen				
		Lösung	Reduzierter	
	Zelle	Name	Endwert	Gradient
9	\$B\$1	Zinssatz	6%	4126091%
10	\$B\$2	Steuersatz	30%	-353665%
Nebenbedingungen				
13	Keine			

Schätzen

Die Option *Schätzen* des ursprünglichen *Solver*-Dialogfelds (siehe Bild 12.5) versucht selbstständig herauszufinden, welche Faktoren (Zellen) für den Inhalt der Zielzelle verantwortlich sind, und fügt die Bezüge dieser Zellen *selbstständig* unter *Veränderbare Zellen* ein!

In unserem Beispiel würde *Schätzen* zusätzlich zu \$B\$1 und \$B\$2 einen dritten Bezug einfügen, \$E\$2, also auf die Zelle mit dem Startkapital, was zweifellos sinnvoll ist, da das nach fünf Jahren resultierende Endkapital außer vom Zinssatz und vom Steuersatz ja auch vom Startkapital abhängt. Excel erkennt die variablen Faktoren in diesem Fall also besser als wir!

Was zunächst wie Zauberei aussieht, ist relativ einfach zu erklären: Excel verfolgt alle Abhängigkeiten der Zielzelle immer weiter rückwärts, bis es Zellen findet, deren Inhalt von keinen anderen Zellen mehr abhängt.

Die Zielzelle H6 enthält die Formel $=G6+E6$, ist also unter anderem von E6 abhängig. E6 enthält die Formel $=H5$, ist also selbst wiederum von H5 abhängig. H5 ist von E5 abhängig, E5 von H4 und so weiter.

Im vorletzten Schritt stellt Excel fest, dass H2 von E2 abhängt, von jener Zelle, die das Startkapital 10.000 enthält. Da diese Zelle keine Formel enthält und ihr Inhalt daher nicht selbst wiederum von einer anderen Zelle abhängt, stellt sie einen bestimmenden Faktor für den Inhalt der Zielzelle H6 dar. Damit erklärt sich der von Excel zusätzlich eingefügte Bezug \$E\$2.

Optionen... öffnet ein weiteres Dialogfeld (Abbildung 12.11).

Optionen

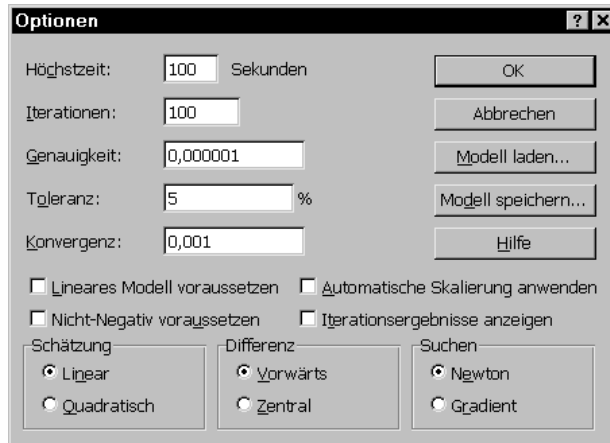


Abbildung 12.11:
Weitere Optionen

Sie können angeben, auf wie viele Iterationen maximal die Problemlösung beschränkt werden soll, wie genau das Resultat ausfallen muss, wie lange höchstens nach einer Lösung gesucht wird und so weiter:

- ➔ *Höchstzeit:* Die maximale Lösungszeit in Sekunden vor Abbruch der Suche
- ➔ *Iterationen:* Die maximale Anzahl an Iterationen vor dem Abbruch der Suche
- ➔ *Genauigkeit:* Genauigkeit, mit der Zielzelle und Bedingung übereinstimmen müssen, damit eine Nebenbedingung als erfüllt angesehen wird
- ➔ *Toleranz:* Speziell für ganzzahlige Lösungen erlaubte Abweichung
- ➔ *Lineares Modell voraussetzen:* Ist dieses Kontrollkästchen aktiviert, wird die Berechnung beschleunigt, wenn das Modell *ausschließlich* lineare Beziehungen enthält (was in unserem Beispiel der Fall ist). Es darf jedoch keinesfalls nicht lineare Beziehungen wie zum Beispiel quadratische oder logarithmische enthalten!
- ➔ *Iterationsergebnisse anzeigen:* Unterbricht die Berechnung nach jeder Iteration und zeigt an, wie weit sich der aktuelle Wert der Zielzelle bereits dem gewünschten Ergebnis genähert hat
- ➔ *Automatische Skalierung anwenden:* Schaltet die automatische Skalierung ein
- ➔ *Schätzung:* *Linear* geht zunächst von linearen Abhängigkeiten zwischen den unabhängigen Faktoren (den zu variierenden Zellen) und der abhängigen Variable (der Zielzelle) aus. *Quadratisch:* nimmt quadrati-

sche Beziehungen an, wodurch die Berechnungen beschleunigt werden, wenn das tatsächlich der Fall ist und Sie es von vornherein wissen.

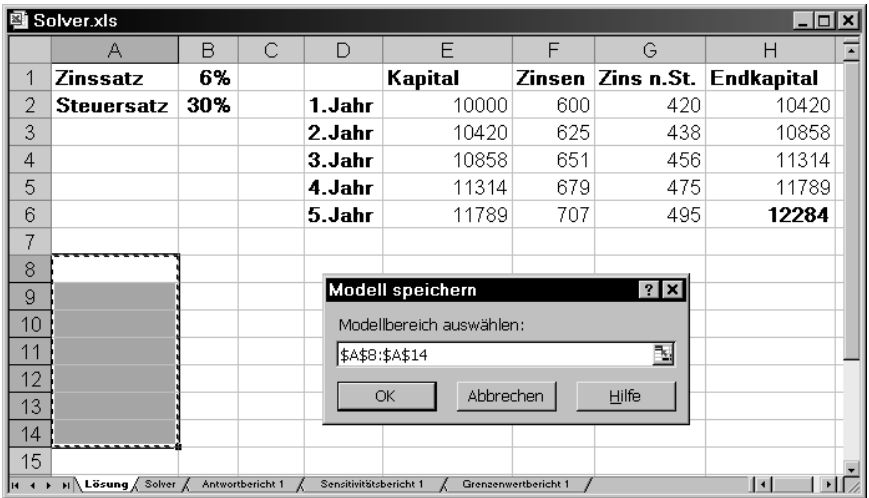
- ➔ *Differenzen*: Legt die Ableitungsmethode für partielle Ableitungen fest: *Zentral* ist die langsamere Methode, kann dafür jedoch manchmal auch dann zu Lösungen verhelfen, wenn mit der Standardmethode *Vorwärts* keine Lösung gefunden wird.
- ➔ *Suchen*: Definiert die verwendete Suchmethode; *Gradient* kann helfen, wenn die Vorgabe *Newton* nur geringe Fortschritte zwischen den einzelnen Iterationsschritten ergibt.

Problemlösungsmodelle Die Schaltflächen *Modell speichern...* und *Modell laden...* ermöglichen es, verschiedene Problemlösungsmodelle im Arbeitsblatt zu speichern und nach Belieben wieder zu laden. Mit »Modell« sind dabei alle Angaben im *Ausgangs*-Dialogfeld gemeint, die Festlegung der Zielzelle, der zu variierenden Zellen und der Nebenbedingungen.

Modell speichern... selektiert in der Umgebung der vor Anwahl dieses Befehls aktiven Zelle einen Bereich, in dem diese Angaben gespeichert werden.

Ein Dialogfeld mit einem Bereichsvorschlag erscheint, der wie folgt aussieht, wenn vor Aufruf des Solvers A8 selektiert war (Abbildung 12.12):

Abbildung 12.12:
Vorgeschlagener
Modellbereich



Der Solver schlägt den Bereich A8:A14 vor, benötigt also offensichtlich sieben Zellen, um alle Angaben zu speichern. Sie können stattdessen einen beliebigen anderen Bereich selektieren, der allerdings nicht kleiner sein sollte. OK speichert die Modellbeschreibung darin (Abbildung 12.13).

The screenshot shows the Excel Solver dialog box for a file named 'Solver.xls'. The 'Ziellabelle' (Target Cell) is set to '\$H\$6'. The 'Zu variierende Zellen' (Changing Variable Cells) are '\$B\$1:\$B\$2'. The 'Nebenbedingungen' (Constraints) are listed as '\$B\$1 >= 0,04', '\$B\$1 <= 0,06', '\$B\$2 >= 0,3', and '\$B\$2 <= 0,5'. The 'Iterationen bis' (Iterations) is set to 100. The 'Modell laden...' (Load Model...) button is highlighted. The background shows a spreadsheet with the following data:

	A	B	C	D	E	F	G	H
1	Zinssatz	6%			Kapital	Zinsen	Zins n.St.	Endkapital
2	Steuersatz	30%		1.Jahr	10000	600	420	10420
3				2.Jahr	10420	625	438	10858
4				3.Jahr	10858	651	456	11314
5				4.Jahr	11314	679	475	11789
6				5.Jahr	11789	707	495	12284
7								
8								12284
9								2
10								WAHR
11								WAHR
12								WAHR
13								WAHR
14								100
15								

Abbildung 12.13:
Problemlösungs-
modelle speichern
und laden

Im Beispiel werden sieben Zellen benötigt, um alle Angaben zu speichern, die anschließend folgende Formeln enthalten:

A8 : =MAX(\$H\$6)
A9 : =ANZAHL(\$B\$1:\$B\$2)
A10: = \$B\$1 >= 0,04
A11: = \$B\$1 <= 0,06
A12: = \$B\$2 >= 0,3
A13: = \$B\$2 <= 0,5
A14: = {100.100.0.000001.0.05.FALSCH.FALSCH.FALSCH.1.1.1}

Sie enthalten alle Informationen über die von Ihnen getroffenen Problemlösungsangaben.

Modell laden... lädt diese Informationen wieder in das *Solver*-Dialogfeld.

Um das auszuprobieren, sollten Sie zuvor mit *Zurücksetzen* die Standardeinstellungen wiederherstellen. Wenn Sie anschließend A8:A14 selektieren und unter *Optionen...* die Schaltfläche *Modell laden...* aktivieren, werden folgende in diesem Bereich abgelegte Einstellungen übernommen:

Ziellzelle: \$H\$6
Zu variierende Zellen: \$B\$1 und \$B\$2
Nebenbedingungen: \$B\$1 >= 0,04, \$B\$1 <= 0,06, \$B\$2 >= 0,3 und \$B\$2 <= 0,5
Und die in A14 definierten Iterationsbedingungen

Beide Schaltflächen zusammen ermöglichen Ihnen, in der Tabelle beliebig viele unterschiedliche Lösungsmodelle zu speichern und nach Bedarf zu benutzen, ohne jedesmal alle Angaben per Hand eingeben zu müssen.

12.3 Szenario-Manager

Tabellenkalkulationen eignen sich hervorragend für Analysen der Art »Was wäre wenn...«.

Zum Beispiel: »Wie teuer wäre mein Darlehen, wenn ich den veranschlagten Zinssatz von 10% um ein viertel oder gar ein halbes Prozent herunterhandeln könnte?«

Oder: »Wie würde sich eine Erhöhung des Spitzensteuersatzes um 3, 4 oder 5 Prozent auf mein Einkommen nach Steuern auswirken?«

All diese Analysen erfordern normalerweise, einen Parameter (Zinssatz, Spitzensteuersatz) Schritt für Schritt zu ändern und sich die Auswirkungen auf die benötigten Zahlen anzuschauen.

Analysen dieser Art vereinfacht der Szenario-Manager: Er variiert einen oder mehrere Parameter in den von Ihnen angegebenen Schritten und zeigt Ihnen auf Tastendruck die jeweils resultierenden Auswirkungen.

Ein gutes Beispiel dafür ist die folgende Tabelle, die im vorangegangenen Kapitel erläutert wurde und die sich auch in der Mappe ZINSEN.XLS befindet (Abbildung 12.14).

Abbildung 12.14:
Zinsen.xls,
»Tabelle 1«

	A	B	C	D	E	F	G	H
1	Zinssatz	5%			Kapital	Zinsen	Zins n.St.	Endkapital
2	Steuersatz	40%		1. Jahr	10000	500	300	10300
3				2. Jahr	10300	515	309	10609
4				3. Jahr	10609	530	318	10927
5				4. Jahr	10927	546	328	11255
6				5. Jahr	11255	563	338	11593
7								



**Szenario
hinzufügen**

Entscheidend ist, dass das in Spalte H nach einem, zwei, drei, vier oder fünf Jahren resultierende Endkapital von zwei Faktoren abhängt: vom Zinssatz und vom persönlichen Spitzensteuersatz, mit dem Sie Ihre Zinseinnahmen versteuern müssen (siehe Kapitel 12.2, »Solver«).

Sicher wäre es interessant, welche Auswirkungen in dieser Zinsezinsberechnung ein geänderter Zins- und/oder Spitzensteuersatz auf das nach fünf Jahren resultierende Endkapital hat.

Mit dem Szenario-Manager können Sie die interessierenden Änderungen durchspielen und sich die resultierenden Auswirkungen auf Tastendruck zeigen lassen.

Wählen Sie EXTRAS|SZENARIO-MANAGER. Ein Dialogfeld mit einer Liste aller momentan definierten Szenarien erscheint (siehe Bild 12.19), die momentan jedoch leer ist. Aktivieren Sie die Schaltfläche *Hinzufügen...*, um das folgende Dialogfeld zu öffnen (Abbildung 12.15).

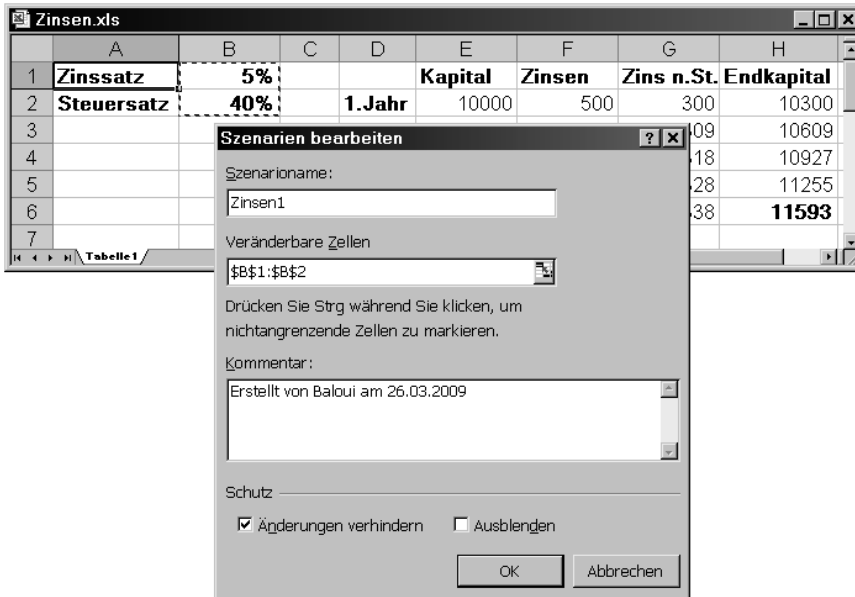


Abbildung 12.15:
Szenario hinzufügen

Geben Sie dem Szenario einen Namen wie »Zinsen1«. Selektieren Sie anschließend die zu verändernden Zellen, im Beispiel die Zellen B1 und B2: Klicken Sie B1 an und danach – bei gedrückter **[Strg]**-Taste – B2.

Szenarienwerte

Das Eingabefeld enthält anschließend die Zellreferenzen in absoluter Form, getrennt durch ein Semikolon. Auf die gleiche Weise – natürlich auch durch Eintippen – können Sie beliebig viele Zellreferenzen eingeben.

Unter *Kommentar* können Sie einen Kommentar zum Zweck dieses Szenarios eingeben.

Aktivieren Sie *Änderungen verhindern*, dann kann dieses Szenario später nicht von anderen Mitgliedern einer Arbeitsgruppe verändert werden.

Aktivieren Sie *Ausblenden*, so wird es in der Szenarienliste (siehe Abbildung 12.19) nicht angezeigt.

Da all das momentan uninteressant ist, wählen Sie nun einfach OK, um dieses Szenario und somit eine erste Modellvariante zu definieren (Abbildung 12.16).

Abbildung 12.16:
Erste Modellvariante
definieren

Nun müssen Sie in den Eingabefeldern die in diesem Szenario für B1 und B2 einzusetzenden Werte eingeben.

Excel schlägt die aktuellen Zellinhalte vor, gibt jedoch statt des Zellinhaltes »5%« die gleichwertige Form »0,05« und statt »40%« entsprechend »0,4« vor.

Für das erste Szenario sollten Sie diesen Vorschlag, der mit den Originalwerten arbeitet, übernehmen.

Danach aktivieren Sie *Hinzufügen* und das *Szenario hinzufügen*-Dialogfeld erscheint wieder (Abbildung 12.17).

Abbildung 12.17:
Zweites Szenario
hinzufügen

Geben Sie dem zweiten Szenario einen Namen wie »Zinsen2«, übernehmen Sie die bereits vorgegebene Selektion von B1 und B2 als veränderbare Zellen und schließen Sie das Dialogfeld mit OK.

Das *Szenariowerte*-Dialogfeld erscheint erneut und Sie definieren diesmal als einzusetzende Werte einen Zinssatz von 6% und wie im ersten Szenario einen Spitzensteuersatz von 40% (Abbildung 12.18).

Abbildung 12.18:
Werte für zweites
Szenario definieren

Auf diese Weise können Sie beliebig viele Szenarien mit unterschiedlichen Variablenwerten definieren, zum Beispiel die folgenden, wenn im vorliegenden Modell alle Kombinationen aus einem Zinssatz von 5%, 6% und 7% und einem Spitzensteuersatz von 40% beziehungsweise 50% in Frage kommen:

Name	Zinssatz	Spitzensteuersatz
Zinsen1	5%	40%
Zinsen2	6%	40%
Zinsen3	7%	40%
Zinsen4	5%	50%
Zinsen5	6%	50%
Zinsen6	7%	50%

Tabelle 12.1:
Modellvarianten

Nach Eingabe der letzten Variante wählen Sie nicht *Hinzufügen*, sondern *OK*. Anschließend erscheint wieder das übergeordnete Dialogfeld, das nun eine Liste der soeben definierten Szenarien enthält (Abbildung 12.19).

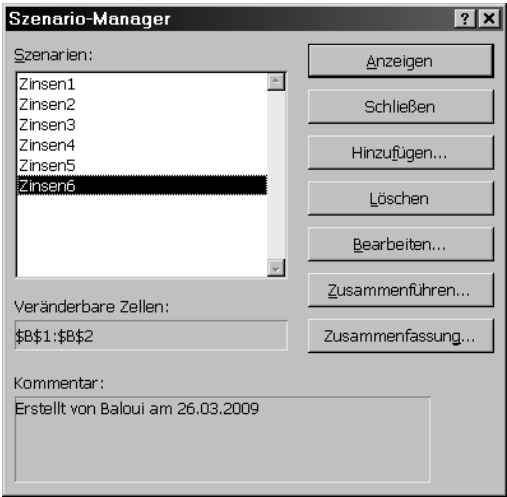
Szenarienliste

Selektieren Sie eines davon, wird unter *Veränderbare Zellen* angezeigt, mit welchen Zellen es operiert und unter *Kommentar* der von Ihnen eingegebene oder von Excel standardmäßig vorgegebene Kommentar.

Löschen entfernt das im Listenfeld selektierte Szenario wieder.

Bearbeiten... öffnet das gleiche Dialogfeld wie *Hinzufügen...*, wobei jedoch das selektierte Szenario darin vorgegeben wird, sodass Sie nun seinen Namen, die definierten Variablenwerte oder den Kommentar nachträglich verändern können.

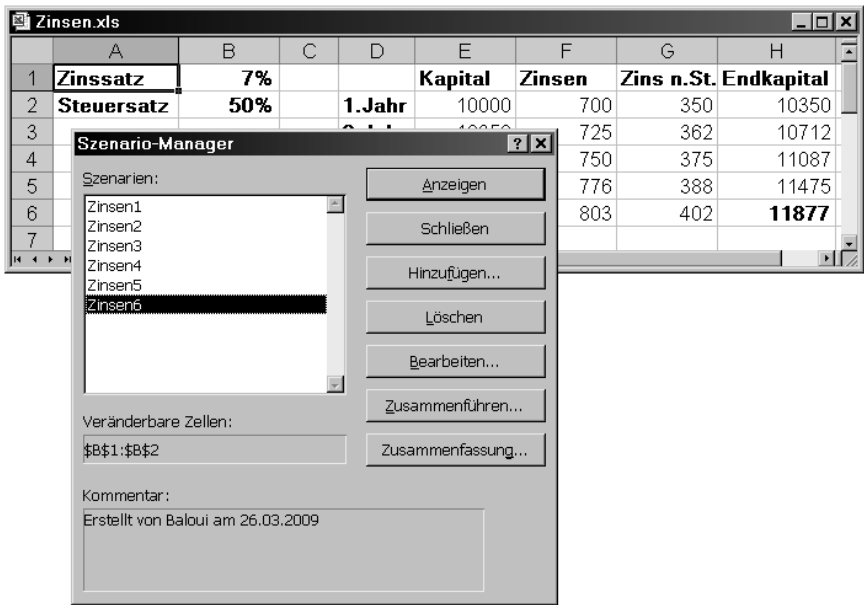
Abbildung 12.19:
Szenarienliste



Szenario anzeigen Wesentlich interessanter ist *Anzeigen*. Diese Schaltfläche oder einfacher ein Doppelklick auf ein Listenelement zeigt, zu welchen Resultaten das betreffende Szenario führt.

Zum Beispiel bewirkt ein Doppelklick auf die Variante »Zinsen6«, dass Excel in die Variablenzellen B1 und B2 die in diesem Szenario definierten Werte einsetzt (Zins: 7%, Spitzensteuersatz: 50%) und die Tabelle entsprechend aktualisiert (Abbildung 12.20).

Abbildung 12.20:
Szenario »Zinsen6«



Szenarien sind an die aktive Tabelle gebunden und werden zusammen mit ihr gespeichert bzw. geladen. Ein einmal erzeugtes Szenario ist somit beim nächsten Öffnen der Tabelle wieder unverändert vorhanden.

Szenarien zusammenführen

Zusätzlich können Sie mit der Schaltfläche *Zusammenführen...* des *Szenario-Manager*-Dialogfelds zu den Szenarien der aktuellen Tabelle Szenarien hinzufügen, die in einer anderen Tabelle oder gar in einer anderen Mappe gespeichert sind (Abbildung 12.21).

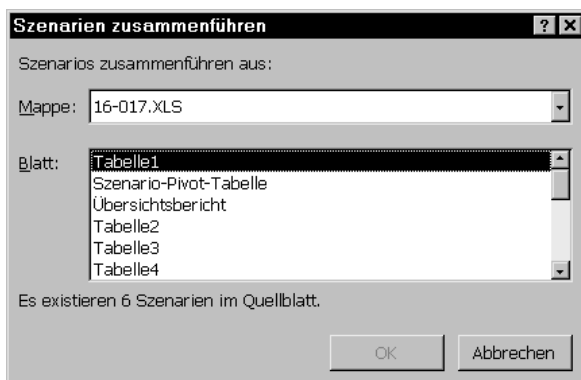


Abbildung 12.21:
Szenarien zusammenführen

Sie selektieren die betreffende Mappe und das darin interessierende Blatt. OK fügt alle in diesem Blatt definierten Szenarien in das aktive Blatt ein – was natürlich nur sinnvoll ist, wenn dieses Blatt identisch aufgebaut ist!

Sehr nützlich ist *Zusammenfassung...*: Excel legt in der aktiven Mappe eine neue Tabelle an, die eine übersichtliche Gesamtdarstellung der verschiedenen Szenarien enthält.

Szenarienberichte

Zunächst müssen Sie jedoch angeben, welche Auswirkungen Sie interessieren, genauer: welche Zellen die für Sie wichtigen Resultate enthalten (Abbildung 12.22).

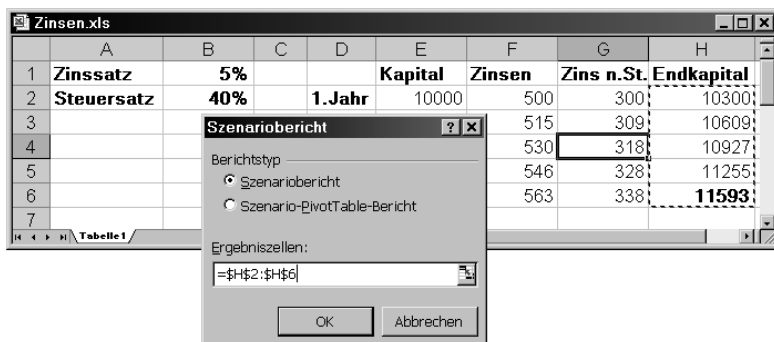


Abbildung 12.22:
Resultatzellen

Im Beispiel würden Sie sich wahrscheinlich gleich für fünf Resultatzellen interessieren, für H2, H3, H4, H5 und H6. Statt diese Zellen einzeln – bei gedrückter **[Strg]**-Taste – zu selektieren, können Sie mit **=H\$2:H\$6** gleich den gesamten Bereich selektieren und danach *Szenariobericht* und **OK** wählen (Abbildung 12.23).

Abbildung 12.23:
Szenarien-Über-
sichtsbericht

Szenariobericht							
Aktuelle Werte:		Zinsen1	Zinsen2	Zinsen3	Zinsen4	Zinsen5	Zinsen6
Veränderbare Zellen:							
\$B\$1	5%	5%	6%	7%	5%	6%	7%
\$B\$2	40%	40%	40%	40%	50%	50%	50%
Ergebniszellen:							
\$H\$2	10300	10300	10360	10420	10250	10300	10350
\$H\$3	10609	10609	10733	10858	10506	10609	10712
\$H\$4	10927	10927	11119	11314	10769	10927	11087
\$H\$5	11255	11255	11520	11789	11038	11255	11475
\$H\$6	11593	11593	11934	12284	11314	11593	11877

Anmerkung: Die Aktuelle Wertespalte repräsentiert die Werte der veränderbaren Zellen zum Zeitpunkt, als der Szenariobericht erstellt wurde. Veränderbare Zellen für Szenarien sind in grau hervorgehoben.

Diese Übersicht zeigt auf einen Blick, welches Endkapital sich nach einem, zwei, drei, vier oder fünf Jahren bei einem bestimmten Zinssatz und einem bestimmten Spitzensteuersatz ergibt.



Wählen Sie statt *Szenariobericht* die Option *Szenario-PivotTable-Bericht*, erzeugt Excel eine gleichnamige Tabelle, die eine *Pivot-Tabelle* enthält (siehe Kapitel 16, »Pivot-Tabellen«) (Abbildung 12.24).

Abbildung 12.24:
Szenario-Pivot-
Tabelle

	A	B	C	D	E	F
1	\$B\$1:\$B\$2 von (Alle)					
2						
3		Ergebniszellen				
4	\$B\$1:\$B\$2	\$H\$2	\$H\$3	\$H\$4	\$H\$5	\$H\$6
5	Zinsen1	10300	10609	10927,27	11255,0881	11592,74074
6	Zinsen2	10360	10732,96	11119,34656	11519,64304	11934,35019
7	Zinsen3	10420	10857,64	11312,66089	11789,82464	12284,96569
8	Zinsen4	10250	10506,25	10769,1038	11038,11255	11475,11877
9	Zinsen5	10300	10609	10733	10858	10927
10	Zinsen6	10350	10712,25	11087	11255	11475
11						

PivotTable-Feldliste

- ☒ \$B\$1:\$B\$2
- ☒ \$B\$1:\$B\$2 von
- ☒ Erg. \$H\$2
- ☒ Erg. \$H\$3
- ☒ Erg. \$H\$4
- ☒ Erg. \$H\$5
- ☒ Erg. \$H\$6

Hinzufügen zu Zeilenbereich

12.4 Ansichten-Manager

Das Add-In *Ansichten-Manager* (zur Einbindung siehe Anhang 6, »Add-Ins«) ermöglicht die Erzeugung unterschiedlicher Ansichten Ihrer Tabelle und das Umschalten zwischen diesen Ansichten auf Tastendruck.

Aktivieren Sie beispielsweise »Tabelle 1« von ZINSEN.XLS und wählen Sie ANSICHT|BENUTZERDEFINIERT ANSICHTEN... (Abbildung 12.25).

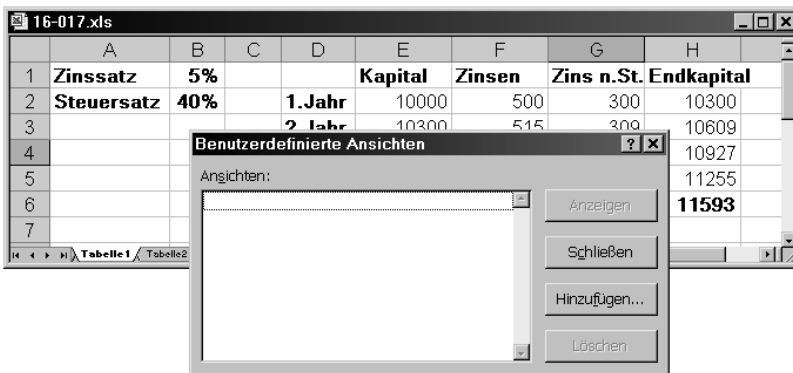


Abbildung 12.25:
Leere Ansichtenliste

Fügen Sie der Ansichtenliste die aktuelle Ansicht der Tabelle hinzu, indem Sie *Hinzufügen...* wählen und ihr im folgenden Dialogfeld einen Namen wie »Ansicht1« geben (Abbildung 12.26).



Abbildung 12.26:
»Ansicht1«
hinzufügen

Blenden Sie anschließend im Register *Ansicht* des Befehls EXTRAS|OPTIONEN... die Gitternetzlinien und die Zeilen- und Spaltenüberschriften aus, wählen Sie erneut ANSICHT|BENUTZERDEFINIERT ANSICHTEN... und fügen Sie mit *Hinzufügen* auch diese zweite Ansicht unter dem Namen »Ansicht2« hinzu (Abbildung 12.27).

Nun sind beide Ansichten im Listenfeld des Befehls ANSICHT|BENUTZERDEFINIERT ANSICHTEN... enthalten und Sie können mit einem Doppelklick auf ein Listenelement (oder einfachem Klick und Wahl von *Anzeigen*) die zugehörige Tabellenansicht aktivieren (Abbildung 12.28).

Löschen entfernt eine nicht länger benötigte Ansicht aus der Liste.

Abbildung 12.27:
»Ansicht2«
hinzufügen

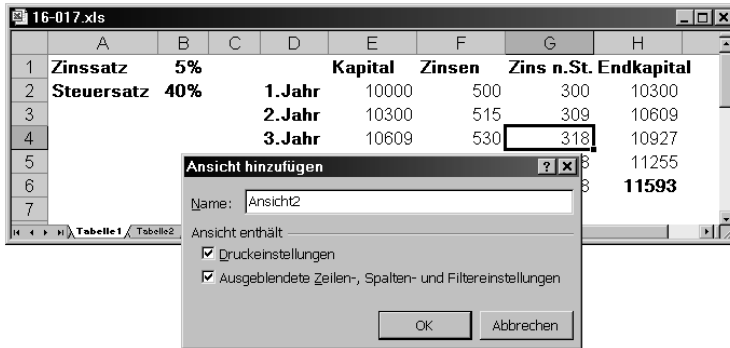


Abbildung 12.28:
Verfügbare
Ansichten



Es ist nicht möglich, Ansichten mit unterschiedlichen Zellformatierungen zu erzeugen, zum Beispiel eine Ansicht, in der eine bestimmte Summenzelle fett formatiert ist, und eine weitere, in der der Zelleninhalt nicht fett, dafür jedoch kursiv dargestellt wird. Stattdessen werden durch unterschiedliche Ansichten allgemeine Eigenschaften des gesamten Arbeitsblatts gespeichert!

**Druck-
einstellungen**

Sehr nützlich ist, dass dazu auch die aktuelle Seiteneinrichtung gehört – vorausgesetzt, beim Hinzufügen einer Ansicht ist das Kontrollkästchen *Druckeinstellungen* aktiviert. Dadurch können Sie vor dem Ausdruck auf Tastendruck zwischen unterschiedlichen Seiteneinrichtungen umschalten.

Aktivieren Sie beispielsweise im Register *Tabelle* von DATEI|SEITE EINRICHTEN... das Kontrollkästchen *Gitternetzlinien* und fügen Sie den aktuellen Zustand als Ansicht »X« hinzu.

Deaktivieren Sie anschließend das Kontrollkästchen *Gitternetzlinien* und fügen Sie den aktuellen Zustand als Ansicht »Y« hinzu.

Sie haben nun zwei Ansichten mit unterschiedlichen Druckeinstellungen definiert. Je nachdem, ob Sie die Tabelle mit oder ohne Gitternetzlinien ausdrucken wollen, müssen Sie vor dem Ausdruck nur noch die gewünschte Ansicht aktivieren.

Gliederungen

Analog dazu wird zusammen mit einer Ansicht der aktuelle Gliederungs- und Filterungszustand gespeichert, wenn beim Hinzufügen das Kontrollkästchen *Ausgeblendete Zeilen, Spalten- und Filtereinstellungen* aktiviert ist.

13 Listen und Datenbanken

Verlassen wir vorübergehend das Gebiet »Kalkulation« und wenden uns einem völlig anderen Thema zu: dem Einsatz von Excel als Datenbank.

Zugegeben, »Dateiverwaltung« ist sicher der passendere Ausdruck, denn mit echten Datenbanken wie Access kann sich Excel nicht ganz messen (aber immerhin fast, wie das Kapitel über Query noch zeigen wird!).

Passend angeordnete Listen bilden für Excel eine kleine Datenbank. Zum Beispiel Adressen, die sich untereinander befinden, wobei sich in jeder Zeile genau eine Adresse befindet und zwar jeweils in Spalte A der Nachname, in Spalte B der Vorname, in Spalte C die Straße und so weiter (Abbildung 13.1).

	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.09	0,00
3	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826125	21.01.09	0,00
4	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.02.09	123,56
5	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.08.09	6254,00
6	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	30.11.09	573,70
7	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.10.09	0,00
8	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.09.09	875,64
9	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.02.09	273,90
10	Böhn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.09	0,00
11	Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/726384	01.02.09	0,00
12	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.09	3864,00
13	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.11.09	1925,00

Abbildung 13.1:
Adressen.xls,
»Tabelle1«

Entscheidend ist, dass sich in der obersten Zeile Spaltenüberschriften befinden, die so genannte Kopfzeile, und alle Zeilen darunter jeweils ähnliche Datengruppen enthalten, beispielsweise je eine Adressen oder Bestellungen. Die einzelnen Zeilen heißen Datensätze, die Inhalte der zugehörigen Spalten sind die Felder und die Spaltenüberschriften die Feldnamen.

Viele Tabellen enthalten derartige Listen. Excel kann sie sortieren, filtern, bestimmte Datensätze darin suchen, einfügen oder löschen.

All diese Aktionen beziehen sich auf die gesamte Liste, wenn vor Anwahl eines Befehls wie DATEN|SORTIEREN... eine einzige Zelle der Liste aktiv ist. Dann selektiert Excel nach der Befehlswahl automatisch die gesamte Liste und bearbeitet sie als Einheit.

Alternativ dazu können Sie vor der Befehlswahl einen bestimmten Bereich selektieren. DATEN|SORTIEREN... und ähnliche Befehle behandeln dann nur diesen Bereich.

Im folgenden Kapitel erläutere ich, wie Sie derartige Listen in vielfältigster Weise manipulieren können:

- ➔ Ich zeige Ihnen, wie Listen sortiert werden, beispielsweise nach Nachname und – bei mehreren identischen Nachnamen – wiederum nach Vorname als zweitem Sortierkriterium.
- ➔ Sie lernen, derartige Listen zu filtern, zum Beispiel um nur Adressen im Raum Mannheim und Nürnberg anzuzeigen oder um sich mit einer »Top 10«-Auswahl beispielsweise nur die fünf Zeilen mit den höchsten Werten im Feld »Verkaufspreis« anzeigen zu lassen.
- ➔ Ich erläutere, wie Sie Datenmasken benutzen, die – wie von Adressverwaltungen her gewohnt – in einer Maske genau einen Datensatz anzeigen, beispielsweise eine Adresse.
- ➔ Sie erfahren, wie Sie sortierte Listen nach dem Inhalt eines beliebigen Feldes gruppieren, um *für jede Gruppe getrennt* Auswertungen vorzunehmen und beispielsweise die Summe oder den Durchschnitt der in einem numerischen Feld enthaltenen Werte ermitteln zu lassen.
- ➔ Zum Schluss zeige ich, wie Sie mithilfe des *Vorlagen*-Assistenten bestimmte Werte von Tabellen in einer Datenbank archivieren und auf welche Art und Weise Excel mit Access zusammenarbeiten kann, beispielsweise um zur Verwaltung einer Excel-Liste ein Access-Formular zu benutzen.

13.1 Sortieren

Um eine Liste zu sortieren, bewegen Sie den Cursor zu irgendeiner Zelle der Liste und wählen DATEN|SORTIEREN... (Abbildung 13.2).

13.1.1 Sortierkriterien

Excel selektiert die gesamte Liste und erkennt dabei zugehörige Feldnamen. Diese Feldnamen befinden sich im Listenfeld *Sortieren nach*. Darin selektieren Sie nun die gewünschte Sortierspalte, beispielsweise »Name«. Klicken Sie danach auf *OK*, wird die Liste nach Namen aufsteigend sortiert (Abbildung 13.3).



Das abgebildete Symbol sortiert eine Liste aufsteigend nach dem Inhalt jener Spalte, in der sich die momentan aktive Zelle befindet, und zwar mit den Standardsortier-Optionen bzw. den im letzten Sortierlauf verwendeten Optionen.

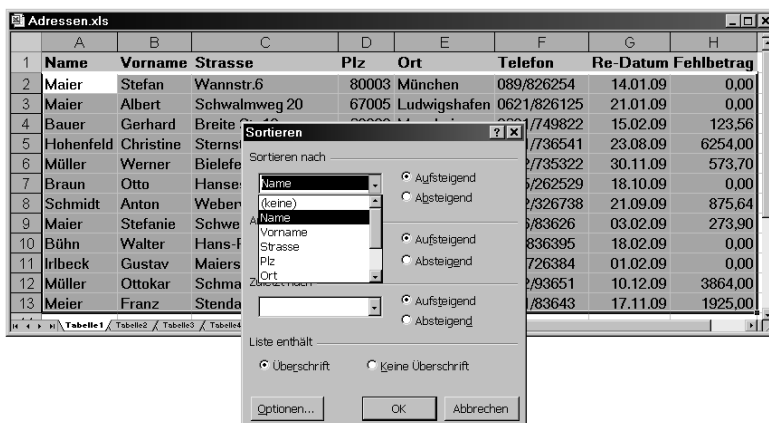


Abbildung 13.2:
Sortieren einer Liste

	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.02.09	123,56
3	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.10.09	0,00
4	Böhm	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.09	0,00
5	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.08.09	6254,00
6	Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/726384	01.02.09	0,00
7	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.09	0,00
8	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826125	21.01.09	0,00
9	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.02.09	273,90
10	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.11.09	1925,00
11	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	30.11.09	573,70
12	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.09	3864,00
13	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.09.09	875,64

Abbildung 13.3:
Ergebnis der
Sortierung

Dieses Symbol sortiert gerade umgekehrt in absteigender Reihenfolge, sodass sich »Bauer« am Ende und »Schmidt« am Anfang der Liste befindet.



Wird die Liste wie in der Abbildung nach »Name« sortiert, befinden sich anschließend etwa in der Mitte drei »Maier«, nämlich »Stefan«, »Albert« und danach »Stefanie« (siehe Bild 13.3).

Enthalten wie hier mehrere Zeilen identische Primärschlüssel, dann sollte die Liste mit einem zusätzlichen Sekundärschlüssel untersortiert werden.

Dazu wählen Sie zusätzlich im Listenfeld *Anschließend nach* das Feld »Vorname« aus und aktivieren beispielsweise *Absteigend* (Abbildung 13.4).

Mit diesen Kriterien werden Datensätze (Zeilen), die in der Spalte »Name« identische Einträge aufweisen, untereinander nach »Vorname« sortiert und zwar in absteigender Reihenfolge, sodass zuerst die Zeile mit dem alphabetisch kleinsten Vornamen kommt, danach jene mit dem zweitkleinsten Vornamen und so weiter; also zuerst »Stefanie«, dann »Stefan« und erst zuletzt »Albert« (Abbildung 13.5).

Abbildung 13.4:
Sekundärschlüssel
und absteigende
Sortierung

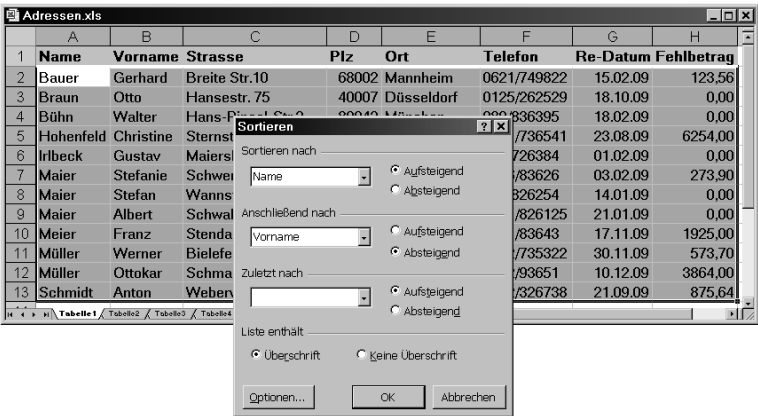


Abbildung 13.5:
Sortierergebnis

Adressen.xls								
	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.02.09	123,56
3	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.10.09	0,00
4	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.09	0,00
5	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.08.09	6254,00
6	Irlbeck	Gustav	Maiersbächerweg 21	80017	München	089/726384	01.02.09	0,00
7	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.02.09	273,90
8	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.09	0,00
9	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826125	21.01.09	0,00
10	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.11.09	1925,00
11	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	30.11.09	573,70
12	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.09	3864,00
13	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.09.09	875,64

Im untersten Feld *Zuletzt nach* können Sie entsprechend einen weiteren Unterschlüssel definieren, der angewandt wird, wenn mehrere Sätze sowohl für den ersten als auch für den zweiten Schlüssel identische Inhalte besitzen; wenn es zum Beispiel zwei oder drei »Ottokar Müller« gibt und Sie diese untereinander zum Beispiel nach »Ort« sortieren wollen.

Überschriften

Excel erkennt meist *selbstständig*, ob sich in der ersten Zeile einer Liste Feldnamen befinden. Wenn ja, werden diese nicht selektiert und unter *Liste enthält* wird die Option *Überschrift* vorselektiert (siehe Bild 13.4).

Allerdings gibt es Listen, bei denen diese Erkennung versagt (Abbildung 13.6).

In diesem Beispiel unterscheidet sich die oberste Tabellenzeile »Name«, »Vorname« und »Straße« nicht von den nachfolgenden Zeilen.

Sie ist weder anders formatiert als diese noch enthält sie in irgendeiner Spalte eine völlig andere Datenart, was in den vorhergehenden Beispielen immer der Fall war: Die oberste Zeile mit den Überschriften enthielt in Spalte D den Text »Plz«, die Felder darunter jedoch Zahlen, sodass sich die Überschriftszeile eindeutig von den folgenden Zeilen unterschied.

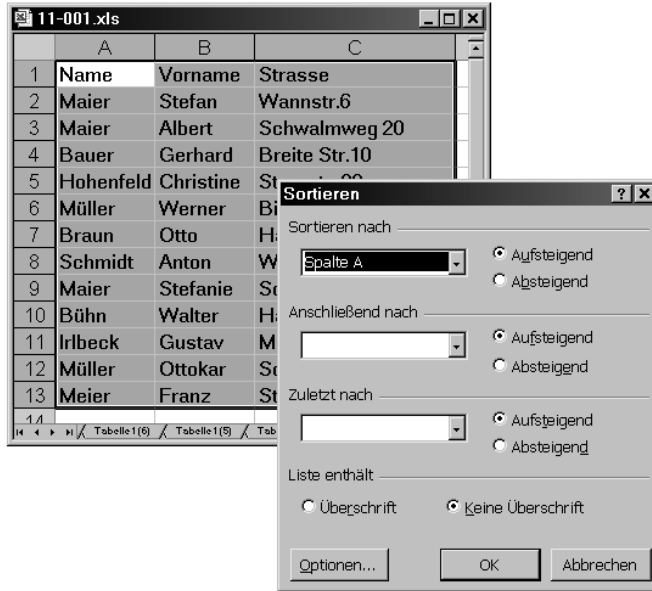


Abbildung 13.6:
Nicht erkannte
Feldnamen

Da diese Unterscheidung nun fehlt, erkennt Excel die Überschriftszeile nicht, selektiert sie, als würde sie zur Liste selbst gehören, und gibt *keine Überschrift* vor.

Da Excel die Überschriften nicht erkennt, fehlt nun unter *Sortieren nach* die Vorgabe der Feldnamen und die Liste enthält stattdessen nur die üblichen Spaltenbezeichnungen A, B, C etc.

Würden Sie die Liste so sortieren lassen, erhielten Sie folgendes Resultat (Abbildung 13.7):

	A	B	C
1	Bauer	Gerhard	Breite Str.10
2	Braun	Otto	Hansestr. 75
3	Bühn	Walter	Hans-Pinsel-Str.2
4	Hohenfeld	Christine	Sternstr. 22
5	Irlbeck	Gustav	Maiersbacherweg 21
6	Maier	Stefan	Wannstr.6
7	Maier	Albert	Schwalmweg 20
8	Maier	Stefanie	Schweriner Weg 18
9	Meier	Franz	Stendaler Weg 23
10	Müller	Werner	Bielefelder Weg 2
11	Müller	Ottokar	Schmalweg 9
12	Name	Vorname	Strasse
13	Schmidt	Anton	Weberweg 54

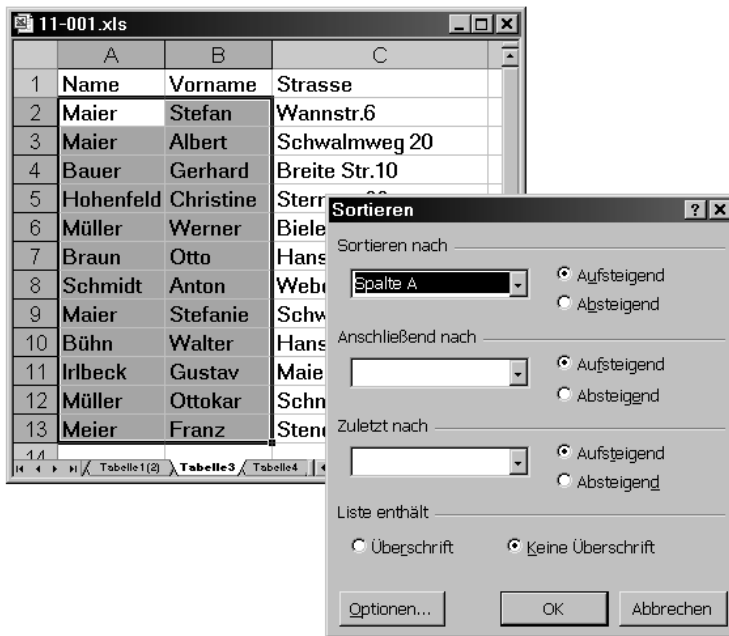
Abbildung 13.7:
Fehlerhafte Sortie-
rung

Die Überschriftszeile befindet sich nun in Zeile 12, wurde also einfach mitsortiert.

Um diesen Fehler zu vermeiden, müssen Sie unter *Liste enthält* die Option *Überschrift* selektieren und Excel damit bekanntgeben, dass die oberste Zeile Überschriften enthält und nicht mitsortiert werden soll! Daraufhin wird Excel die Selektion ändern und die Zeile nicht mehr mitselektieren – und entsprechend auch nicht mitsortieren.

Alternativ dazu können Sie den zu sortierenden Bereich gezielt angeben, indem Sie ihn vor der Befehlswahl selektieren (Abbildung 13.8).

Abbildung 13.8:
Manuelle Bereichs-
selektion



Selektieren Sie wie hier den Bereich A2:B13, so ist die Option *keine Überschrift* korrekt, da der manuell selektierte Bereich ja tatsächlich keine Überschriftszeile enthält.



Zusätzlich werden mit dieser Selektion nur die Spalten A und B sortiert. Die Spalte C bleibt unverändert (Abbildung 13.9).

Dass Spalte C nicht mitsortiert wurde, ist hier zwar unsinnig, allerdings gibt es in der Praxis sicher Fälle, in denen Excel tatsächlich zu viel selektieren und sortieren würde und in denen Sie den zu sortierenden Bereich daher gezielt angeben müssen.

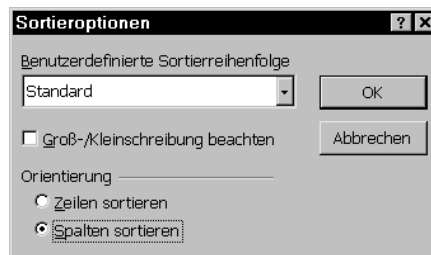


	A	B	C
1	Name	Vorname	Strasse
2	Bauer	Gerhard	Wannstr.6
3	Braun	Otto	Schwalmweg 20
4	Bühn	Walter	Breite Str.10
5	Hohenfeld	Christine	Sternstr. 22
6	Irlbeck	Gustav	Bielefelder Weg 2
7	Maier	Stefan	Hansestr. 75
8	Maier	Albert	Weberweg 54
9	Maier	Stefanie	Schweriner Weg 18
10	Meier	Franz	Hans-Pinsel-Str.2
11	Müller	Werner	Maiersbacherweg 21
12	Müller	Ottokar	Schmalweg 9
13	Schmidt	Anton	Stendaler Weg 23

Abbildung 13.9:
Resultierende
Sortierung

13.1.2 Sortieroptionen

Die Schaltfläche *Optionen...* des *Sortieren*-Dialogfelds öffnet folgendes Dialogfeld (Abbildung 13.10):



Sortieroptionen

Benutzerdefinierte Sortierreihenfolge
 Standard

☐ Groß-/Kleinschreibung beachten

Orientierung
☐ Zeilen sortieren
☒ Spalten sortieren

OK Abbrechen

Abbildung 13.10:
Sortieroptionen

Aktivieren Sie *Groß-/Kleinschreibung beachten*, sortiert Excel bei aufsteigender Sortierung »maier« vor »Maier« ein und betrachtet beides nicht wie sonst als gleichwertig.

Unter *Orientierung* legen Sie fest, ob wie zuvor Zeilen oder aber diesmal Spalten sortiert werden sollen; beispielsweise um die Spalten der Demo-Anwendung von links nach rechts aufsteigend anzuordnen.

*Spalten oder
Zeilen sortieren*

Nach OK sind in den drei Sortier-Listenfeldern statt Spaltenbezeichnungen Zeilennummern enthalten (Abbildung 13.11).

Die Auswahl von »Zeile 1« unter *Sortieren nach* bedeutet, dass nach dem Inhalt der ersten *Zeile* sortiert wird, in der sich die verschiedenen Überschriften befinden (Abbildung 13.12).

Abbildung 13.11:
Zeilenkriterium



Abbildung 13.12:
Zeilenkriterium

	A	B	C	D	E	F	G	H
1	Fehlbetrag	Name	Ort	Plz	Re-Datum	Strasse	Telefon	Vorname
2	0,00	Maier	München	80003	14.01.2009	Wannstr.6	089/826254	Stefan
3	0,00	Maier	Ludwigshafen	67005	21.01.2009	Schwalmweg 20	0621/82612	Albert
4	123,56	Bauer	Mannheim	68002	15.02.2009	Breite Str.10	0621/74982	Gerhard
5	6254,00	Hohenfeld	Ludwigshafen	67009	23.08.2009	Sternstr. 22	0621/73654	Christine
6	573,70	Müller	Berlin	10023	30.11.2009	Bielefelder Weg 2	0372/73532	Werner
7	0,00	Braun	Düsseldorf	40007	18.10.2009	Hansestr. 75	0125/26252	Otto
8	875,64	Schmidt	Heidelberg	69002	21.09.2009	Weberweg 54	0662/32673	Anton
9	273,90	Maier	Hamburg	20034	03.02.2009	Schweriner Weg 18	0326/83626	Stefanie
10	0,00	Bühn	München	80042	18.02.2009	Hans-Pinsel-Str.2	089/836395	Walter
11	0,00	Irlbeck	München	80017	01.02.2009	Maiersbacherweg 21	089/726384	Gustav
12	3864,00	Müller	Berlin	10065	10.12.2009	Schmalweg 9	0372/93651	Ottokar
13	1925,00	Meier	Mannheim	68003	17.11.2009	Stendaler Weg 23	0621/83643	Franz

Die Spalte, die in Zeile 1 den alphabetisch kleinsten Text enthält, »Fehlbetrag«, kommt ganz nach links, und jene Spalte, die in dieser Zeile den größten Text »Vorname« enthält, ganz nach rechts. Bei absteigender Sortierung wäre die Reihenfolge genau umgekehrt.



Was für diese Abbildung bereits erledigt wurde: Nach dem Vertauschen von Spalten müssen Sie die Spaltenbreiten noch an die neuen Spalteninhalte anpassen (alle Spalten selektieren und Doppelklick auf einen Trennstrich zwischen zwei Spaltenbuchstaben).

Sortieren Sie nicht nach dem Inhalt von Zeile 1, der Überschriftszeile, sondern beispielsweise nach dem von Zeile 2, wären die Folgen höchst unerwünscht (Abbildung 13.13).

Sortierreihenfolge

Um dieses Sortierergebnis zu verstehen, müssen Sie Excels Prioritäten beim Sortieren unterschiedlicher Datenarten kennen:

1. Zahlen
2. Texte
3. Wahrheitswerte
4. Fehlerzellen
5. Leere Zellen

	A	B	C	D	E	F	G	H
	Fehlbetrag	Re-Datum	Plz	Telefon	Name	Ort	Vorname	Strasse
1								
2	0,00	14.01.2009	80003	089/826254	Maier	München	Stefan	Wannstr.6
3	0,00	21.01.2009	67005	0621/826125	Maier	Ludwigshafen	Albert	Schwalmweg 20
4	123,56	15.02.2009	68002	0621/749822	Bauer	Mannheim	Gerhard	Breite Str.10
5	6254,00	23.08.2009	67009	0621/736541	Hohenfeld	Ludwigshafen	Christine	Sternstr. 22
6	573,70	30.11.2009	10023	0372/735322	Müller	Berlin	Werner	Bielefelder Weg 2
7	0,00	18.10.2009	40007	0125/262529	Braun	Düsseldorf	Otto	Hansestr. 75
8	875,64	21.09.2009	60002	0662/326738	Schmidt	Heidelberg	Anton	Weberweg 54
9	273,90	03.02.2009	20034	0326/83626	Maier	Hamburg	Stefanie	Schweriner Weg 18
10	0,00	18.02.2009	80042	089/836395	Bühn	München	Walter	Hans-Pinsel-Str.2
11	0,00	01.02.2009	80017	089/726384	Irlbeck	München	Gustav	Maiersbacherweg 21
12	3864,00	10.12.2009	10065	0372/93651	Müller	Berlin	Ottokar	Schmalweg 9
13	1925,00	17.11.2009	68003	0621/83643	Meier	Mannheim	Franz	Stendaler Weg 23

Abbildung 13.13:
Zeile 2 als Sortierkriterium

Jene Spalte kommt ganz nach links, die in Zeile 2 gemäß dieser Prioritäten den kleinsten Wert enthält: die Spalte »Fehlbetrag«, die in Zeile 2 eine Zahl enthält, nämlich 0.

Daneben befindet sich die Spalte mit dem zweitkleinsten Wert, dem Datum 14.1.09, das für Excel ja ebenfalls eine Zahl ist, nämlich die zugehörige serielle Zahl.

Daneben wiederum befindet sich die Spalte »Plz« mit der Zahl 80003.

Erst jetzt folgen jene Spalten, die in Zeile 2 keine Zahlen, sondern Texte wie »089/826254« (eine Telefonnummer mit Vorwahl ist wegen des »/« keine Zahl, sondern ein Text), »Maier« oder »München« enthalten.

Für spezielle Anwendungen können Sie im Register Benutzerdefinierte Listen des Befehls EXTRAS|OPTIONEN... eigene Sortierreihenfolgen definieren (siehe Anhang A.7, »Excel individuell konfigurieren«), zum Beispiel die Reihenfolge »Kiwis«, »Äpfel«, »Orangen«, »Birnen«.



Die zusätzliche Liste erscheint im Listenfeld *Benutzerdefinierte Sortierreihenfolge* des *Sortieroptionen*-Dialogfelds (Abbildung 13.14).

Wenden Sie sie auf die im Hintergrund sichtbare Tabelle an und sortieren Sie sie nach »Obsttyp«, hält sich Excel dabei an die von Ihnen angegebene Sortierreihenfolge (Abbildung 13.15).

Abbildung 13.14:
Benutzerdefinierte
Sortierreihenfolge

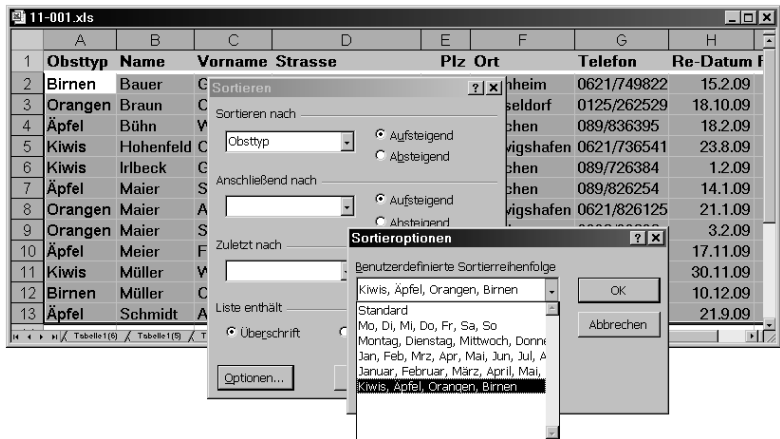


Abbildung 13.15:
Ergebnis der
Sortierung

	A	B	C	D	E	F	G	H
1	Obsttyp	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum
2	Kiwis	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.8.09
3	Kiwis	Irlbeck	Gustav	Maierbachweg 21	80017	München	089/726384	1.2.09
4	Kiwis	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	30.11.09
5	Äpfel	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.2.09
6	Äpfel	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.1.09
7	Äpfel	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.11.09
8	Äpfel	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.9.09
9	Orangen	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.10.09
10	Orangen	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826125	21.1.09
11	Orangen	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	3.2.09
12	Birnen	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.2.09
13	Birnen	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.09

13.2 Filtern

In Listen organisierte Daten kann Excel filtern, das heißt nur ausgewählte Daten anzeigen.



Merken Sie sich bitte gleich, dass Sie mit DATEN|FILTER|ALLE ANZEIGEN jegliche Filterung aufheben und wieder die gesamte Liste anzeigen können.

13.2.1 AutoFilter



Selektieren Sie eine beliebige Zelle der Liste und wählen Sie DATEN|FILTER|AUTOFILTER oder klicken Sie das abgebildete Symbol an (Kategorie *Daten* des Anpassen-Dialogfelds) (Abbildung 13.16).

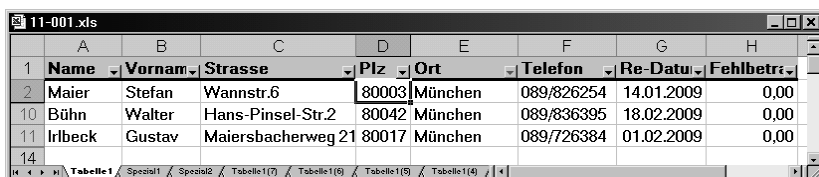
Excel versteht die einzelnen Feldnamen mit DropDown-Pfeilen. Durch Anklicken öffnen Sie entsprechend der Abbildung die zugehörige Liste, die alle Elemente der betreffenden Spalte enthält.



	A	B	C	D	E	F	G	H
	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
1	Maier	Stefan	Wannstr.6	80003	(Alle)	089/826254	14.01.2009	0,00
2	Maier	Albert	Schwalmweg 20	67005	(Top 10...)	0621/826125	21.01.2009	0,00
3	Bauer	Gerhard	Breite Str.10	68002	Berlin	0621/749822	15.02.2009	123,56
4	Hohenfeld	Christine	Sternstr. 22	67009	Düsseldorf	0621/736541	23.08.2009	6254,00
5	Müller	Werner	Bielefelder Weg 2	10023	Hamburg	0372/735322	30.11.2009	573,70
6	Braun	Otto	Hansestr. 75	40007	Heidelberg	0125/262529	18.10.2009	0,00
7	Schmidt	Anton	Weberweg 54	60002	Ludwigshafen	0662/326738	21.09.2009	875,64
8	Maier	Stefanie	Schweriner Weg 18	20034	Mannheim	0326/83626	03.02.2009	273,90
9	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.2009	0,00
10	Irlbeck	Gustav	Maierbachweg 21	80017	München	089/726384	01.02.2009	0,00
11	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.2009	3864,00
12	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.11.2009	1925,00

Abbildung 13.16:
AutoFilter

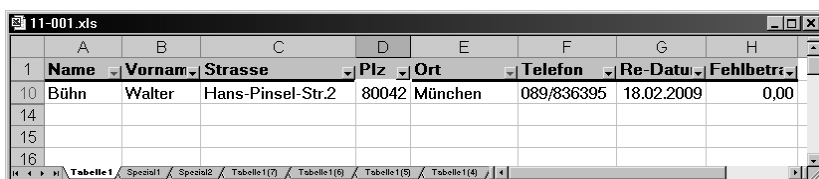
Selektieren Sie in der Spalte »Ort« einen Eintrag wie »München«, so werden nur noch jene Zeilen angezeigt, die in der Spalte »Ort« diesen Eintrag enthalten (Abbildung 13.17).



	A	B	C	D	E	F	G	H
	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.2009	0,00
10	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.2009	0,00
11	Irlbeck	Gustav	Maierbachweg 21	80017	München	089/726384	01.02.2009	0,00

Abbildung 13.17:
Gefilterte Liste

Selektieren Sie zusätzlich in der Drop-Down-Liste der Spalte »Name« den Eintrag »Bühn«, werden nur noch jene Zeilen angezeigt, die im Feld »Ort« den Eintrag »München« *und* im Feld »Name« den Eintrag »Bühn« enthalten (Abbildung 13.18).



	A	B	C	D	E	F	G	H
	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
10	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.2009	0,00

Abbildung 13.18:
UND-Verknüpfung

Mehrere Filter werden somit mit *UND* verknüpft und nur jene Zeilen angezeigt, die *allen* Filterbedingungen genügen.

Jede Filterliste enthält einige vordefinierte Elemente:

- ➔ »(Alle)« hebt den betreffenden Filter wieder auf.
- ➔ »(Top 10...)« zeigt nur die Zeilen an, die die größten bzw. die kleinsten Werte enthalten.
- ➔ »(Benutzerdefiniert...)« ermöglicht die Eingabe individueller Filterbedingungen.

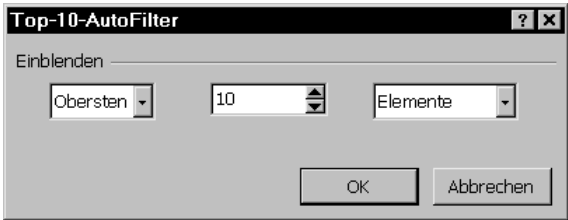
Dass eine Tabelle gefiltert ist, zeigt Excel durch einen blauen statt schwarzen Drop-Down-Pfeil neben der betreffenden Spaltenbezeichnung an. Um den Filter für diese Spalte wieder aufzuheben, wählen Sie aus der Filterliste der Spalte das Element »(Alle)«.

DATEN|FILTER|AUTOFILTER hebt dagegen die gesamte AutoFilterung auf und entfernt alle Drop-Down-Pfeile (DATEN|FILTER|ALLE ANZEIGEN hebt dagegen nur die Filterung auf).

13.2.2 Top 10-Auswahl

Das Listenelement »(Top 10...)« können Sie nur in Feldern verwenden, die wie »Plz« oder »Fehlbetrag« numerische Werte enthalten (Abbildung 13.19).

Abbildung 13.19:
Top 10-AutoFilter



Unter *Einblenden* wählen Sie, ob die höchsten oder die niedrigsten Werte angezeigt werden sollen.

Daneben legen Sie fest, wie viele Datensätze Excel zeigen soll oder übernehmen einfach die Vorgabe 10.

Ganz rechts selektieren Sie »Elemente«, um (mit den Standardvorgaben »Obersten« und »10« die zehn Zeilen mit den höchsten Werten beispielsweise im Feld »Plz« anzeigen zu lassen (Abbildung 13.20).

Abbildung 13.20:
Anzeige der Top 10

Adressen.xls								
	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datu	Fehlbetr
2	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.09	0,00
3	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826125	21.01.09	0,00
4	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.02.09	123,56
5	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.08.09	6254,00
7	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.10.09	0,00
8	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.09.09	875,64
9	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.02.09	273,90
10	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.09	0,00
11	Irlbeck	Gustav	Maibacherweg 21	80017	München	089/726384	01.02.09	0,00
13	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.11.09	1925,00
14								
Tabellenkette: Tabelle1 Tabelle2 Tabelle3 Tabelle4 Tabelle5 Tabelle6 Tabelle7 Tabelle8								

Die Alternative »Prozent« würde stattdessen 10% aller Zeilen anzeigen. Da Excel dabei abrundet, würde von den zwölf Datensätzen des Beispiels nur einer angezeigt werden, eben jener mit der größten Postleitzahl aller zwölf Sätze.

13.2.3 Filterkriterien

Das Listenelement »(Benutzerdefiniert...)« öffnet ein Dialogfeld (Abbildung 13.21).

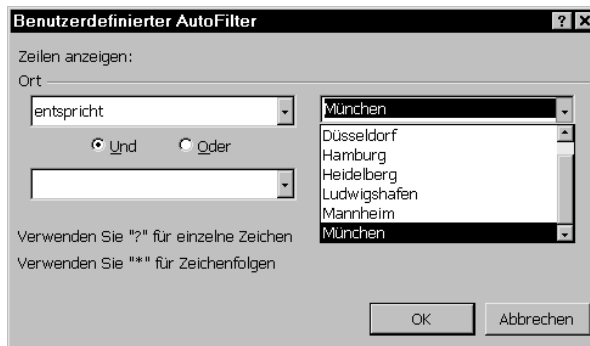


Abbildung 13.21:
Benutzerdefinierter
Filter

Eine Filterbedingung besteht aus einem Vergleichswert, der mithilfe eines Vergleichsoperators mit dem Feldinhalt verglichen wird.

Das linke Listefeld enthält alle verfügbaren Vergleichsoperatoren und das rechte Listefeld die Vergleichswerte, bei denen es sich ganz einfach um alle in der betreffenden Spalte enthaltenen Werte handelt.

Definieren Sie für die Spalte »Ort« einen Filter, indem Sie links »entspricht« selektieren und rechts »München«, so lautet die Filterbedingung

=München.

Dieser Filter blendet alle Zeilen aus, die in dieser Spalte einen anderen Eintrag als München enthalten.

Sie können sogar zwei Filterkriterien eingeben und sie mithilfe der gleichnamigen Optionen des Dialogfelds mit den logischen Operatoren *UND* und *ODER* verknüpfen.

Bei der *UND*-Verknüpfung werden nur Datensätze angezeigt, die *beiden* Bedingungen entsprechen.

UND-Verknüpfung

Bei der *ODER*-Verknüpfung werden Datensätze angezeigt, die *wenigstens einer* der beiden Bedingungen entsprechen.

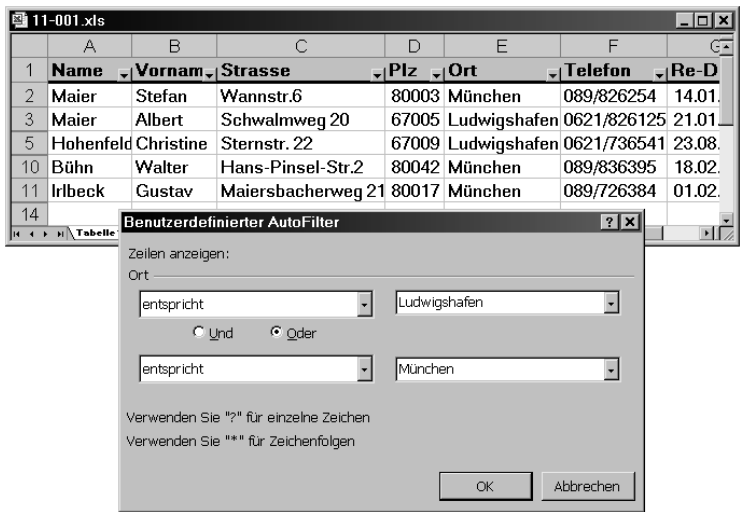
*ODER-
Verknüpfung*

Suchen Sie alle Personen, die in Ludwigshafen *oder* in München wohnen, dann definieren Sie für das Feld »Ort« den Filter

entspricht Ludwigshafen Oder entspricht München.

Für beide Kriterien selektieren Sie den Vergleichsoperator »entspricht«. Einmal geben Sie als Vergleichswert »Ludwigshafen« und das zweite Mal »München« ein, vor allem aber: Sie selektieren als Verknüpfungs-Option *Oder*. Daraufhin werden tatsächlich nur noch die gewünschten Datensätze angezeigt (Abbildung 13.22).

Abbildung 13.22:
ODER-Verknüpfung



Mit der *UND*-Verknüpfung wären dagegen nur Sätze angezeigt worden, die beiden Teilbedingungen »entspricht Ludwigshafen« und »entspricht München« *gleichzeitig* entsprechen – was natürlich unsinnig ist.

Vergleichsoperatoren

Im linken Listenfeld stehen Ihnen unterschiedlichste Vergleichsoperatoren zur Verfügung, außer »entspricht« beispielsweise »entspricht nicht«, »ist größer als«, »beginnt mit« etc.

Zum Beispiel zeigt der Operator »ist kleiner als« zusammen mit dem Vergleichswert »M« im Feld »Name« nur Personen an, deren Name alphabetisch gesehen kleiner als »M« ist; beispielsweise »Bauer« und »Hohenfeld«, aber weder »Maier« noch »Müller« (Abbildung 13.23).

Umgekehrt findet »ist größer als« zuammen mit dem Vergleichswert »Maier« Namen wie »Müller« oder »Meier«, aber weder »Bauer« noch »Maier« selbst (Letzteres wäre bei Verwendung von »ist größer oder gleich« und »Maier« der Fall, also bei einem Vergleich der Art »Suche alles, was größer oder gleich »Maier« ist«).



Abbildung 13.23:
Vergleichsoperatoren

Wie das letzte Beispiel zeigt, müssen Sie im rechten Listenfeld keines der vorgegebenen Elemente wie »München« selektieren, sondern können völlig beliebige Vergleichswerte eintippen. Dabei können Sie auch die Sonderzeichen »?« und »*« verwenden.

Sonderzeichen

Ein in einer Zeichenkette enthaltenes Fragezeichen »?« bewirkt, dass Excel sich für das Zeichen an der betreffenden Position nicht interessiert. »M??er« findet daher sowohl »Maier« als auch »Mayer« und »Meier«, da sich Excel weder für den zweiten noch für den dritten Buchstaben des Namens interessiert.

?

Ist die Länge der zu ignorierenden Zeichenkette unbestimmt, verwenden Sie stattdessen »*«. Dieses Zeichen ignoriert eine Zeichenfolge unbestimmter Länge. Zum Beispiel findet »M*feld« alle Namen, die mit »M« beginnen und mit »feld« enden, egal welche und wie viele Zeichen sich dazwischen befinden; zum Beispiel »Maierfeld« und »Müllerfeld«.

*

Beide Sonderzeichen können Sie kombinieren: »M??er*f« findet unter anderem »Meiersdorf« und »Mayersbacherdorf«; eben alle Zeichenketten, die mit »M« beginnen, die als vierten und fünften Buchstaben »e« und »r« enthalten und die mit »f« enden.

Eine Anwendung wäre beispielsweise der Vergleichsoperator »entspricht« zusammen mit dem Vergleichswert »M???r«, der alle »Maier« findet, unabhängig von der Schreibweise (also auch »Meier«, »Mayer« etc.) (Abbildung 13.24).

Abbildung 13.24:
Anwendung von
Sonderzeichen



Wahrscheinlich werden Sie Vergleichsoperatoren häufiger auf Zahlen als auf Zeichenketten anwenden, zum Beispiel auf das Feld »Fehlbetrag«, wenn Sie alle Kunden auswählen wollen, bei denen der noch ausstehende Restbetrag höher als 5.000 DM ist (Filter »ist größer oder gleich« als »5000« für das Feld »Fehlbetrag«).

Oder alle Müller, die im Postleitzahlgebiet 50000 und größer wohnen (zwei Filterkriterien in verschiedenen Spalten, »entspricht« »Müller« im Feld »Name« und »ist größer oder gleich« als »50000« im Feld »Plz«).

Datumsfilter

Sie können auch einen Datumsfilter erstellen. Interessieren Sie nur Rechnungen, die zwischen dem 1.3.09 und dem 30.11.09 liegen, so verwenden Sie folgenden Filter für das Feld »Re-Datum« (Abbildung 13.25).



Das Datumsfeld muss dabei in einem der Datumsformate formatiert sein und die als Kriterien zu verwendenden Vergleichsdaten müssen Sie im exakt gleichen Format eingeben!

13.2.4 Spezialfilter

Der Befehl DATEN|FILTER|SPEZIALFILTER... ermöglicht nahezu beliebig komplexe Filterbedingungen, die die Möglichkeiten des erläuterten Dialogfelds bei weitem überschreiten.

Zusätzlich ermöglicht er, Datensätze zu selektieren und danach automatisch für eine spätere Weiterverwendung an eine beliebige andere Stelle in der Tabelle zu kopieren.

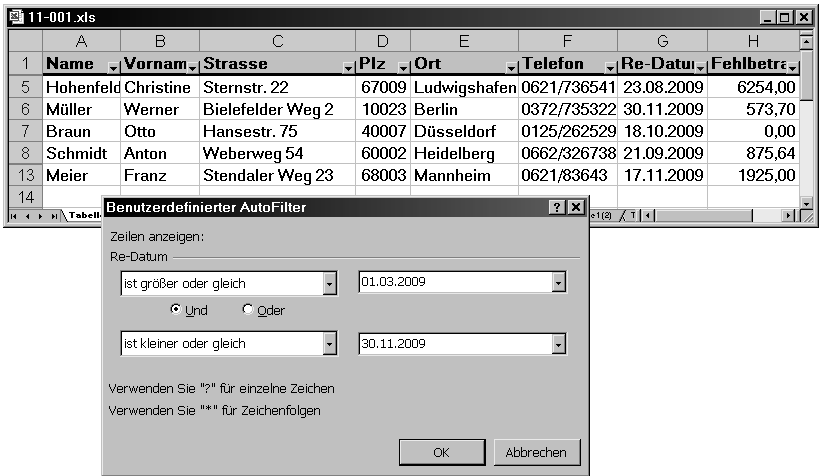


Abbildung 13.25:
Datumsselektion

Zur Anwendung dieses Befehls benötigen Sie einen Kriterienbereich. Dabei handelt es sich um einen Bereich, dessen oberste Zeile die Namen der zu filternden Felder enthält; und zwar mit exakt den gleichen Formatierungen wie in den Spaltenüberschriften! Darunter muss sich mindestens eine Zeile befinden, die Filterkriterien enthält.

Kriterienbereich

Ein Beispiel mit dem Kriterienbereich B15:C16 ist dieses (Abbildung 13.26):

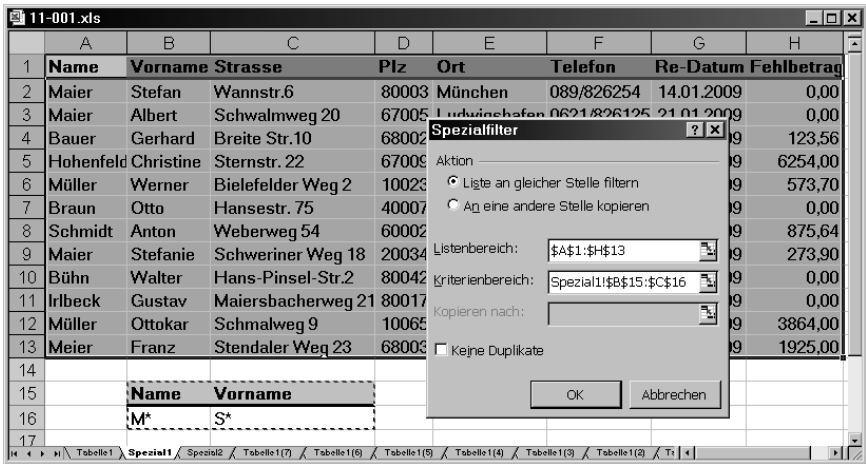


Abbildung 13.26:
Kriterienbereich

Wählen Sie DATEN|FILTER|SPEZIALFILTER..., aktivieren Sie *Listebereich* und zeigen Sie auf die von A1:H13 reichende Liste (einfacher: aktivieren Sie vor der Befehlswahl irgendeine Zelle der Liste: Excel erkennt die Liste und gibt den benötigten Bezug automatisch vor).

Aktivieren Sie anschließend *Kriterienbereich* und geben Sie (am einfachsten ebenfalls durch *Daraufzeigen*) den Bereich B15:C16 an, der die Suchkriterien enthält – *inklusive der zugehörigen Feldnamen!*

Die beiden Suchkriterien werden mit *UND* verknüpft. Also bedeutet dieser Kriterienbereich

Name="M*" UND Vorname="S*"

und zeigt nur Sätze an, deren Name mit »M« und deren Vorname mit »S« beginnt (Abbildung 13.27).

Abbildung 13.27:
Gefilterte Tabelle

	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.2009	0,00
9	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.02.2009	273,90
14								
15		Name	Vorname					
16		M*	S*					
17								



Das Ganze klappt nur unter einer sehr wichtigen Voraussetzung: Excel erkennt anhand der Überschriften der Kriterien die Zugehörigkeit zu den entsprechenden Spalten des Datenbereichs.

Zum Beispiel stellt das Suchkriterium »M*« mit der Überschrift »Name« ein Kriterium für das gleichnamige Feld »Name« des Datenbereichs dar.

Diese Erkennung klappt allerdings nur, wenn die Überschrift des Kriteriums mit der entsprechenden Spaltenüberschrift *absolut identisch* ist, also auch die gleichen Formatierungen besitzt.

Geben Sie die benötigten Kriterienüberschriften daher bitte nicht manuell ein, sondern kopieren Sie sie aus der Überschriftszeile des Datenbereichs!

Dass sich Excel an den Namen (und Formatierungen) der Kriterienüberschriften orientiert, besitzt den enormen Vorteil, dass sich die Kriterien wie in den vorhergehenden Abbildungen nicht in den gleichen Spalten wie die zugehörigen Listfelder befinden müssen, sondern ihre Position frei wählbar ist.

In der folgenden Abbildung werden ebenfalls mehrere Kriterien verwendet. Sie sind jedoch nicht in einer Zeile angeordnet, sondern befinden sich untereinander in mehreren Zeilen der gleichen Kriterienspalte (Abbildung 13.28).

Abbildung 13.28:
ODER-Verknüpfung

In diesem Fall werden alle Kriterien der betreffenden Spalte mit ODER verknüpft. Die untereinander angeordneten Kriterien »Berlin«, »Heidelberg« und »Düsseldorf« bedeuten daher, dass Excel nur Sätze anzeigt, die im Feld mit dem zugehörigen Namen »Ort« einen dieser drei Einträge enthalten (Abbildung 13.29).

Abbildung 13.29:
Ergebnis der
Filterung

Mit AutoFilter wäre diese Filterung nicht möglich gewesen, da Sie im zugehörigen Dialogfeld nur zwei Filterbedingungen miteinander verknüpfen können.

Wie das Beispiel zeigt, müssen Sie die Spaltenüberschriften nicht für jede neue Filterung ändern. Am einfachsten ist es, die komplette Kopfzeile der Liste zu kopieren und wie im Beispiel nur unter jenen Feldnamen Bedingungen einzugeben, die für die Filterung verwendet werden sollen. Alle übrigen Zellen lassen Sie einfach leer.

Bei der anschließenden Angabe des Kriterienbereichs können Sie dann wahlweise nur den tatsächlich interessanten Bereich angeben (im Beispiel E15:E18) oder den gesamten Kriterienbereich, im Beispiel A15:H18.





Passen Sie bei der Angabe des Kriterienbereichs dennoch auf: Enthält er eine Leerzeile, ist Excel der Ansicht, dass jeder Datensatz diesem Kriterium genügt, egal was er enthält – und zeigt alle Sätze an! Es findet also überhaupt keine Filterung statt.

Im Beispiel dürfte daher keinesfalls versehentlich als Kriterienbereich E15:E19 oder A15:H19 angegeben werden, da beide Bereiche unterhalb der eigentlichen Kriterien eine Leerzeile enthalten!

Selbstverständlich können Sie *UND*- und *ODER*-Verknüpfungen miteinander kombinieren. Ein Kriterienbereich mit den beiden Spalten

NameOrt
M* Heidelberg
M* München
M* Berlin

entspricht der Bedingung

(Name="M*" UND Ort="Heidelberg") ODER (Name="M*" UND Ort="München") ODER
(Name="M*" UND Ort="Berlin")

und zeigt nur jene Personen an, deren Name mit »M« beginnt und die in Heidelberg, München oder Berlin wohnen.



Suchkriterien werden im Kriterienbereich etwas anders verwendet als im entsprechenden Dialogfeld der AutoFilterung: Geben Sie einen Text wie »Müller« ein, werden auch ohne das Sonderzeichen »« alle Zeichenketten gefunden, die mit »Müller« beginnen, außer »Müller« also auch »Müllerfeld« oder »Müllerbach«.*

Passt Ihnen das nicht, veranlassen Sie Excel zu einer exakten Textsuche, indem Sie wie bei einer Formel zunächst ein Gleichheitszeichen eingeben. Ihm folgt Ihr ebenfalls mit einem Gleichheitszeichen beginnender Suchtext, eingeschlossen in Anführungszeichen:

=»Müller«

Dieses Suchkriterium findet nur »Müller«, aber nicht »Müllerfeld« oder »Müllerbach«.

Berechnete Kriterien

Eine berechnetes Kriterium ist eine Formel, die *WAHR* oder *FALSCH* liefert und unter anderem mindestens einen relativen Bezug auf ein Feld Ihrer Datenbank enthält. Ein Beispiel:

=(D2>=30000)

Der Bezug D2 bezieht sich auf das Feld »Plz« des ersten Datensatzes, der sich in Zeile 2 befindet. Excel interessiert sich nur für die Spaltenangabe D in diesem Bezug und durchsucht der Reihe nach die Inhalte des Felds »Plz« aller Datensätze. Der Bezug D2 steht somit stellvertretend für D2, D3, D4 und so weiter.

Die Formel testet der Reihe nach bei allen Sätzen, ob der Inhalt des Felds »Plz« größer oder gleich 30000 ist. Excel zeigt alle Sätze an, für die das zutrifft, für die diese Formel also **WAHR** ist.

Das ermöglicht beispielsweise folgende Anwendung: Nehmen wir an, Sie wollen alle Rechnungen sehen, bei denen seit mehr als 90 Tagen noch ein Fehlbetrag aussteht. Irgendwo in Ihrer Tabelle, sagen wir in C16, befindet sich die Formel

=JETZT().

Sie liefert das aktuelle Datum. Im Kriterienbereich verwenden Sie folgende Formel:

=(\$C\$16-G2)>90

Der Kriterienbereich ist A15:A16. Er besteht nur aus dieser Formel und der zugehörigen Überschrift (Abbildung 13.30).

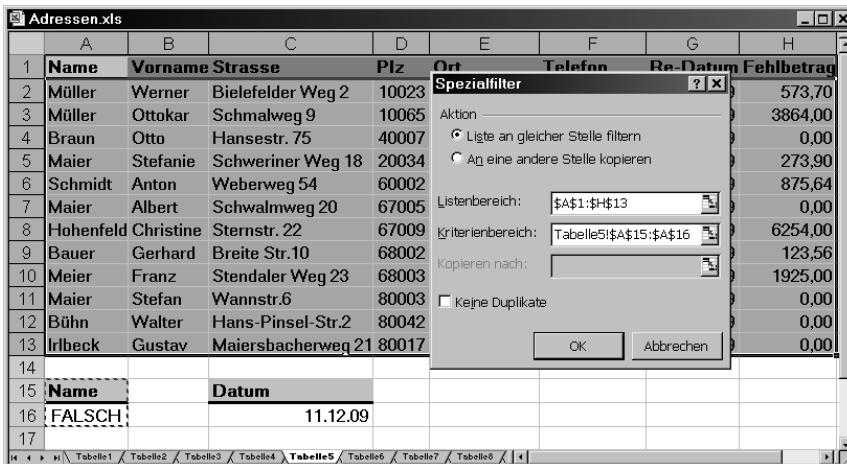


Abbildung 13.30:
Adressen.xls,
»Tabelle5«

Excel wird bei der Filterung den Bezug G2 der Reihe nach durch G3, G4 etc. ersetzen, jeweils die Differenz in Tagen zwischen dem in C16 enthaltenen Datum und dem momentan referenzierten Datum ermitteln und prüfen, ob die Differenz größer als 90 ist.

Falls ja, ist die Formel **WAHR** und der betreffende Datensatz wird angezeigt (Abbildung 13.31).

Abbildung 13.31:
Ergebnis der
Filterung

Adressen.xls								
	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
5	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.02.09	273,90
7	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826125	21.01.09	0,00
8	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.08.09	6254,00
9	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.02.09	123,56
11	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.09	0,00
12	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.09	0,00
13	Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/726384	01.02.09	0,00
14								
15	Name		Datum					
16	FALSCH		11.12.09					
17								



Für Bezüge auf Zellen außerhalb der Datenbank wie im Beispiel auf Zelle C16 müssen absolute Bezüge verwendet werden! Sonst würde Excel ebenso wie G2 auch diesen Bezug beim Durchlaufen der einzelnen Datensätze nacheinander durch C17, C18 etc. ersetzen!



Die abgebildete Tabelle ist »Tabelle5« der Arbeitsmappe ADRESSEN.XLS der Demodiskette. Tatsächlich befindet sich in Zelle C16 jedoch nicht die Formel =JETZT(), sondern einfach das Datum 11.12.2009. Ersetzen Sie es durch die erläuterte Formel, müssen Sie Ihre Rechneruhr vorübergehend auf das im Beispiel verwendete Datum 11.12.2009 setzen, um beim Nachvollziehen zum gleichen Ergebnis zu kommen!

Daten kopieren

Sollen die mit DATEN|FILTER|SPEZIALFILTER... ausgewählten Sätze kopiert werden, so aktivieren Sie die Option *An eine andere Stelle kopieren*, anschließend *Kopieren nach* und geben dort einen Bezug auf eine Zelle außerhalb der Datenbank an oder zeigen einfach darauf.

Nach OK bleibt die Liste selbst unverändert. Stattdessen werden die ausgewählten Sätze in den betreffenden Bereich kopiert, wobei die angegebene Zelle die linke obere Bereichsecke bildet.

Aktivieren Sie zusätzlich *Keine Duplikate*, so wird bei Sätzen, die mehrfach vorkommen, nur ein einziger in den Zielbereich kopiert.

13.3 Datenmasken

Excel kann in Tabellen enthaltene Listen als kleine Datenbank behandeln und darin Datensätze suchen, einfügen, löschen und ändern.

Die Liste muss die am Anfang dieses Kapitels erläuterte Struktur besitzen (ein Datensatz ist eine Zeile und in Spalten – die Felder – unterteilt und die erste Zeile enthält die Feldüberschriften).

Ist diese Voraussetzung gegeben, selektieren Sie irgendeine Zelle der Liste und wählen DATEN|MASKE... (Abbildung 13.32).

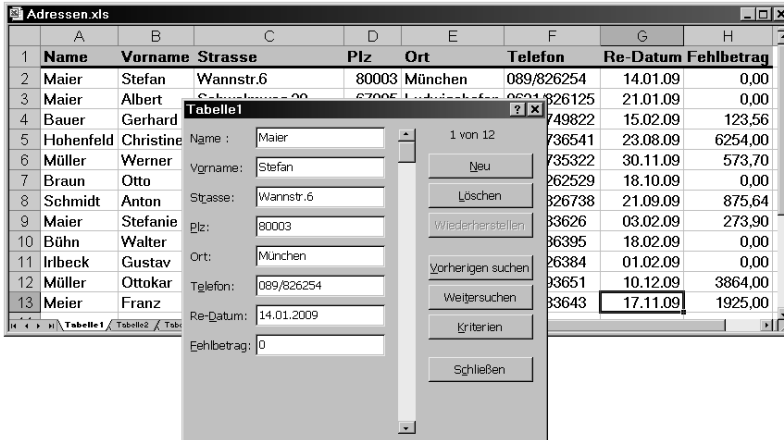


Abbildung 13.32:
Datenmaske

Excel zeigt eine Maske an, die zur vorgegebenen Liste passt. Darin ist zunächst der erste Datensatz zu sehen, der sich in Zeile 2 befindet. Zusätzlich wird Ihnen mitgeteilt, dass dies Satz Nummer »1 von 12«, die Liste also zwölf Sätze enthält.

Die restlichen Sätze können Sie sich ansehen, indem Sie die Liste mit *Vorherigen suchen* und *Weitersuchen* Satz für Satz durchblättern bzw. die Bildlaufpfeile benutzen.

Verschiedene Tastenkombinationen sind für größere Sprünge zuständig: **Bild ↓** (zehn Sätze weiter), **Bild ↑** (zehn Sätze zurück), **Strg + Bild ↑** (zum ersten Satz) und **Strg + Bild ↓** (zum ersten leeren Satz am Ende der Tabelle).

Sie ändern den aktuellen Satz, indem Sie einfach die entsprechenden Feld-einträge editieren. Die Änderungen werden wirksam, wenn Sie **↵** drücken oder einen Bildlauf zu einem anderen Satz durchführen.

Solange der momentan bearbeitete Satz angezeigt wird und die Änderungen noch nicht mit **↵** abgeschlossen wurden, können Sie sie mit *Wiederherstellen* wieder rückgängig machen.

Löschen löscht den angezeigten Satz.

Neu fügt am Ende der Datenbank einen neuen Satz ein: Excel gibt leere Maskenfelder vor, in die Sie den gewünschten Datensatz eingeben. **↵** oder Klicken auf *Neu* trägt den Satz ein und gibt anschließend erneut leere Felder vor, in die Sie einen weiteren Datensatz eingeben können.

Stattdessen können Sie auch bis zum letzten Satz plus eins vorblättern, also zum ersten noch nicht existierenden Datensatz, und ihn nun eingeben.

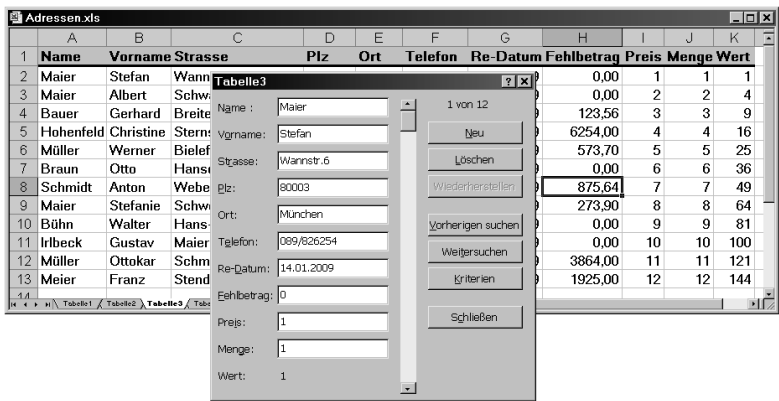
Berechnete Felder Außer Feldern, in die Sie die Daten selbst eintragen müssen, kennt Excel zusätzlich berechnete Felder, deren Werte auf dem Inhalt anderer Felder basieren.

Zum Beispiel könnte unsere Bestelltabelle zwei zusätzliche Spalten I und J mit den Überschriften »Preis« und »Menge« enthalten, in denen Sie bei jeder Bestellung eintragen, was ein Stück des bestellten Artikels kostet und wie viele davon bestellt wurden.

Spalte K könnte ein zusätzliches berechnetes Feld »Wert« oder »Gesamtpreis« enthalten, mit der Formel $=I2 * J2$ in Zelle K2, der Formel $=I3 * J3$ in Zelle K3 und so weiter.

In der Maske wird der Inhalt dieses Felds nur angezeigt, Editierungen sind – sinnvollerweise – nicht möglich (Abbildung 13.33).

Abbildung 13.33:
Berechnete Felder



Ändern Sie eine der Zellen, von denen das berechnete Feld abhängt, so wird sein Wert automatisch entsprechend aktualisiert.

Tragen Sie einen neuen Datensatz ein, so wird die im berechneten Feld enthaltene Formel auch in diesen Satz kopiert – was bedeutet, dass Sie für berechnete Felder nur relative Bezüge verwenden dürfen!

Suchkriterien Kriterien löscht die angezeigten Felder und ermöglicht die Eingabe von Suchkriterien. Dabei können Sie die im vorhergehenden Kapitel behandelten Suchmöglichkeiten mit »*« und »?« nutzen (Abbildung 13.34).

Im einfachsten Fall geben Sie entsprechend der Abbildung in einem Feld wie »Name« ein Kriterium wie »Maier« ein.

Abbildung 13.34:
Suchkriterien
definieren

Löschen löscht die eingegebenen Suchkriterien.

 oder *Maske* beendet die Definition der Suchkriterien. Anschließend können Sie mit den Bildlaufpfeilen weiterhin die gesamte Liste durchblättern.

Vorherigen suchen und *Weitersuchen* zeigt nun jedoch nicht mehr den physisch nächsten bzw. vorhergehenden Satz an, sondern den nächsten/vorhergehenden, der *Ihren Suchkriterien entspricht*.

13.4 Gruppierte Teilergebnisse

DATEN|TEILERGEBNISSE... ermöglicht es, sortierte Listen nach dem Inhalt eines beliebigen Feldes zu gruppieren.

Sie können *für jede Gruppe getrennt* Auswertungen vornehmen und beispielsweise die Summe oder den Durchschnitt der in einem beliebigen numerischen Feld enthaltenen Werte ermitteln lassen.

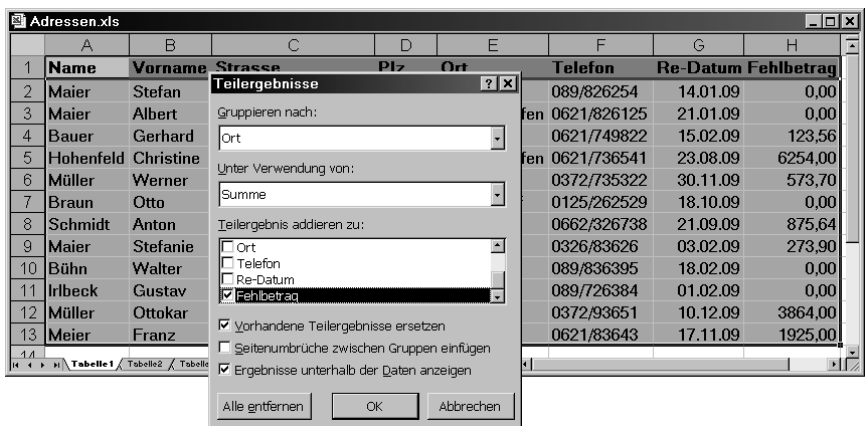
Nehmen wir an, Sie interessieren sich für die Summe aller Fehlbeträge Ihrer Kunden, unterteilt nach Wohnort: Sie wollen wissen, wie viel Ihnen die Kunden in Berlin schulden, wie viel die Kunden in Ludwigshafen und so weiter.

Zunächst müssen Sie die Liste nach dem Inhalt jener Spalte sortieren, die zum Gruppieren verwendet werden soll, im Beispiel nach »Ort«. Dieses Sortieren ist eine unbedingt notwendige Vorbereitung, die Sie keinesfalls auslassen dürfen!



Anschließend selektieren Sie irgendeine Zelle der Liste und wählen DATEN|TEILERGEBNISSE... (Abbildung 13.35).

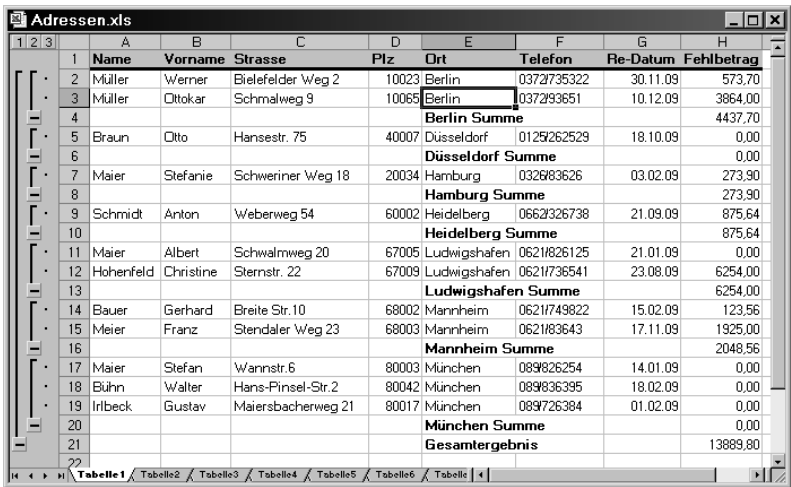
Abbildung 13.35:
Sortierte
Liste gruppieren



Unter *Gruppieren nach* selektieren Sie nun die Spalte »Ort« und unter *Unter Verwendung von* die anzuwendende Funktion, beispielsweise »Summe«.

Unter *Teilergebnis addieren zu* wählen Sie jenes Feld bzw. jene Felder aus, für das bzw. die Sie Teilergebnisse wünschen, im Beispiel also das Feld »Fehlbetrag« (Abbildung 13.36).

Abbildung 13.36:
Resultierende Teil-
ergebnisse



Alle Datensätze, die im Feld »Ort« den gleichen Inhalt besitzen, bilden eine Gruppe. Unterhalb jeder Gruppe fügt Excel eine zusätzliche Zeile ein, die die Summe der im Feld »Fehlbetrag« enthaltenen Werte der betreffenden Datensätze enthält.

Sie sehen daher auf einen Blick, dass Ihnen die Kunden in Berlin noch 4437,70 DM schulden, die Kunden in Ludwigshafen noch 6254 DM etc.

In der untersten Zeile befindet sich das Gesamtergebnis, die Summe aller Teilergebnisse.

Die automatisch eingesetzte Gliederungsfunktion zeigt auf der nächsttieferen Ebene nur die Teilergebnisse an und blendet die Details – die zugehörigen Datensätze – aus (Abbildung 13.37).

Gliederung

	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
4					Berlin	Summe		4437,70
6					Düsseldorf	Summe		0,00
8					Hamburg	Summe		273,90
10					Heidelberg	Summe		875,64
13					Ludwigshafen	Summe		6254,00
16					Mannheim	Summe		2048,56
20					München	Summe		0,00
21					Gesamtergebnis			13889,80

Abbildung 13.37:
Nur Teilergebnisse anzeigen

Auf der tiefsten Ebene wird nur noch das Gesamtergebnis angezeigt.

Interessiert Sie anschließend eine andere Auswertung, zum Beispiel nicht die Summe, sondern der Mittelwert der Fehlbeträge, erneut nach »Ort« gruppiert, müssen Sie den gesamten Vorgang nicht rückgängig machen.

Stattdessen selektieren Sie wieder eine beliebige Zelle der Liste, wählen erneut DATEN|TEILERGEBNISSE... und selektieren als gewünschtes Ergebnis diesmal »Mittelwert« (Abbildung 13.38).

Abbildung 13.38:
Gruppierte Mittelwerte

Mehrere
Teilergebnisse

Deaktivieren Sie zuvor *Vorhandene Teilergebnisse ersetzen*, so werden die alten Teilergebnisse nicht ersetzt, sondern die neuen Teilergebnisse zusätzlich eingefügt. Die Tabelle enthält danach beide Teilergebnisse, die gruppierten Mittelwerte und zusätzlich die zuvor gebildeten gruppierten Summen (Abbildung 13.39).

Abbildung 13.39:
Alte Teilergebnisse
behalten

Adressen.xls								
1	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372735322	30.11.09	573,70
3	Müller	Ottokar	Schmalweg 9	10065	Berlin	037293651	10.12.09	3864,00
4					Berlin Mittelwert			2218,85
5					Berlin Summe			4437,70
6	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125262529	18.10.09	0,00
7					Düsseldorf Mittelwert			0,00
8					Düsseldorf Summe			0,00
9	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	032983626	03.02.09	273,90
10					Hamburg Mittelwert			273,90
11					Hamburg Summe			273,90
12	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662326738	21.09.09	875,64
13					Heidelberg Mittelwert			875,64
14					Heidelberg Summe			875,64
15	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621826125	21.01.09	0,00
16	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621736541	23.08.09	6254,00
17					Ludwigshafen Mittelwert			3127,00
18					Ludwigshafen Summe			6254,00
19	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621749822	15.02.09	123,56
20	Meier	Franz	Stendaler Weg 23	68003	Mannheim	062183643	17.11.09	1925,00
21					Mannheim Mittelwert			1024,28
22					Mannheim Summe			2048,56
23	Maier	Stefan	Wannstr.6	80003	München	089826254	14.01.09	0,00
24	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089836395	18.02.09	0,00
25	Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089726384	01.02.09	0,00
26					München Mittelwert			0,00
27					München Summe			0,00
28					Gesamtmittelwert			1157,48
29					Gesamtergebnis			13889,80
30								

Gruppierung
wechseln

Interessiert Sie nun ein anderes Gruppierungskriterium, wollen Sie die Liste beispielsweise nach »Plz« gruppieren, so wählen Sie zunächst DATEN|SORTIEREN... und sortieren nach »Plz«.

Die eingefügten Teilergebnisse werden dabei entfernt: Sie haben wieder Ihre normale Liste vor sich, diesmal jedoch nach »Plz« sortiert. Nun wählen Sie DATEN|TEILERGEBNISSE... und unter *Gruppieren nach* die Spalte »Plz«.

Optionen

Seitenumbrüche zwischen Gruppen einfügen fügt vor jeder Gruppe eine Seitenumbruchlinie ein, sodass jede Gruppe anschließend beim Drucken auf einer eigenen Seite ausgegeben wird.

Deaktivieren Sie *Ergebnisse unterhalb der Daten anzeigen*, so werden die Teilergebniszeilen oberhalb anstatt unterhalb der einzelnen Gruppen eingefügt.

Um die Teilergebnisse aufzuheben, wählen Sie erneut DATEN|TEILERGEBNISSE... und aktivieren die Schaltfläche *Alle Entfernen*.

14 Microsoft Query – Zugriff auf externe Datenbanken

Im Lieferumfang von Excel befindet sich MS Query. Query ermöglicht Ihnen, mit Hilfe von Abfragen in absolut einheitlicher Art und Weise auf unterschiedlichste Datenbanken zuzugreifen, auf dBase-, FoxPro-, Access-Datenbanken und viele weitere. Zusätzlich können Sie mit Query eigene Datenbanken in jedem dieser Formate erstellen und bearbeiten.

Das folgende Kapitel stellt einen Schnellkursus dar, der für Datenbank-Laien möglicherweise etwas zu schnell und daher leicht ungenießbar ist. Leider geht es nicht anders. Bedenken Sie, dass Query eine Art Mini-Microsoft Access ist – und dass mein Kompendium über dieses Programm über 1000 Seiten stark ist!

Ich empfehle Ihnen daher dieses Access-Kompendium, wenn Sie mehr über Access und damit gleichzeitig über Query wissen wollen, beispielsweise über den Aufbau von Datenbanken und den Entwurf von Abfragen.

14.1 Die Funktionsweise von Query

Installierten Sie Excel vollständig, wurde auch Microsoft Query installiert, ein eigenständiges und von Excel völlig unabhängig einsetzbares Programm.



Ich erläutere Query an dieser Stelle, weil es mit dem zuvor behandelten Thema »Listen und Datenbanken« sehr eng verwandt ist. Query ist in der Lage, Datenbanken in verschiedensten Formaten zu verwalten. Wie gesagt, es ermöglicht Ihnen, mithilfe von Abfragen in absolut einheitlicher Art und Weise auf die unterschiedlichsten Datenbanken zuzugreifen, auf dBase-, FoxPro-, Access-Datenbanken und viele weitere.

Zusätzlich können Sie mit Query eigene Datenbanken in jedem dieser Formate erstellen und bearbeiten. Das Originalprogramm – beispielsweise dBase – kann darauf später zugreifen, als wäre die Datenbank mit dem betreffenden Programm selbst erstellt worden.

Dahinter steckt das Schlüsselwort ODBC, ein Standard, den Microsoft erstmals mit Access 1.0 einführte.

ODBC

Unter ODBC ist eine genormte Schnittstelle zum Zugriff auf Datenbanken verschiedenster Formate zu verstehen, die folgendermaßen funktioniert: Sie arbeiten mit einem Programm wie Query, das ODBC unterstützt. Damit wollen Sie irgendwelche Daten in einer bestehenden dBase-Datenbank selektieren oder neue Daten darin speichern.

Query reicht diese Aufgabe an den ODBC-Server weiter, eine Windows-DLL (dynamische Link-Library), die nicht in Query selbst eingebaut ist, sondern verschiedensten Programmen zur Verfügung steht und ihnen den Zugriff auf Fremddatenbanken ermöglicht.

Der ODBC-Server wiederum reicht die Aufgabe an einen von mehreren spezialisierten ODBC-Treibern weiter. Ein ODBC-Treiber ist ein Programm in Form einer DLL, das mit genau einer Datenbank umgehen kann (der dBase-ODBC-Treiber beispielsweise mit dBase) und weiß, wie Daten in einer Datenbank des betreffenden Formats selektiert oder darin gespeichert werden.

Hat der Treiber seine Aufgabe erledigt, beispielsweise einige Adressen in einer dBase-Datenbank selektiert, reicht er das Resultat an den ODBC-Server weiter. Dieser übergibt es wiederum der Anwendung, mit der Sie gerade arbeiten, beispielsweise Query.

Das Ergebnis: Mit Query benutzen Sie eine dBase-Datenbank genauso wie eine FoxPro- oder Access-Datenbank, ohne den geringsten Unterschied. Sie merken nicht einmal, mit welcher Art von Datenbank Sie es zu tun haben!

Die einzige Voraussetzung für diesen einheitlichen Zugriff auf unterschiedlichste Datenbanken ist das Vorhandensein des ODBC-Servers und der benötigten ODBC-Treiber.

Diese Treiber sind möglicherweise bereits seit längerem bei Ihnen. Beispielsweise, wenn Sie irgendwann zuvor Access oder eine andere ODBC-unterstützende Anwendung installierten, die Treiber mitbrachte und bei der Programminstallation ebenfalls installierte.

Sie sind jedoch spätestens dann vorhanden, wenn Sie Excel *vollständig* installierten. Wenn nicht, sollten Sie das Installationsprogramm erneut aufrufen, um nachträglich die darauf enthaltenen ODBC-Treiber zu installieren – oder zumindest jene, die Sie benötigen.

Nach dem Öffnen des Add-Ins XLODBC.XLA können Sie übrigens von VBA aus auch ohne Query mit SQL-Statements auf externe Datenbanken zugreifen.



Dieses Add-In erweitert Excel um verschiedene Funktionen (die Namen dieser Funktionen beginnen jeweils mit *SQL...*), denen Sie das jeweils benötigte SQL-Statements als Argument übergeben.

Es gibt mehrere Möglichkeiten, Query aufzurufen. Zunächst einmal können Sie Query durch Aufruf der zugehörigen Programmdatei MSQRY32.EXE starten (im Verzeichnis PROGRAMME\MICROSOFT OFFICE\OFFICE, falls Office im Standardverzeichnis MICROSOFT OFFICE installiert wurde). Benutzen Sie Query häufiger, erstellen Sie einfach eine Verknüpfung zu dieser Datei auf dem Desktop.

Query aufrufen

Query in Verbindung mit Excel zu nutzen ist noch einfacher, da Excel optimal auf die Zusammenarbeit mit Query eingerichtet ist. Voraussetzung ist jedoch, dass das Add-In »Microsoft Query-Add-In« eingebunden ist.

Mit Hilfe dieses Add-Ins ist Excel in der Lage, Query selbst aufzurufen. Sie können anschließend in Query Abfragen an Datenbanken verschiedenster Formate stellen und die Ergebnisse an Excel übergeben, beispielsweise in Form der in Kürze erläuterten Pivot-Tabellen (siehe Kapitel 16.8, »Zugriff auf externe Datenquellen«).



Sie haben folgende Möglichkeiten, Query aus Excel heraus aufzurufen:

- ➔ Entweder wählen Sie DATEN|PIVOTTABLE- UND PIVOTCHART-BERICHT... und im zugehörigen Dialogfeld die Option *externer Datenquelle*. Im folgenden Schritt klicken Sie auf die Schaltfläche *Daten importieren...*, um Query zu starten.
- ➔ Oder aber – wenn Sie keine der bisher unbekannten Pivot-Tabellen erzeugen wollen – Sie wählen DATEN|EXTERNE DATEN IMPORTIEREN|NEUE ABFRAGE ERSTELLEN..., um Query zu starten. Beenden Sie Query, so werden die selektierten Daten automatisch in einer ganz normalen Excel-Tabelle gespeichert.

14.2 Externe Daten in Excel importieren

Um die folgenden Schritte nachzuvollziehen, kopieren Sie bitte alle dBase-Dateien der Begleit-CD (alle Dateien mit der Endung DBF) ins Verzeichnis C:\EIGENE DATEIEN!



Eine kleine Warnung: Ich zeige Ihnen nun, wie eine Datenquelle zum Zugriff auf dBase-Dateien definiert und danach benutzt wird. Diese Erläuterungen sind jedoch nicht unverändert auf andere Datenquellen übertragbar, da sich die einzelnen Treiber nicht hundertprozentig identisch verhalten!



Definieren Sie beispielsweise eine Datenquelle zum Zugriff auf Access-Datenbanken, so können Sie sich anschließend die zu verwendende Access-Datenbank völlig uneingeschränkt aussuchen, egal in welchem Verzeichnis sie sich befindet – was mit dem dBase-Treiber prinzipiell zwar ebenfalls möglich ist, jedoch Klimmzüge und das Arbeiten ohne den nachfolgend beschriebenen Query-Assistenten erfordert.

Egal ob Sie eine neue Tabelle (=Datenbank) erzeugen oder mithilfe einer Abfrage Daten aus einer bereits bestehenden Tabelle auswählen wollen, in jedem Fall müssen Sie Query bekannt geben, welche Datenquelle Sie dazu verwenden wollen.

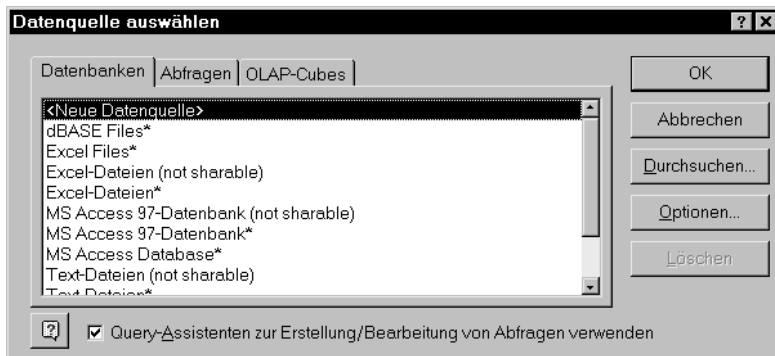
Unter einer Datenquelle ist die Kombination aus dem *Ort* und dem *Typ* der Daten zu verstehen, auf die Sie zugreifen wollen: also aus einem Ordner, der Datenbanken eines bestimmten Typs enthält, beispielsweise der Ordner C:\EIGENE DATEIEN, und einem Dateityp, beispielsweise »dBase IV-Dateien«.

Nach der Definition dieser Datenquelle können Sie in diesem Ordner gespeicherte dBase-Datenbanken benutzen oder neu erzeugen und Abfragen erstellen, die auf die in diesen Datenbanken enthaltenen Daten zugreifen (und sie gegebenenfalls an Excel übergeben).



Rufen Sie Query nun bitte von Excel aus auf, beispielsweise mit DATEN|EXTERNE DATEN IMPORTIEREN|NEUE ABFRAGE ERSTELLEN... . Ebenso wie beim Aufruf von Query als eigenständigem Programm und folgender Wahl von DATEI|NEU... oder DATEI|TABELLENDEFINITION... erscheint zunächst das Datenquelle auswählen-Dialogfeld, in dem Sie die zu verwendende *Datenquelle auswählen* müssen (Abbildung 14.1).

Abbildung 14.1:
Datenquelle
auswählen



Möglicherweise ist das Dialogfeld nahezu leer, weil noch keine Datenquellen definiert wurden. In jedem Fall ist jedoch der Eintrag »<Neue Datenquelle>« vorhanden, der Ihnen die Definition einer neuen Datenquelle ermöglicht. Wählen Sie ihn aus und klicken Sie auf OK.

Im folgenden Dialogfeld geben Sie der neuen Datenquelle einen Namen wie »Test« und selektieren im Listenfeld darunter den Treiber, der das gewünschte Format unterstützt, beispielsweise »Microsoft dBase Treiber (*.dbf)« (Abbildung 14.2).

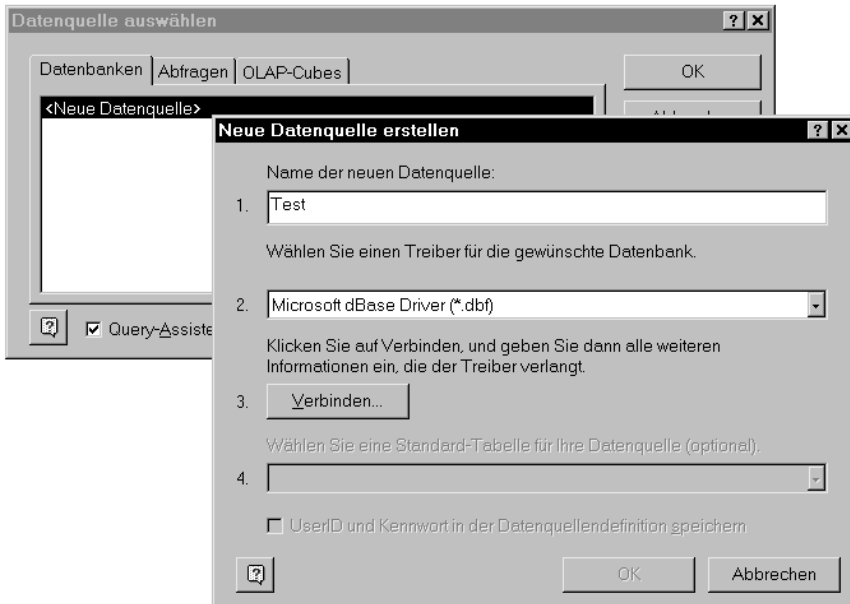


Abbildung 14.2:
ODBC-Treiber

Wollen Sie auf bereits existierende Datenbanken zugreifen, müssen Sie den Treiber selektieren, der das gewünschte Format unterstützt.

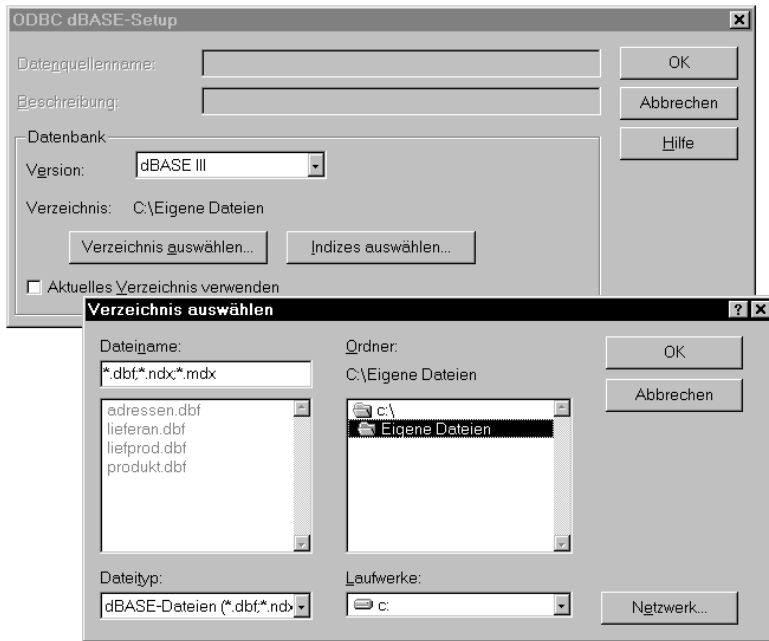
Wollen Sie mit Query eine neue Datenbank erstellen, ist es eigentlich gleichgültig, welchen Treiber Sie selektieren. Sind Sie an möglichst flexiblem Datenaustausch interessiert, wäre dBase sicher keine schlechte Wahl, da das dBase-Format nach wie vor einen Standard darstellt, den alle möglichen Programme kennen.

Anschließend klicken Sie auf die Schaltfläche *Verbinden...*, um die Informationen anzugeben, die der betreffende Treiber benötigt (Abbildung 14.3).

Im Fall des dBase-Treibers müssen Sie nur angeben, um welche dBase-Version es sich handelt, beispielsweise dBASE III, und mit welchem Verzeichnis der Treiber verbunden sein soll.

Ist das Kontrollkästchen *Aktuelles Verzeichnis verwenden* aktiviert, ist das das aktuelle Verzeichnis.

Abbildung 14.3:
ODBC-dBase-Setup



Deaktivieren Sie das Kontrollkästchen, so können Sie anschließend auf *Verzeichnis auswählen...* klicken und im daraufhin erscheinenden *Dateiauswahl*-Dialogfeld jenen Ordner selektieren, in dem sich die Sie interessierenden dBase-Datenbanken befinden bzw. in dem Sie neue dBase-Datenbanken anlegen wollen. Wählen Sie nun bitte das Verzeichnis C:\EIGENE DATEIEN aus.

Wählen Sie danach in diesem und in den beiden übergeordneten Dialogfeldern jeweils OK. Das Listenfeld des *Datenquelle auswählen*-Dialogfelds enthält die soeben definierte Datenquelle »Test«, die einen einfachen Zugriff auf alle dBase-Datenbanken im Ordner C:\EIGENE DATEIEN ermöglicht (Abbildung 14.4).

Nach Auswahl der soeben definierten Datenquelle »Test« meldet sich der Query-Assistent, der zunächst wissen will, welche Spalten die neue Abfrage enthalten soll.

Jede Spalte entspricht einem Feld einer Datenbanktabelle. Im linken Listenfeld werden alle Tabellen vorgegeben, die im Verzeichnis vorhanden sind, mit dem die Datenquelle verbunden ist, im Beispiel also alle dBase-Dateien, die sich im Verzeichnis C:\EIGENE DATEIEN befinden. Selektieren Sie eine dieser Tabellen, beispielsweise ADRESSEN.DBF, und klicken Sie auf >, dann werden alle Felder, die die Tabelle enthält, ins rechte Listenfeld befördert (Abbildung 14.5).

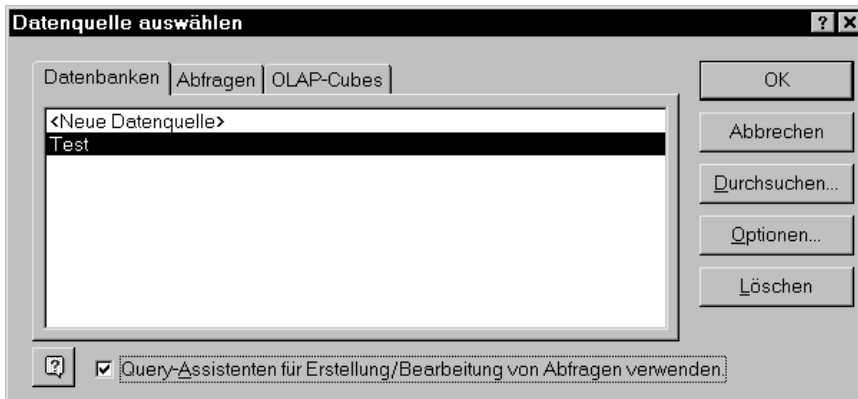


Abbildung 14.4:
Neue Datenquelle

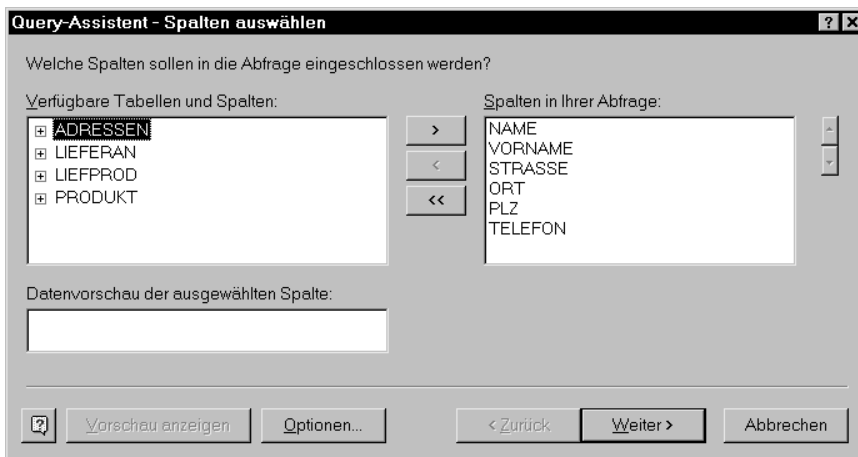


Abbildung 14.5:
Für Abfrage zu
verwendende
Tabelle

Analog dazu können Sie auch die Felder einer weiteren Tabelle in die neue Abfrage befördern. Interessieren Sie nicht alle Felder einer Tabelle, so klicken Sie auf das »+«-Zeichen der betreffenden Tabelle, um im linken Listenfeld unter der Tabelle die darin enthaltenen Felder einzublenden.



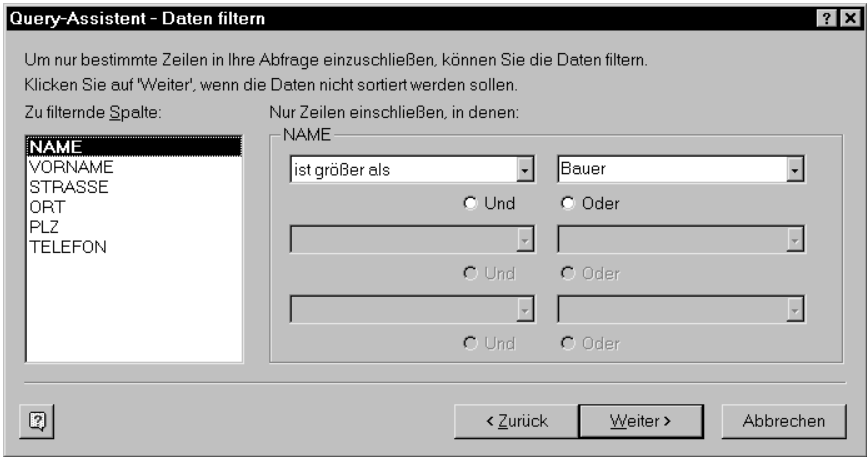
Anschließend können Sie statt der gesamten Tabelle einzelne Felder selektieren, um nur diese nach rechts zu verschieben.

Im nächsten Schritt können Sie für jedes einzelne Feld der Abfrage Filterbedingungen festlegen, um nur bestimmte Daten der betreffenden Spalte anzeigen zu lassen (Abbildung 14.6).

Sie kennen das Prinzip bereits aus dem Kapitel 13.2.3, »Filterkriterien«: Sie klicken auf ein Feld wie »Name«, wählen im linken Listenfeld eine Vergleichsbedingung wie »ist größer als« und im rechten einen Vergleichswert wie »Maier«, um nur Namen anzeigen zu lassen, die alphabetisch größer als »Bauer« sind.



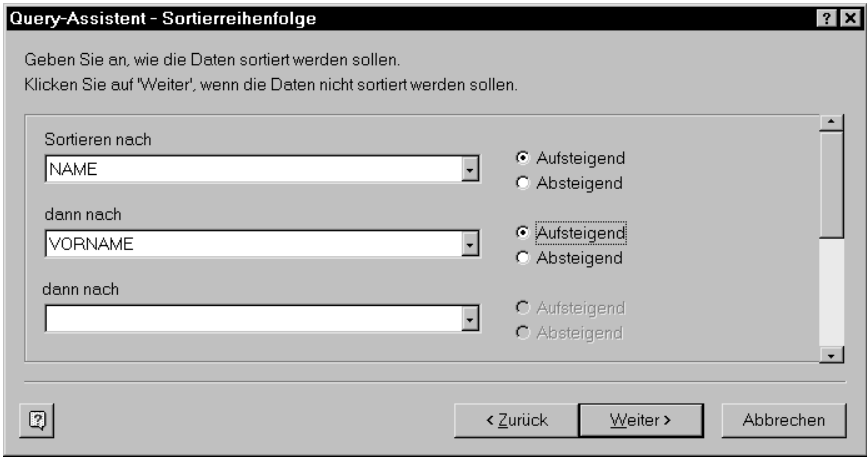
Abbildung 14.6:
Filterkriterien



Anschließend legen Sie für beliebige weitere Felder Kriterien fest und wählen, ob die einzelnen Kriterien mit *UND* oder mit *ODER* verknüpft werden sollen.

Danach legen Sie die Sortierung der Daten fest, ein Schritt, der ebenfalls nichts Neues bietet (Abbildung 14.7).

Abbildung 14.7:
Sortierung



Im letzten Schritt können Sie die Abfrage mit der Schaltfläche *Abfrage speichern...* unter einem beliebigen Dateinamen (Endung DQY) speichern, um sie bei Bedarf jederzeit wieder laden oder editieren zu können, ohne die vorherigen Schritte erneut durchlaufen zu müssen (Abbildung 14.8).

Speichern Sie die Abfrage bitte unter einem Namen wie »Abfrage von Test.dqy« im vorgegebenen Verzeichnis QUERIES, da sie in Kürze erneut benötigt wird.

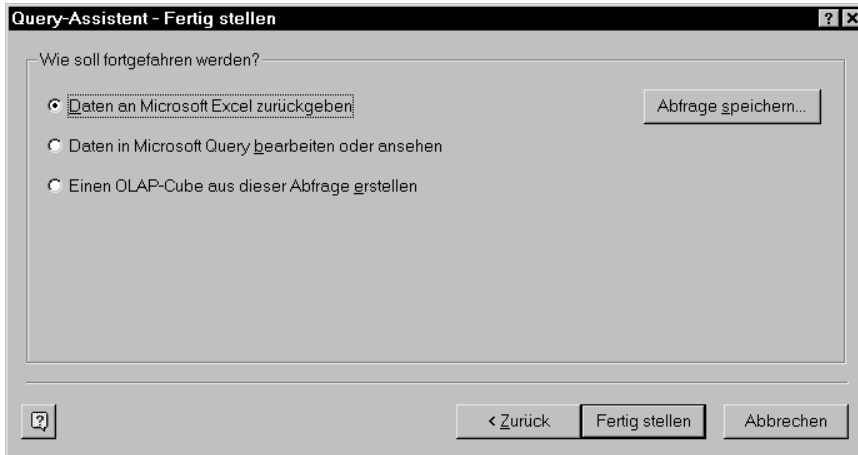


Abbildung 14.8:
Letzter Schritt

Vor allem jedoch wählen Sie in diesem Schritt, ob

- ➔ Sie die Abfrage in Query weiterbearbeiten wollen: dann können Sie die erzeugte Abfrage im Abfragefenster nachträglich manipulieren (siehe Kapitel 14.4, »Abfragen«),
- ➔ oder die definierten Daten an Excel zurückgegeben werden sollen: dann wird Query geschlossen und die ausgewählten und sortierten Daten werden in eine Excel-Tabelle importiert. Zuvor werden Sie noch gefragt, wohin die Daten importiert werden sollen (Abbildung 14.9).

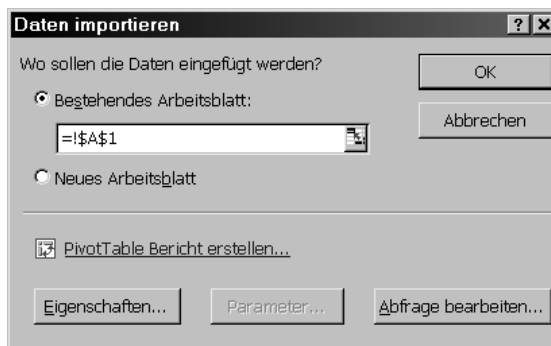


Abbildung 14.9:
Dialogfeld zum
Datenimport

Überlegen Sie es sich anders, können Sie den Übergabeprozess mit *Abfrage bearbeiten* jetzt noch abbrechen und die Daten editieren.

Ansonsten können Sie mit *PivotTable Bericht erstellen...* den in Kapitel 16, »Pivot-Tabellen«, erläuterten Pivot-Tabellen-Assistenten aufrufen, um aus den Daten eine Pivot-Tabelle zu erzeugen.



Um die externen Daten stattdessen einfach in eine Excel-Tabelle einzufügen, wählen Sie nun aus, in welches Blatt sie eingefügt werden sollen; handelt es sich um das aktuelle Blatt, geben Sie zusätzlich die linke Ecke des Bereichs an, in den die Daten dort eingefügt werden sollen. OK fügt die Daten ein und blendet automatisch die Symbolleiste *Externe Daten* ein (Abbildung 14.10).

Abbildung 14.10:
Importierte
Abfragedaten

	A	B	C	D	E	F
1	NAME	VORNAME	STRASSE	ORT	PLZ	TELEFON
2	Maier	Anton	Waldstr.15	Hamburg	70008	10373/9373
3	Maier	Willi	Breite Str.5	München	80003	73270/98364
4						

Klicken Sie vor dem Einfügen auf die Schaltfläche *Eigenschaften...*, so können Sie die Art und Weise näher beeinflussen, in der Excel die Daten einfügt.

Aktivieren Sie beispielsweise im zugehörigen Dialogfeld *Abfragedefinition speichern*, dann werden außer den Daten selbst zusätzlich alle Informationen über die zugehörige Abfrage in der Tabelle gespeichert – wovon Sie allerdings nichts sehen werden.

*Daten
aktualisieren*

Dafür können Sie die eingefügten Daten nun jederzeit aktualisieren, indem Sie eine beliebige Zelle im eingefügten Bereich selektieren und den nur dann verfügbaren Befehl DATEN|DATEN AKTUALISIEREN wählen.

Dieser Befehl führt die Abfrage erneut aus und importiert die aktualisierten Daten, die nun in der Tabelle die zuvor darin enthaltenen alten Daten ersetzen.

Weitere Möglichkeiten sind beispielsweise die Aktivierung des Kontrollkästchens *Feldnamen einschließen*, um die Feldnamen »Name«, »Vorname« etc. als Spaltenüberschriften in die erste Tabelle zu übernehmen; oder von *Zeilennummern einschließen*, um ganz links eine zusätzliche Spalte mit fortlaufenden Zeilennummern einzufügen, praktisch Datensatznummern.

*Datenbereichs-
Eigenschaften*

Sie können diese Eigenschaften des importierten Datenbereichs jederzeit nachträglich verändern, indem Sie erneut eine beliebige Zelle im eingefügten Bereich selektieren und den nur dann verfügbaren Befehl DATEN|EXTERNE DATEN IMPORTIEREN|DATENBEREICH-EIGENSCHAFTEN wählen.

*Abfrage
bearbeiten*

Wählen Sie stattdessen DATEN|EXTERNE DATEN IMPORTIEREN|ABFRAGE BEARBEITEN..., so erscheint erneut der Assistent, Sie können die Abfrage verändern und das geänderte Resultat mit DATEI|DATEN AN MICROSOFT EXCEL ZURÜCKGEBEN an Excel übergeben.

Wie erläutert können Sie einmal erstellte Abfragen unter beliebigen Namen speichern. Der Excel-Befehl DATEN|EXTERNE DATEN IMPORTIEREN|DATEN IMPORTIEREN... öffnet ein Dialogfeld, in dem Sie eine gespeicherte Abfrage auswählen und erneut von Query ausführen lassen können (Abbildung 14.11).

Abfrage ausführen

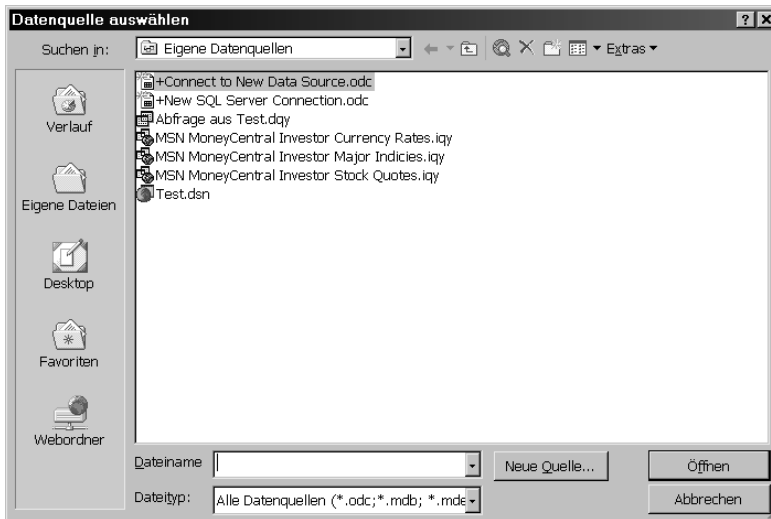


Abbildung 14.11:
Abfrage ausführen

Zusätzlich können Sie in diesem Dialogfeld unterschiedlichste Datenquellen öffnen bzw. definieren:

- ➔ Sie können die von Ihnen gespeicherten Query-Abfragen (Endung DQY) ausführen.
- ➔ Sie können gespeicherte Webabfragen (Endung IQY) ausführen, die Sie mit dem Befehl DATEN|EXTERNE DATEN IMPORTIEREN|NEUE WEBABFRAGE... erstellt (siehe Kapitel 28, »Webabfragen«).
- ➔ Sie können auf eine bereits definierte Datenquelle klicken (Endung DSN), um zugehörige Daten mit Query zu bearbeiten.
- ➔ Außerdem können Sie mithilfe des zugehörigen Assistenten eine neue Datenquelle definieren, indem Sie auf eine der Dateien mit der Endung ODC klicken oder *Neue Quelle...* wählen.

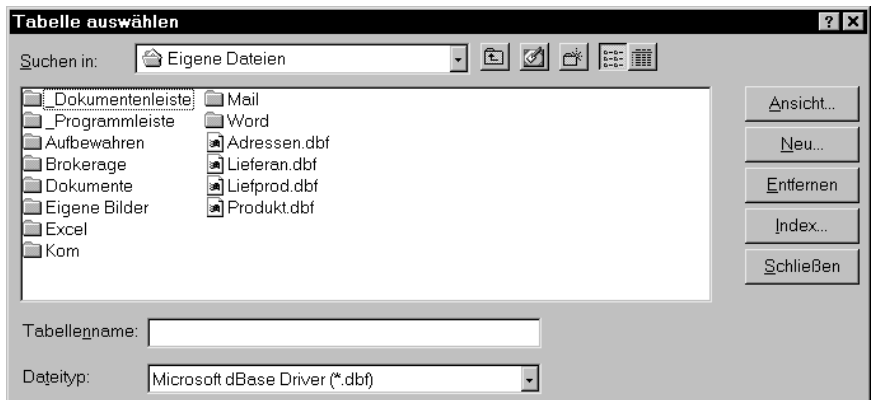
14.3 Datenbanken erstellen/verändern

Statt nur vorhandene Datenbanken zu benutzen, können Sie mit Query auch neue Datenbanken in beliebigen Formaten erstellen. Dazu rufen Sie Query als eigenständiges Programm auf (Doppelklick auf Programmdatei MSQRY32.EXE im Verzeichnis PROGRAMME\MICROSOFT OFFICE\OFFICE) und wählen DATEI|TABELLENDEFINITION....

Im daraufhin erscheinenden *Datenquelle auswählen*-Dialogfeld selektieren Sie eine passende Datenquelle, beispielsweise die zuvor definierte Datenquelle »Test«, wenn Sie eine dBase-Datenbank erstellen wollen.

Ein *Dateiselektions*-Dialogfeld erscheint, in dem der mit der Datenquelle verbundene Ordner vorgegeben ist. (Abbildung 14.12).

Abbildung 14.12:
Tabelle auswählen



Tabellendefinition

Sie können nun eine in diesem oder in einem beliebigen anderen Ordner enthaltene dBase-Tabelle selektieren und sich mit *Ansicht...* die Struktur der Tabelle anschauen oder verändern.

Um stattdessen eine neue dBase-Tabelle zu erzeugen, aktivieren Sie *Neu...* (Abbildung 14.13).

Zunächst tippen Sie einen Tabellennamen wie »Adressen« oder besser – ADRESSEN.DBF gibt es in diesem Verzeichnis bereits – »Anschrift« ein.

Danach definieren Sie die einzelnen Tabellenfelder: Sie geben den ersten Feldnamen wie »Name« ein, legen seinen Datentyp, die Feldlänge und – bei numerischen Feldern – die Anzahl der Nachkommastellen fest und aktivieren *Erforderlich*, wenn der Anwender gezwungen sein soll, im betreffenden Feld etwas einzugeben.

Hinzufügen schließt die Felddefinition ab. Das Feld wird in die Liste der vorhandenen Tabellenfelder am unteren Rand des Dialogfelds eingefügt.

Auf die gleiche Weise definieren Sie die restlichen Felder, beispielsweise »Vorname«, »Straße«, »Ort«, »Plz« und »Telefon« (Abbildung 14.14).

Haben Sie sich vertan, selektieren Sie das betreffende Feld im unteren Listenfeld und ändern seine Definition oder löschen es mit *Entfernen*.

Neue Tabellendefinition

Tabellenname: Erstellen

Abbrechen

Feldbeschreibung:

Feldname: Hinzufügen

Typ:

Länge: Dezimal:

Erforderlich: ☐

Felder:

Entfernen

Abbildung 14.13:
Tabelle definieren

Neue Tabellendefinition

Tabellenname: Erstellen

Abbrechen

Feldbeschreibung:

Feldname: Hinzufügen

Typ:

Länge: Dezimal:

Erforderlich: ☐

Felder:

Entfernen

Abbildung 14.14:
Tabellenstruktur festlegen

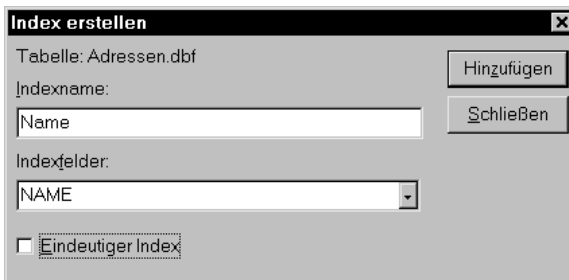
Sind Sie fertig, aktivieren Sie *Erstellen*. Query meldet, dass die Tabelle »Anschrift« erfolgreich erstellt wurde. Das Tabellenauswahlfeld erscheint wieder und im zugehörigen Listenfeld befindet sich nun der Eintrag ANSCHRIFT.DBF, der darauf hinweist, dass der betreffende Ordner die soeben erzeugte Tabelle enthält.

Sie können nun mit *Neu...* eine weitere Tabelle erzeugen. Stattdessen können Sie auch die soeben erzeugte Tabelle »Anschritt« oder eine andere in diesem Ordner eventuell vorhandene dBase-Datei selektieren, mit *Ansicht...* wieder das *Tabellendefinitions*-Dialogfeld öffnen oder mit *Entfernen* die selektierte Tabelle löschen.

Indexdefinition

Mit *Index...* können Sie in der momentan im Listenfeld selektierten Tabelle für Felder, die häufig zur Suche verwendet werden, einen Index erstellen, der die Suche nach bestimmten Inhalten dieser Felder erheblich beschleunigt (Abbildung 14.15).

Abbildung 14.15:
Index erstellen



Hier wird ein Index für das Feld »Name« erstellt, um die Suche nach einem bestimmten Namen wie »Maier« oder »Huber« zu beschleunigen.

Unter *Indexfelder* selektieren Sie das gewünschte Tabellenfeld und unter *Indexname* geben Sie dem Index einen Namen. Er muss keineswegs mit dem Namen des Tabellenfelds identisch sein, sondern kann »Name_Index« oder »X« lauten. Anschließend klicken Sie auf *Hinzufügen* und definieren danach gegebenenfalls einen weiteren Index.

Schließen Sie dieses und das übergeordnete Dialogfeld jeweils mit *Schließen*, so präsentiert sich Query wieder nackt wie zu Beginn.



Bei einem Feld wie »Name«, in das häufig identische Inhalte eingetragen werden, müssen Sie Eindeutiger Index deaktivieren, sonst lässt Query nicht zu, dass Sie mehrere Datensätze mit gleichem Inhalt im Indexfeld »Name« eingeben, was bei Dutzenden von Adressen mit dem identischen Namen »Maier« fatal wäre!

Sinnvoll ist diese Option dagegen für ein Indexfeld wie »Kundennummer«, in dem dadurch verhindert wird, dass mehrere Kunden versehentlich die gleiche Kundennummer erhalten.

14.4 Abfragen

In den beiden vorhergehenden Kapiteln sahen Sie, wie Sie neue Datenquellen definieren, vorhandene Datenquellen für Abfragen verwenden und sogar neue Tabellen anlegen können.

Abfragen bieten jedoch weitaus mehr. Unter anderem können Sie auch neue Datensätze in die zu Grunde liegenden Tabellen eintragen oder bereits vorhandene verändern oder löschen. Grundlage für all das ist Querys Abfragefenster.

Um darin Daten zu bearbeiten, können Sie in Excel DATEN|EXTERNE DATEN|NEUE ABFRAGE ERSTELLEN... wählen oder Query als eigenständiges Programm aufrufen (MSQUERY.EXE im Unterverzeichnis OFFICE des OFFICE-Verzeichnisses) und DATEI|NEU... wählen.

In beiden Fällen müssen Sie anschließend im *Datenquelle auswählen*-Dialogfeld eine passende Datenquelle selektieren, beispielsweise die zuvor definierte Datenquelle »Test«.

Im Query-Assistenten fügen Sie anschließend die neue Tabelle in die Abfrage ein, übergehen die folgenden Schritte und wählen zuletzt *Daten in Microsoft Query bearbeiten oder ansehen*.

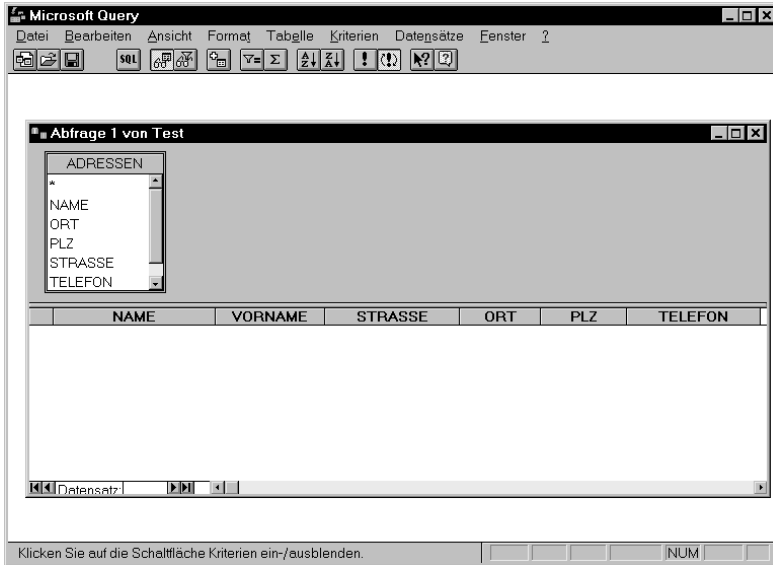
Um stattdessen eine bereits bestehende Abfrage wie die in Kapitel 14.2, »Externe Daten in Excel importieren«, erstellte und unter dem Namen »Abfrage 1 aus Test.dqy« gespeicherte Abfrage zu bearbeiten, wählen Sie in Excel DATEN|EXTERNE DATEN IMPORTIEREN|DATEN IMPORTIEREN..., selektieren die gespeicherte Abfrage, klicken anschließend auf Abfrage bearbeiten, dann jeweils auf Weiter und wählen im letzten Schritt Daten in Microsoft Query bearbeiten oder ansehen. (Alternative: Beim Aufruf von Query als eigenständiges Programm wählen Sie darin DATEI|ÖFFNEN..., selektieren die zu öffnende Abfrage und gehen in den folgenden Schritten genauso vor.)

In jedem Fall erscheint die neue oder geöffnete Abfrage daraufhin im Abfragefenster, einem geteilten Fenster mit zwei voneinander unabhängigen Hälften (mindestens zwei; definierten Sie Kriterien, gibt es ein drittes Kriterienfenster, auf das ich im Kapitel 14.4.1, »Sortieren und Filtern«, eingehe), das prinzipiell folgendermaßen aussieht (Abbildung 14.16).

Ob die untere Fensterhälfte mit den von Ihnen ausgewählten Tabellenspalten entsprechend der Abbildung leer ist oder nicht, hängt von der Datenquelle ab. Stammen die Daten nicht aus leeren Tabellen, sondern wie in der Abfrage »Abfrage 1 aus Test.dqy« aus Tabellen, die bereits ein paar Datensätze enthalten, so werden diese Datensätze im unteren Fensterteil angezeigt.



Abbildung 14.16:
Abfragefenster



Sie können die Teilungslinie durch Anklicken und Ziehen beliebig verschieben, um die Unterteilung zu verändern. Zusätzlich können Sie das Abfragefenster frei verschieben und beliebig vergrößern oder verkleinern.

Die obere Hälfte enthält die Feldliste der Tabelle »Adressen«. Auch dieses Fenster können Sie durch Ziehen am Fenstertitel »ADRESSEN« – nur in der oberen Fensterhälfte – verschieben und durch Ziehen an den vier Randlinien vergrößern oder verkleinern.

Felder einfügen

Haben Sie beim Einfügen der Tabellenfelder mit dem Assistenten bestimmte Felder vergessen, können Sie es jederzeit nachträglich durch Ziehen aus der Feldliste in die untere Fensterhälfte als zusätzliche Spalte in die Abfrage einfügen.

An welcher Position es eingefügt wird, hängt davon ab, wo Sie es fallen lassen: Es wird immer vor jenem Feldnamen eingefügt, der sich auf gleicher Höhe befindet.

Lassen Sie den Feldnamen ganz rechts fallen, wird er entsprechend auch ganz rechts eingefügt, neben dem letzten Feld. Lassen Sie ihn auf oder unterhalb eines bereits vorhandenen Feldnamens fallen, wird er davor eingefügt und dieser entsprechend nach rechts verschoben.

Mehrfachselektion

Sie können in der Feldliste mehrere Felder gleichzeitig selektieren, indem Sie auf einen Feldnamen, bei gedrückter **[Strg]**-Taste auf einen weiteren Feldnamen klicken und so weiter. Einen zusammenhängenden Feldbereich selektieren Sie durch Anklicken des obersten Felds und folgendes Anklicken des untersten Felds mit gedrückter **[⇧]**-Taste.

Alle Felder gleichzeitig selektieren Sie durch einen Doppelklick auf die Titelleiste »Adressen« des Tabellenfensters.

Klicken Sie nach der Selektion irgendeines der selektierten Felder an, halten Sie die Maustaste gedrückt und ziehen Sie es in die untere Hälfte des Abfragefensters, so werden dort alle selektierten Felder gemeinsam eingefügt.

Löschen Sie bitte alle zuvor in die untere Fensterhälfte eingefügten Feldnamen wieder (auf die Feldnamen klicken und BEARBEITEN|LÖSCHEN wählen oder **Entf** drücken) und ziehen Sie das Sternchen »*« in die untere Fensterhälfte. Es steht stellvertretend für alle Feldnamen, fügt also alle Felder gleichzeitig in die untere Fensterhälfte ein.

* = Alle Felder

Alle Tabellenfelder gleichzeitig sehen Sie nur, wenn Sie das Abfragefenster entsprechend vergrößern oder die Breite der einzelnen Spalten auf die gleiche Weise wie in Excel-Tabellen verringern (an den Trennlinien ziehen oder darauf doppelklicken). Sonst müssen Sie den gewünschten Ausschnitt mit den Bildlaufpfeilen am unteren Fensterrand scrollen.



Um Datensätze einzugeben, müssen Sie DATENSÄTZE|BEARBEITEN ZULASSEN wählen. Daraufhin erscheint eine Leerzeile, in die Sie auf die gleiche Weise wie in eine Zeile einer Excel-Tabelle Daten eingeben können (Abbildung 14.17).

Dateneingabe

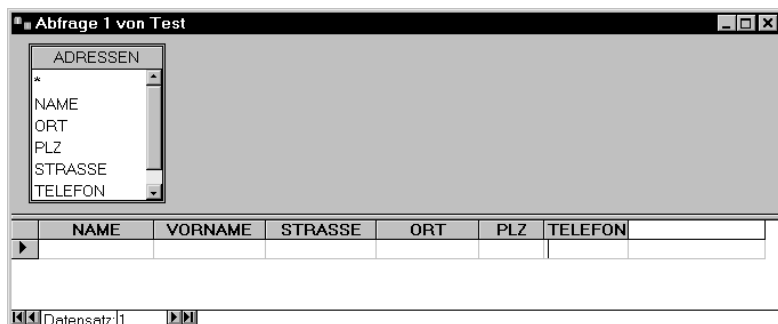


Abbildung 14.17:
Dateneingabe
vorbereiten

Mit **Tab** gelangen Sie zum nächsten, mit **Shift + Tab** zum vorhergehenden Feld. Nach dem letzten Feld führt **Tab** automatisch zur ersten Spalte der zweiten Zeile, also zum ersten Feld des zweiten Datensatzes (Abbildung 14.18).

Haben Sie einen Datensatz eingegeben oder geändert, so wird er nach dem Verlassen der betreffenden Zeile automatisch gespeichert.

Tab und **Rücktab** selektieren immer ein komplettes Feld. Selektierte Felder oder Feldteile können Sie wie gewohnt mit Tastenkombinationen wie **Entf** oder mit den Befehlen des BEARBEITEN-Menüs löschen, kopieren und einfügen.

Feld selektieren

Abbildung 14.18:
Datensätze
eingeben

NAME	VORNAME	STRASSE	ORT	PLZ	TELEFON
Bauer	Gerd	Schmidtstr.12	Mannheim	68005	0621/97878
Maier	Anton	Waldstr.15	Hamburg	70008	10373/9373
Maier	Willi	Breite Str.5	München	80003	73270/9836

Sehr nützlich zum Editieren eines Feldinhalts ist die Taste **[F2]**, die wie bei Excel den Bearbeitungsmodus einschaltet.

*Datensatz
selektieren*

Durch Anklicken der kleinen Zelle ganz links neben jedem Datensatz (seinem Selektionskästchen) wird der betreffende Satz komplett selektiert. **[Entf]** würde diesen Satz nun löschen (nach Nachfrage von Query, ob Sie das auch wirklich wollen).

Mehrere Datensätze gleichzeitig selektieren Sie wie die Feldnamen im Tabellenfenster: Halten Sie einfach beim Anklicken jedes weiteren Satzes bzw. des zugehörigen Selektionskästchens die **[Strg]**-Taste gedrückt.

Um einen Bereich von Sätzen zu selektieren, klicken Sie auf das Selektionskästchen des obersten Satzes und danach bei gedrückter **[⇧]**-Taste auf das des untersten Satzes.

*Spalten
selektieren*

Die Spaltenüberschriften können Sie durch Anklicken selektieren und danach mit **[Entf]** löschen. Mit den gleichen Techniken wie bei Zeilen können Sie sogar mehrere Spalten gleichzeitig selektieren.

*Gesamte Tabelle
selektieren*

Klicken Sie auf das kleine Kästchen in der linken oberen Tabellenecke, so wird wie in Excel die gesamte Tabelle selektiert und kann anschließend beispielsweise mit **[Entf]** gelöscht werden.

Spaltenbreiten

Durch Ziehen an den Trennlinien zwischen Spaltenüberschriften oder Zeilennummern können Sie wie in Excel die Spaltenbreiten bzw. die Zeilenhöhen verändern. Sogar der Doppelklick (optimale Breite) ist möglich.

*Spalten-
reihenfolge*

Um die Reihenfolge der Spalten zu ändern, selektieren Sie eine Spalte, lassen die Maustaste wieder los, klicken die Spaltenüberschrift (den Feldnamen) erneut an und ziehen ihn bei gedrückter Maustaste zu jener Spalte, vor der er eingefügt werden soll.

Mit dem Menü ANSICHT können Sie die Optik des Abfragefensters beeinflussen; beispielsweise mit dem bekannten ANSICHT|ZOOM...-Befehl den Fensterinhalt vergrößern oder verkleinern oder mit ANSICHT|TABELLEN die Anzeige der oberen Fensterhälfte mit dem Tabellenfenster deaktivieren.

Menübefehle

Mit dem Menü FORMAT können Sie den Fensterinhalt formatieren beispielsweise mit FORMAT|SCHRIFTART... die verwendete Schrift definieren, mit FORMAT|ZEILENHÖHE... und FORMAT|SPALTENBREITE... die Höhe der momentan selektierten Zeilen und die Breite der selektierten Spalten bestimmen oder mit FORMAT|SPALTEN AUSBLENDEN die momentan selektierten Spalten ausblenden bzw. mit FORMAT|SPALTEN EINBLENDEN... wieder einblenden.

Die Anzeige »Datensatz: X« zeigt die Nummer des momentan bearbeiteten Satzes an. Enthält Ihre Tabelle viele Sätze, können sie nicht alle gleichzeitig auf dem Bildschirm dargestellt werden. Die Pfeile neben der Anzeige ersetzen jedoch die vertikalen Bildlaufpfeile und blättern zum ersten (<|), letzten (>|), vorhergehenden (<) oder nächsten (>) Satz.

DATENSÄTZE|GEHE ZU... ermöglicht die Eingabe einer direkt anzuspringenden Datensatznummer. Noch einfacher ist es, in der Anzeige »Datensatz: X« auf die angezeigte Nummer zu klicken und diese danach durch die gewünschte Datensatznummer zu überschreiben.

14.4.1 Sortieren und Filtern

Jede Abfrage ist eine individuelle Sicht auf eine Tabelle, die mithilfe von Kriterien beliebig sortiert und gefiltert werden kann. Um eine erarbeitete Sortierung und Filterung nicht zu verlieren, können Sie eine Abfrage mit DATEI|SPEICHERN oder DATEI|SPEICHERN UNTER... unter einem individuellen Namen mit der Endung DQY speichern.

Abfrage speichern und öffnen

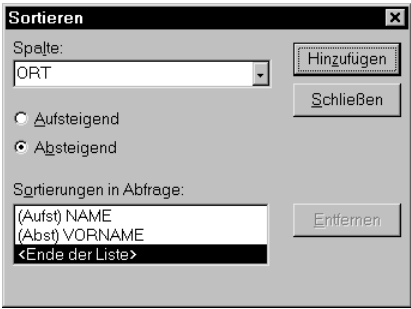
Sie können eine Abfrage mit DATEI|SPEICHERN UNTER... beliebig oft speichern, beispielsweise nach »NAME« sortiert unter dem Namen »Sort (Name)« speichern und zweites Mal nach »VORNAME« sortiert unter dem Namen »Sort (Vorname)«.

Später können Sie jederzeit mit DATEI|ÖFFNEN... die gewünschte Abfrage wieder öffnen, ohne die zugehörigen Sortier- und Filterkriterien neu definieren zu müssen.

Diese Kriterien können Sie nämlich trotz der einleitenden Selektionsschritte jederzeit auch nachträglich ändern, beispielsweise mit DATENSÄTZE|SORTIEREN... nachträglich die Sortierreihenfolge verändern (Abbildung 14.19).

Sortieren

Abbildung 14.19:
Datensätze
sortieren



Unter *Spalte* selektieren Sie den gewünschten Feldnamen, beispielsweise »NAME«, und klicken auf *Hinzufügen*. Die dadurch definierte Sortierung wirkt sich sofort auf die Datensatzreihenfolge aus, noch bevor das Dialogfeld geschlossen wird.

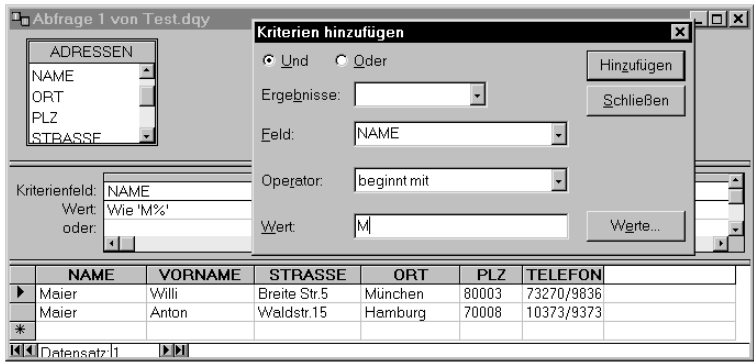
Sie können der Liste entsprechend der Abbildung ein weiteres Sortierkriterium wie »VORNAME« als Untersortierkriterium hinzufügen. Weitere Sortierkriterien werden immer vor dem momentan im Listenfeld *Sortierungen in Abfrage* selektierten Eintrag eingefügt.

Selektieren Sie darin »<Ende der Liste>«, so wird »VORNAME« entsprechend vor diesem Eintrag eingefügt, also unterhalb des Eintrags »(Aufst) NAME«. Wählen Sie für »VORNAME« absteigende Sortierung, so wird entsprechend der Abbildung der Eintrag »(Abst) VORNAME« eingefügt.

»NAME« bildet nun das Hauptsortierkriterium und das darunter eingefügte Kriterium »VORNAME« das Sekundärsortierkriterium. Die Datensätze werden daher zunächst aufsteigend nach Namen sortiert und bei gleichen Namen absteigend nach Vornamen. Entsprechend der Abbildung befindet sich daher »Maier, Willi« nun vor »Maier, Anton«.

Filtern Mit KRITERIEN|KRITERIEN HINZUFÜGEN... können Sie die Tabelle filtern (Abbildung 14.20).

Abbildung 14.20:
Kriterium definieren



Unter *Feld* selektieren Sie einen Feldnamen wie »NAME«, unter *Operator* wählen Sie den gewünschten Vergleichsoperator, und unter *Wert* geben Sie den gewünschten Vergleichswert ein – oder aktivieren *Werte...* und erhalten eine Liste aller Werte der betreffenden Spalte, aus der Sie den gewünschten auswählen; beispielsweise »Maier«, wenn nur alle Maier angezeigt werden sollen.

Hinzufügen aktualisiert die Tabelle sofort, sodass im Beispiel entsprechend der Abbildung nur noch die beiden »Maier« angezeigt werden.

Mit ANSICHT|KRITERIEN können Sie den bei der Definition von Kriterien automatisch eingeblendeten Kriterienbereich jederzeit aus- oder durch erneute Anwahl wieder einblenden.

In diesem Bereich werden die zuvor definierten Filterkriterien angezeigt und können darin bei Bedarf auch editiert werden. Darin können Sie auch beliebig viele weitere Kriterien definieren, was jedoch mit dem Befehl KRITERIEN|KRITERIEN HINZUFÜGEN... wie gezeigt erheblich einfacher ist.

KRITERIEN|ALLE KRITERIEN ENTFERNEN löscht alle definierten Kriterien wieder.

ANSICHT|SQL... ermöglicht Ihnen, den SQL-Ausdruck zu editieren, der für die Auswahl der anzuzeigenden Daten und die Sortierung verantwortlich ist – vorausgesetzt, Sie kennen sich mit SQL aus (Abbildung 14.21).

SQL

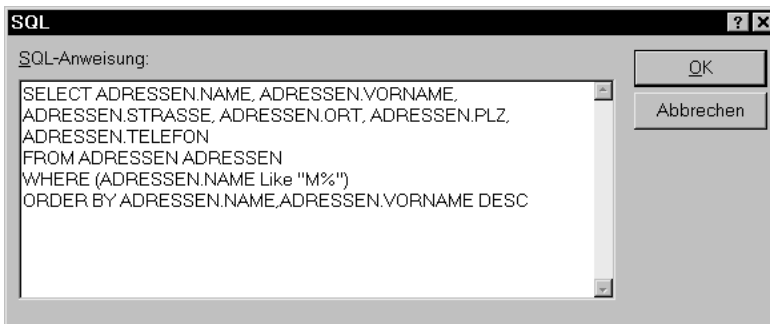


Abbildung 14.21:
SQL-Ausdruck

DATEI|SQL AUSFÜHREN... besitzt eine ähnliche Aufgabe. Ein Fenster erscheint, in dem Sie einen SQL-Ausdruck eingeben und ihn speichern oder sofort ausführen lassen können.

14.4.2 Verknüpfte Tabellen

Eine Abfrage kann mehrere miteinander verknüpfte Tabellen enthalten. Dabei wird zwischen 1:n-, n:1- und n:m-Beziehungen unterschieden. Da dies ein Schnellkursus ist, erläutere ich die zu Grunde liegenden Prinzipien gleich am allgemeinsten und komplexesten Fall, der n:m-Beziehung.



Die Mappe *QUERY.XLS* der Begleit-CD enthält die nachfolgend verwendeten drei Tabellen. Alle drei befinden sich zusätzlich als dBase-Dateien mit den Namen *LIEFERAN.DBF*, *PRODUKT.DBF* und *LIEFPROD.DBF* auf der Begleit-CD (Abbildung 14.22).

Um die folgenden Schritte nachzuvollziehen, kopieren Sie diese drei Dateien bitte in jenes Verzeichnis Ihrer Festplatte, mit dem die zuvor definierte Datenquelle »Test« verbunden ist.

Abbildung 14.22:
Drei dBase-Tabellen

Query.xls:1

	A	B
1	LIEF-NR	PRODUKT-NR
2	1	3
3	1	2
4	2	2
5	3	2
6	4	1
7	5	1
8	5	2
9	5	3
10	6	3
11	7	2
12	7	3
13	8	2
14	9	1
15	10	1
16	10	2
17	11	3
18	12	3

Query.xls:2

	A	B	C	D	E	F
1	LIEF-NR	NAME	VORNAME	STRASSE	PLZ	ORT
2	1	Maier	Stefan	Wannstr. 6	80003	München
3	2	Maier	Albert	Schwalmweg 20	67005	Ludwigst
4	3	Bauer	Gerhard	Breite Str. 10	68002	Mannhei
5	4	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigst
6	5	Müller	Werner	Bielefelder Weg 2	10023	Berlin
7	6	Braun	Otto	Hansestr. 75	40007	Dusseld
8	7	Schmidt	Anton	Weberweg 54	60002	Heidels
9	8	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg
10	9	Bühn	Walter	Hans-Pinsel-Str. 2	80042	München
11	10	Irlbeck	Gustav	Maiersbacherweg 21	80017	München
12	11	Müller	Ottokar	Schmalweg 9	10065	Berlin
13	12	Meier	Franz	Stendaler Weg 23	68003	Mannhei

Query.xls:3

	A	B	C	D	E
1	PRODUKT-NR	TYP	BEZEICHNUNG	STÜCK	PREIS
2	1	Monitor	Eizo 9070S	4	1998
3	2	Monitor	Nokia 447B	2	3587
4	3	Drucker	HP IIIP	8	2294

Datenstruktur Rechts sehen Sie die Tabelle *LIEFERAN.DBF*, die die Adressen einiger Lieferanten und die jeweilige Lieferantenummer enthält.

Darunter befindet sich die Tabelle *PRODUKT.DBF*, die die von diesen Lieferanten bezogenen drei Produkte mit der jeweiligen Produktnummer enthält.

Links sehen Sie die Verbindungstabelle *LIEFPROD.DBF*. Sie verknüpft die Lieferanten- und die Produkttabelle, was unumgänglich ist, da es sich um eine so genannte m:n-Beziehung handelt: Ein Lieferant kann mehrere Produkte liefern und umgekehrt können Sie das gleiche Produkt von mehreren Lieferanten beziehen.

LIEFPROD.DBF stellt die Verbindung zwischen den beiden Tabellen her und informiert eindeutig darüber, welche Produkte Sie von welchen Lieferanten beziehen: Jeder Datensatz dieser Tabelle enthält die Nummer eines Lieferanten und die Nummer des Produkts, das Sie von ihm beziehen.

Beispielsweise beziehen Sie vom Lieferanten mit der Nummer 1, von Herrn Maier, die Produkte mit den Nummern 3 und 2, also den Drucker HP IIIP und den Monitor Nokia 447B. Typisch für eine n:m-Beziehung ist, dass Sie

beide Produkte nicht nur von Herrn Maier, sondern auch von anderen Lieferanten beziehen.

Über all das gibt die Tabelle LIEFPROD.DBF zusammen mit den beiden anderen Tabellen eindeutig Aufschluss.

Um auf die drei dBase-Dateien zuzugreifen, können Sie beispielsweise die zuvor definierte Datenquelle »Test« verwenden.

Wollen Sie die drei Dateien ständig benutzen, können Sie stattdessen mit DATEI|TABELLENDEFINITION... eine eigene Datenquelle definieren, der Sie beispielsweise den Namen »Lieferanten-Produkt-Beziehung« geben.

*Indexdateien
verwenden*

Der Vorteil: Sie können mit der Schaltfläche *Verzeichnis auswählen...* des ODBC dBase-Setup-Dialogfelds den Ordner, in dem sich die drei dBase-Tabellen befinden, als Standardordner angeben, um sich später die Dateiauswahl zu erleichtern.

Vor allem aber können Sie – wenn vorhanden, was im Beispiel nicht der Fall ist – zusätzlich mit der Schaltfläche *Indizes auswählen...* dieses Dialogfelds Indexdateien bekannt geben, die zu den in diesem Ordner vorhandenen dBase-Dateien gehören.

Selektieren Sie eine der dBase-Dateien, so werden die zugehörigen Indexdateien aufgelistet und Sie können jene auswählen, die benutzt werden sollen – falls im Gegensatz zum Beispiel welche vorhanden sind.

Wählen Sie nun bitte in Excel DATEN|EXTERNE DATEN|NEUE ABFRAGE ERSTELLEN... und selektieren Sie in Query als zu verwendende Datenquelle »Test«.

Abfrage erstellen

Fügen Sie ins rechte Listenfeld des Query-Assistenten die drei Tabellen »LIEFERAN«, »LIEFPROD« und »PRODUKT« ein (Abbildung 14.23).

Weiter führt zu einer Warnung, die besagt, dass die Verknüpfung der Tabellen fehlt und per Hand nachgeholt werden muss.

Wegen dieser fehlenden Verknüpfung können die folgenden Selektionsschritte nicht ausgeführt werden und das Abfragefenster erscheint nun sofort (Abbildung 14.24).

Momentan wird im Abfragefenster jeder Lieferant dreifach angezeigt, so als würde jeder Lieferant *jedes* der drei Produkte liefern, was absolut nicht der Fall ist! Die Erstellung einer sinnvollen Abfrage erfordert, dass Sie Query die Beziehungen zwischen diesen drei Tabellen bekannt geben.

*Beziehungen
definieren*

Abbildung 14.23:
Tabellen hinzufügen

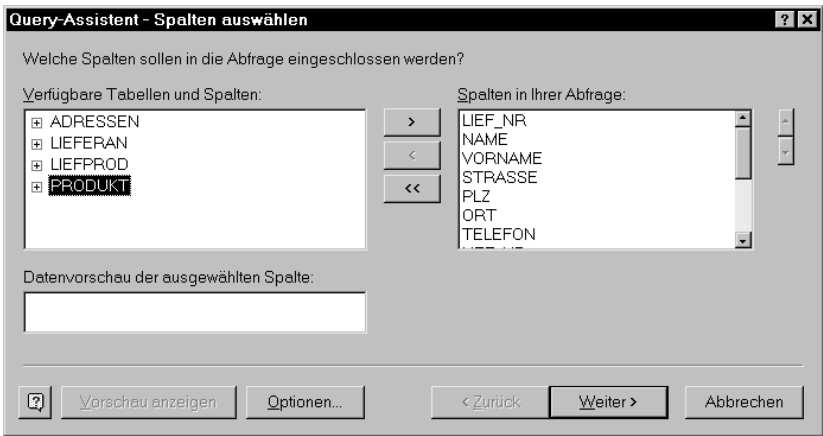
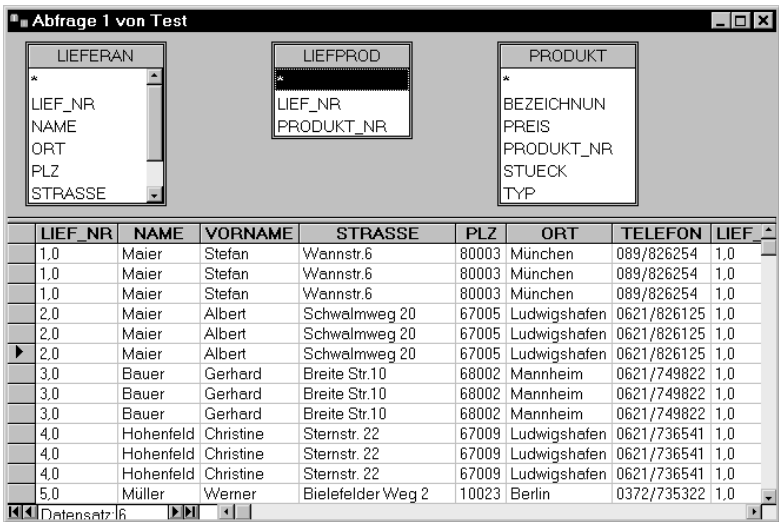


Abbildung 14.24:
Eingefügte Tabellen



Dazu klicken Sie das Feld »LIEF-NR« der Tabelle »LIEFERAN« an und ziehen es zum korrespondierenden Feld »LIEF-NR« der Tabelle »LIEFPROD«: Query fügt zwischen den beiden Feldern eine Verbindungslinie ein.

Verbinden Sie auf die gleiche Weise das Feld »PRODUKT-NR« der Tabelle »PRODUKT« mit dem Feld »PRODUKT-NR« der Tabelle »LIEFPROD« (Abbildung 14.25).

Falsch definierte Beziehungen können Sie durch Selektion der Linie und Löschen mit **[Entf]** korrigieren.

Erst jetzt, nachdem Query die Beziehungen zwischen den drei Tabellen bekannt sind, wird korrekt angezeigt, welcher Lieferant welches Produkt liefert.

Abfrage 1 von Test

LIEFERAN

LIEFPROD

PRODUKT

LIEF_N	NAME	VORNAME	STRASSE	PLZ	ORT	TELEFON	LIEF_N
10,0	Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/726384	10,0
9,0	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	9,0
4,0	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	4,0
5,0	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	5,0
7,0	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	7,0
1,0	Maier	Stefan	Wannstr.6	80003	München	089/826254	1,0
10,0	Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/726384	10,0
8,0	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	8,0
2,0	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826125	2,0
3,0	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	3,0
5,0	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	5,0
1,0	Maier	Stefan	Wannstr.6	80003	München	089/826254	1,0
6,0	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	6,0
12,0	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	12,0
7,0	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	7,0
11,0	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	11,0
5,0	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	5,0

Datensatz: 5

Abbildung 14.25:
Beziehungen
bekannt geben

Löschen Sie überflüssige Spalten durch Klicken auf die Spaltenüberschrift und Wahl von BEARBEITEN|LÖSCHEN (oder einfach **Entf** drücken), so wird das noch deutlicher (Abbildung 14.26).

Abfrage 1 von Test

LIEFERAN

LIEFPROD

PRODUKT

NAME	VORNAME	TYP	BEZEICHNUN	STUECK
Irlbeck	Gustav	Monitor	Eizo 9070S	4,0
Bühn	Walter	Monitor	Eizo 9070S	4,0
Hohenfeld	Christine	Monitor	Eizo 9070S	4,0
Müller	Werner	Monitor	Eizo 9070S	4,0
Schmidt	Anton	Monitor	Nokia 447B	2,0
Maier	Stefan	Monitor	Nokia 447B	2,0
Irlbeck	Gustav	Monitor	Nokia 447B	2,0
Maier	Stefanie	Monitor	Nokia 447B	2,0
Maier	Albert	Monitor	Nokia 447B	2,0
Bauer	Gerhard	Monitor	Nokia 447B	2,0
Müller	Werner	Monitor	Nokia 447B	2,0
Maier	Stefan	Drucker	HP IIIIP	8,0
Braun	Otto	Drucker	HP IIIIP	8,0
Meier	Franz	Drucker	HP IIIIP	8,0
Schmidt	Anton	Drucker	HP IIIIP	8,0
Müller	Ottokar	Drucker	HP IIIIP	8,0
Müller	Werner	Drucker	HP IIIIP	8,0

Datensatz: 17

Abbildung 14.26:
Endgültige Abfrage

Sie sehen auf einen Blick, welche Produkte Sie von den einzelnen Lieferanten beziehen. Query geht dabei folgendermaßen vor: Name und Vorname des ersten Datensatzes der Tabelle »LIEFERAN« werden gelesen, also »Maier, Stefan«. Die zugehörige Lieferantenummer ist 1.

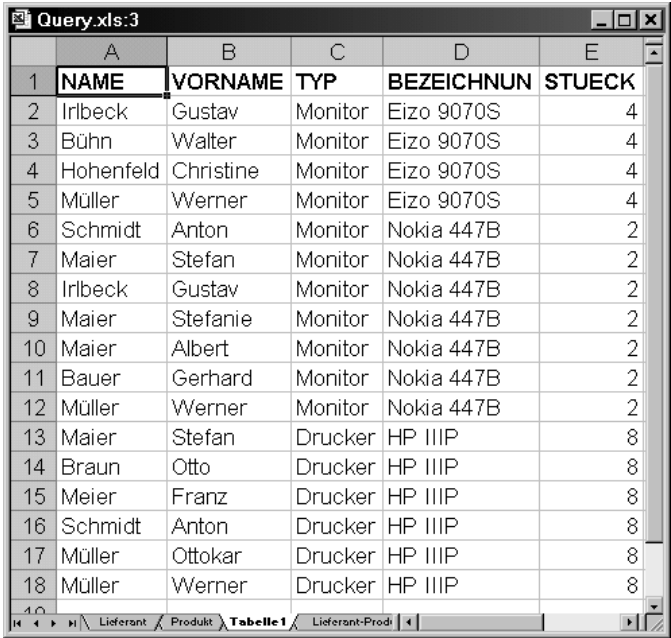
Das Feld »LIEF-NR« dieser Tabelle ist mit dem gleichnamigen Feld der Tabelle »LIEFPROD« verknüpft: Für Query bedeutet das, dass nun in der Tabelle »LIEFPROD« nachgeschaut wird, in welchen Datensätzen das Feld »LIEF-NR« die Lieferantenummer 1 enthält. Jeder dieser Datensätze enthält die zugehörige Produktnummer im Feld »PRODUKT-NR«.

Dieses Feld ist wiederum mit dem gleichnamigen Feld der Tabelle »PRODUKT« verknüpft, in der alle Datensätze mit dieser Produktnummer gesucht werden.

Im Beispiel werden zwei Datensätze gefunden, das heißt zwei Produkte, die von Herrn Stefan Maier geliefert und nun aufgelistet werden.

Beenden Sie Query und importieren Sie diese Daten wie zuvor erläutert in Excel, so erhalten Sie folgendes Resultat (Abbildung 14.27).

Abbildung 14.27:
Importierte Abfrage-
daten



	A	B	C	D	E
1	NAME	VORNAME	TYP	BEZEICHNUN	STUECK
2	Irlbeck	Gustav	Monitor	Eizo 9070S	4
3	Buhn	Walter	Monitor	Eizo 9070S	4
4	Hohenfeld	Christine	Monitor	Eizo 9070S	4
5	Müller	Werner	Monitor	Eizo 9070S	4
6	Schmidt	Anton	Monitor	Nokia 447B	2
7	Maier	Stefan	Monitor	Nokia 447B	2
8	Irlbeck	Gustav	Monitor	Nokia 447B	2
9	Maier	Stefanie	Monitor	Nokia 447B	2
10	Maier	Albert	Monitor	Nokia 447B	2
11	Bauer	Gerhard	Monitor	Nokia 447B	2
12	Müller	Werner	Monitor	Nokia 447B	2
13	Maier	Stefan	Drucker	HP IIIP	8
14	Braun	Otto	Drucker	HP IIIP	8
15	Meier	Franz	Drucker	HP IIIP	8
16	Schmidt	Anton	Drucker	HP IIIP	8
17	Müller	Ottokar	Drucker	HP IIIP	8
18	Müller	Werner	Drucker	HP IIIP	8

Felder und
Tabellen einfügen

Sie können übrigens jederzeit weitere Felder und Tabellen in die Abfrage einfügen.

Wie Sie die Felder vorhandenen Feldlisten einfügen, erläuterte ich bereits. Zusätzlich können Sie einer Abfrage mit TABELLE|TABELLEN HINZUFÜGEN... im zugehörigen *Dateiauswahl*-Dialogfeld jederzeit weitere Tabellen hinzufügen, die Feldlisten dieser Tabellen wie gezeigt mit den bereits vorhandenen Tabellen verknüpfen und anschließend beliebige Felder der zusätzlichen Tabellen in die Abfrage einfügen.

Um umgekehrt eine Tabelle aus der Abfrage zu entfernen, selektieren Sie sie durch Anklicken irgendeines Felds im zugehörigen Tabellenfenster und wählen TABELLE|TABELLEN ENTFERNEN oder drücken einfach **[Entf]**.

TABELLE|VERKNÜPFUNGEN... ermöglicht die gezielte Festlegung von Verknüpfungen (Abbildung 14.28), setzt allerdings voraus, dass Sie sich mit SQL auskennen und wissen, was beispielsweise ein *OUTER JOIN* oder eine Reflexivverknüpfung ist (Tipp: all das wird in meinem Access-Kompodium erläutert).

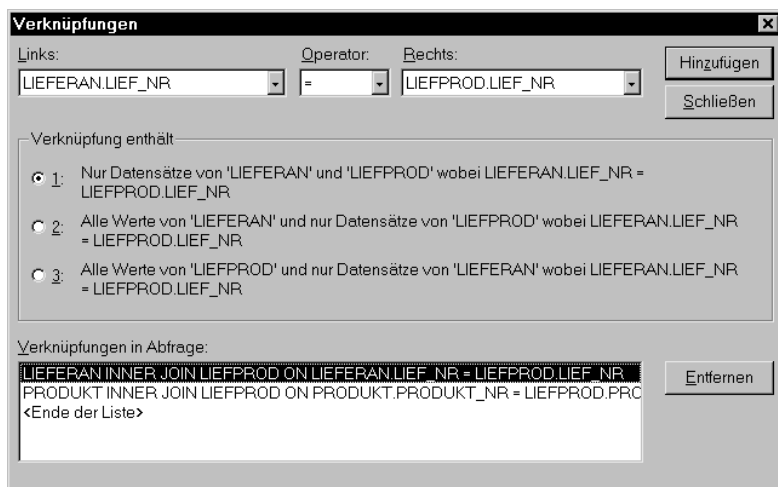


Abbildung 14.28:
Verknüpfungen
definieren

15 Konsolidieren

Konsolidieren bedeutet, dass in mehreren Quellbereichen enthaltene Daten für Auswertungen miteinander verknüpft werden und in einem Zielbereich eine Zusammenfassung erscheint, beispielsweise eine Übersichtstabelle, die die Summen wichtiger Werte mehrerer Tabellen enthält.

Excel erleichtert das mit dem Befehl DATEN|KONSOLIDIEREN... (allerdings finde ich persönlich die Anwendung der in einem früheren Kapitel erläuterten 3D-Verknüpfungen wesentlich einfacher), den ich nun detailliert erläutere.

Zunächst zeige ich Ihnen, wie Sie damit in mehreren Quellbereichen enthaltene Daten in einem Zielbereich zusammenfassen. Der resultierende Bereich besitzt die gleiche Größe und Form wie die Quellbereiche.

Jede Zelle des Zielbereichs enthält anschließend die Summe der Inhalte all jener Zellen, die sich in den Quellbereichen an der gleichen relativen Position befanden.

Statt Summierungen sind selbstverständlich auch andere Verknüpfungen möglich, beispielsweise können die Zellen der Quellbereiche miteinander multipliziert werden.

Anschließend erfahren Sie, wie tabellenübergreifende Konsolidierungen erstellt werden, die es beispielsweise ermöglichen, die Inhalte von zwölf monatlichen Buchführungsblättern in einer Konsolidierungstabelle zusammenzufassen.

Das funktioniert, wenn gewünscht, dynamisch. Dann trägt Excel nicht nur einmalig die resultierenden Ergebnisse in den Zielbereich ein, sondern aktualisiert diesen Bereich automatisch, sobald sich irgendwelche Quellbereichsdaten ändern.

Beachten Sie bitte, dass die im folgenden Kapitel 15.1, »Statische Verknüpfungen«, erläuterten sehr einfachen Probleme wesentlich einfacher mit den im Kapitel 8.6, »3D-Verknüpfungen«, erläuterten 3D-Verknüpfungen gelöst werden können und dass diese gegenüber statischen Verknüpfungen zudem noch den Vorteil ständiger Aktualisierung besitzen!



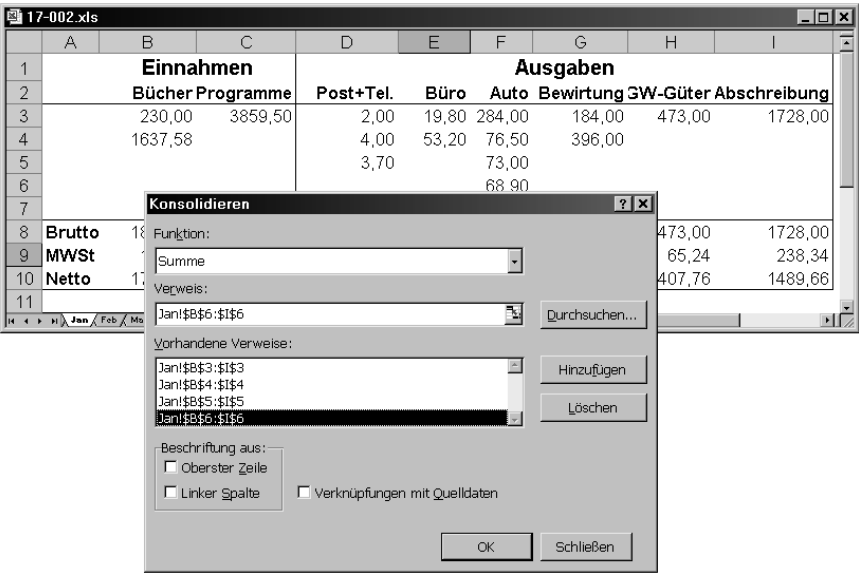


Echte Vorteile gegenüber 3D-Verknüpfungen besitzen Konsolidierungen eigentlich nur durch die in Kapitel 15.2 erläuterte Orientierung an Beschriftungen und – möglicherweise – durch die in Kapitel 15.3 erläuterte automatisch eingesetzte Gliederungsfunktion bei dynamischen Verknüpfungen.

15.1 Statische Verknüpfungen

Ein gut geeignetes Demonstrationsobjekt ist die Mappe BUCHFRG.XLS mit den darin enthaltenen zwölf gleichartigen Monats-Tabellen. Legen Sie darin bitte eine neue Tabelle namens »Konsolidierung« an und selektieren Sie darin die Zelle A1. Wählen Sie nun DATEN|KONSOLIDIEREN... (Abbildung 15.1).

Abbildung 15.1:
Datenkon-
solidierung



Quellbereiche Aktivieren Sie das Feld *Verweis* und selektieren Sie die Tabelle »Jan« und darin wiederum den Bereich B3:I3: Excel fügt den zugehörigen Bezug Jan!\$B\$3:\$I\$3 ein. Klicken Sie auf *Hinzufügen*, so wird er in das Listenfeld *Vorhandene Verweise* eingefügt.

Fügen Sie der Liste auf die gleiche Weise Bezüge auf die Bereiche B4:I4, B5:I5 und B6:I6 von »Jan« hinzu, so erhalten Sie den in der vorhergehenden Abbildung gezeigten Zustand.

Haben Sie sich bei einem Bezug vertan, selektieren Sie ihn in der Liste und löschen ihn mit *Löschen*.

OK verknüpft die Daten der vier Quellbereiche durch die unter *Funktion* vorselektierte Funktion SUMME (Abbildung 15.2).

Buchfrg.xls:1

	A	B	C	D	E	F	G	H	I
1	Einnahmen			Ausgaben					
2		Bücher Programme		Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
9	MWSt	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
10	Netto	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66

Buchfrg.xls:2

	A	B	C	D	E	F	G	H
1	=SUMME(Buchfrg.xls:1!B3:B6)	=SUMME(Buchfrg.xls:1!C3:C6)	=SUMME(Buchfrg.xls:1!D3:D6)	=SUMME(Buchfrg.xls:1!E3:E6)	=SUMME(Buchfrg.xls:1!F3:F6)	=SUMME(Buchfrg.xls:1!G3:G6)	=SUMME(Buchfrg.xls:1!H3:H6)	=SUMME(Buchfrg.xls:1!I3:I6)

Abbildung 15.2:
Quellbereiche und
Zielbereich

Der resultierende Zielbereich besitzt die gleiche Größe und Form wie die vier Quellbereiche, wobei die bei Anwahl von DATEN|KONSOLIDIEREN... aktive Zelle als obere linke Bereichsecke verwendet wird.

Zielbereich

Jede Zelle dieses Bereichs enthält »die Summe der Inhalte all jener Zellen, die sich in den Quellbereichen an der gleichen relativen Position befanden«.

Zur Erläuterung: Der Inhalt der Zelle A1 in der linken oberen Ecke des Zielbereichs ergibt sich durch Addition der vier Zellen B3, B4, B5 und B6 von »Jan«; also jener Zellen, die sich ebenfalls in den linken oberen Ecken der vier Quellbereiche befinden.

Analog dazu wird der Inhalt von B2 durch Addition der vier Quellbereichszellen C3, C4, C5 und C6 gebildet und so weiter.

Sehr interessant sind tabellenübergreifende Konsolidierungen.

Tabellenübergreifend konsolidieren

Bild 15.3 zeigt in zwei Fenstern die Tabellen »Jan« und »Feb« der Mappe BUCHFRG.XLS. »Jan« enthält die gleichen Zahlen wie zuvor. Zu Demonstrationszwecken gab ich zusätzlich in die Spalte »Bücher« von »Feb« einige Zahlen ein (Abbildung 15.3).

Wird als Zielbereich wieder ein Bereich in der Tabelle »Konsolidierung« verwendet, gibt Excel nach erneuter Wahl von DATEN|KONSOLIDIEREN... im Dialogfeld automatisch die zuletzt verwendeten Bezüge vor.

Da die alten Bezüge momentan unbrauchbar sind, müssen sie jedoch nacheinander selektiert und jeweils mit *Löschen* aus der Liste entfernt werden.

Anschließend selektieren Sie »Jan« und darin den Bereich B3:I7 und fügen ihn der leeren Liste hinzu. Danach selektieren Sie »Feb« und auch darin den Bereich B3:I7 und fügen ihn ebenfalls hinzu.

Abbildung 15.3:
Die Quellbereiche
der Tabellen »Jan«
und »Feb«

17-002.xls:1									
	A	B	C	D	E	F	G	H	I
1	Einnahmen			Ausgaben					
2	Bücher Programme			Post+Tel.	Büro	Auto	Bewirtung	GW-Güter	Abschreibung
3		230,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00
4		1637,58		4,00	53,20	76,50	396,00		
5				3,70		73,00			
6						68,90			
7									
8	Brutto	1867,58	385						1728,00
9	MWSt	122,18	53						238,34
10	Netto	1745,40	332						1489,66

17-002.xls:2									
	A	B	C	D	E	F	G	H	I
1		Einnahme							
2		Bücher Pro							
3		487,00							
4		973,00							
5									
6									
7									
8	Brutto	1460,00							
9	MWSt	95,51	0,00	0,00	0,00	0,00	0,00	0,00	
10	Netto	1364,49	0,00	0,00	0,00	0,00	0,00	0,00	

Konsolidieren

Funktion:
Summe

Verweis:
Feb!\$B\$3:\$I\$7

Durchsuchen...

Vorhandene Verweise:
Feb!\$B\$3:\$I\$7
Jan!\$B\$3:\$I\$7

Hinzufügen
Löschen

Beschriftung aus:
☐ Oberster Zeile
☐ Linker Spalte
☐ Verknüpfungen mit Quelldaten

OK
Schließen

Nun aktivieren Sie die Konsolidierung mit OK (Abbildung 15.4).

Abbildung 15.4:
Tabellen-
übergreifende
Konsolidierung

Buchfrg.xls:2									
	A	B	C	D	E	F	G	H	
1	717,00	3859,50	2,00	19,80	284,00	184,00	473,00	1728,00	
2	2610,58		4,00	53,20	76,50	396,00			
3			3,70		73,00				
4					68,90				
5									

Erneut werden Zellen, die sich in den beiden Bezugsbereichen an den gleichen relativen Positionen befinden, über die *SUMME*-Funktion miteinander verknüpft und das Ergebnis in der entsprechenden Position des Zielbereichs gespeichert.

Auf die gleiche Weise können Sie in der Demomappe die Inhalte aller zwölf Tabellen konsolidieren, indem Sie entsprechend zwölf Bezüge hinzufügen, JAN!\$B\$3:\$I\$7, FEB!\$B\$3:\$I\$7, ..., NOV!\$B\$3:\$I\$7, DEZ!\$B\$3:\$I\$7.

Konsolidierungs-
funktionen

Excel kann die Werte in den Quellbereichen mit nahezu beliebigen statistischen Funktionen verknüpfen. Da es die gängigste Funktion ist, wird zwar immer *SUMME* vorgegeben, insgesamt stellt das Listenfeld *Funktion* jedoch rund ein Dutzend Funktionen zur Verfügung.

Sie zu summieren, können Sie beispielsweise mit der Funktion *MITTELWERT* den Mittelwert der Quellbereichsdaten ermitteln oder mit *STANDARDABWEICHUNG* die Standardabweichung berechnen, die durchschnittliche Abweichung der Werte vom Mittelwert.

Bei statischen Konsolidierungen ist die Position des Zielbereichs frei wählbar. Er darf sich in der gleichen Tabelle wie einer oder mehrere der Zielbereiche befinden, er kann sich stattdessen wie in den vorhergehenden Beispielen in einer anderen Tabelle der gleichen Mappe befinden und sogar in einer Tabelle *einer anderen Mappe*!

*Bereichs-
positionen*

Ist momentan die Zielbereichsmappe aktiv, die Quellbereichsmappe jedoch nicht, müssen Sie letztere noch nicht einmal öffnen, um auf die Quellbereiche zeigen zu können.

Stattdessen können Sie auch die Schaltfläche *Durchsuchen...* aktivieren, die das bekannte *Dateiselektions*-Dialogfeld öffnet. Darin selektieren Sie jene Mappe, die in die Datenkonsolidierung mit einbezogen werden soll.

Excel fügt daraufhin den vollständigen Pfad und den Dateinamen dieser Mappe in das Listenfeld *Vorhandene Verweise* ein. Sie selektieren nun diesen Eintrag und ergänzen ihn um die Angabe der Sie interessierenden Tabelle, ein Ausrufezeichen und den Bezug des zu konsolidierenden Quellbereichs, ergänzen also beispielsweise folgende Angaben: »Jan!\$B\$3:\$I\$7«.

Statt einer aktiven Zelle können Sie vor der Konsolidierung den gewünschten Zielbereich selektieren. Dann erstreckt sich die Konsolidierung nur auf diesen Bereich – wobei die Auswertung jedoch zwangsläufig unvollständig ist, wenn der gewählte Bereich kleiner als der größte Quellbereich ist!

Vorhandene Konsolidierungen können Sie sehr einfach ändern: Ist die Tabelle aktiv, die die Konsolidierung enthält (also den Zielbereich), werden nach dem Öffnen des *Konsolidieren*-Dialogfelds darin die zuvor definierten Bezüge vorgegeben. Sie können nun die Bezüge und/oder die angewendete Konsolidierungsfunktion ändern und mit *OK* eine neue Konsolidierung starten.

*Konsolidierung
ändern*

Das vereinfacht die Aktualisierung einer Konsolidierung enorm: Hat sich in einem Quellbereich ein Wert geändert, enthält der Zielbereich daher veraltete Konsolidierungsdaten, so aktivieren Sie einfach die Zielbereichstabelle, öffnen das *Konsolidieren*-Dialogfeld und schließen es mit *OK*.

Die Konsolidierung wird daraufhin erneut durchgeführt und der Zielbereich aktualisiert.



Alternativ dazu können Sie für eine automatische Aktualisierung des Zielbereichs sorgen, indem Sie das Kontrollkästchen *Verknüpfungen mit Quelldaten aktivieren* (siehe Kapitel 15.3, »Dynamische Verknüpfungen«). Voraussetzung ist jedoch, dass sich der Zielbereich in einer anderen Tabelle befindet als die Quellbereiche!

15.2 Orientierung an Beschriftungen

Unterschiedlicher Tabellenaufbau

Für die soeben durchgeführte tabellenübergreifende Auswertung wären 3D-Verknüpfungen völlig ausreichend gewesen und der Einsatz der Datenkonsolidierung daher überflüssig, im Gegensatz zu folgendem Beispiel (Abbildung 15.5).

Abbildung 15.5:
Quellbereiche mit
unterschiedlichem
Aufbau

17-005.xls:1			
	A	B	C
1	Netto	MWSt	Brutto
2	32,45	5,192	37,642
3			
4			
Tabelle1 Tabelle2 Tabelle3 Tabelle4 Tabelle5 Tabelle6			

17-005.xls:2			
	A	B	C
1	Netto	Brutto	MWSt
2	64,5	74,82	10,32
3			
4			
Tabelle2 Tabelle3 Tabelle4 Tabelle5 Tabelle6 Tabelle7			

17-005.xls:3			
	A	B	C
1			
2			
3			
4			
Tabelle3 Tabelle4 Tabelle5			

Beide oberen Tabellen enthalten im Bereich A2:C2 einen Nettobetrag, die zugehörige Umsatzsteuer und den resultierenden Bruttobetrag.

Die Anordnung der Kategorien »USt« und »Brutto« in diesen Bereichen ist jedoch unterschiedlich: In »Tabelle1« enthält B2 die Umsatzsteuer und C2 den Bruttobetrag, in »Tabelle2« ist es genau umgekehrt.

3D-Verknüpfungen können Sie hier ebenso wenig zur Zusammenfassung einsetzen wie die bisher gezeigte Form der Datenkonsolidierung.

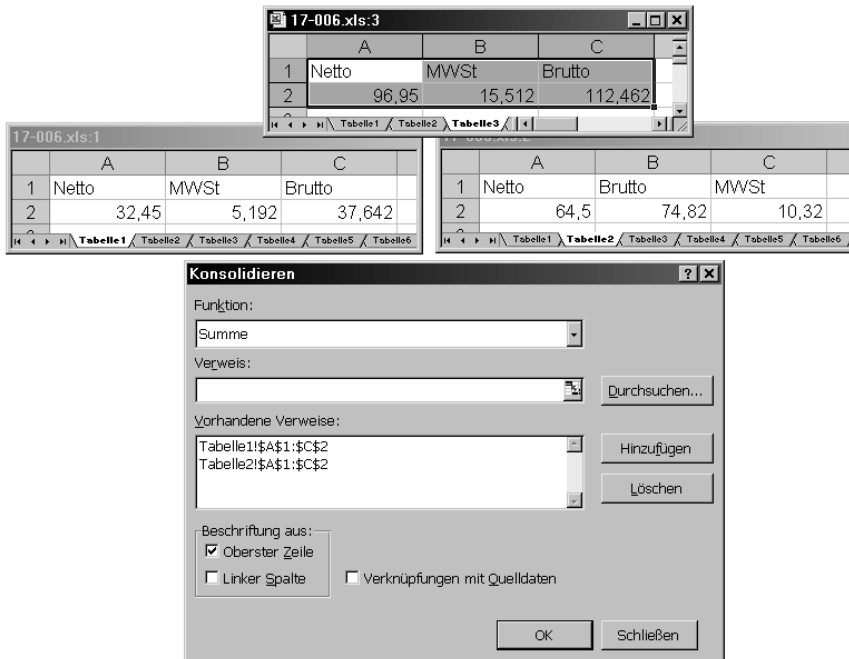
Allerdings können Sie Excel mit dem Optionsfeld *Beschriftung aus* beibringen, wie die Werte in den zu konsolidierenden Bereichen angeordnet sind – vorausgesetzt, dass die Bereiche entsprechend der Abbildung nicht nur Werte, sondern auch zugehörige Zeilen- oder Spaltenüberschriften enthalten!

Dann ist Excel in der Lage, sich bei der Datenkonsolidierung statt an den relativen Positionen der Felder in den Quellbereichen an diesen Beschriftungen zu orientieren.

Dazu müssen Sie die Quellbereiche *inklusive der Zeilenüberschriften* angeben, im Beispiel also jeweils die Bereiche A1:C2 der beiden Tabellen. Anschließend aktivieren Sie die Option *oberster Zeile* und starten die Konsolidierung (Abbildung 15.6).

Oberste Zeile

Abbildung 15.6:
Ergebnis der Datenkonsolidierung



Excel übernimmt bei der Konsolidierung auch die zusätzlich selektierten Überschriften. Und addiert unter »USt« wie geplant die in B2 beziehungsweise C2 enthaltenen Umsatzsteuerbeträge.

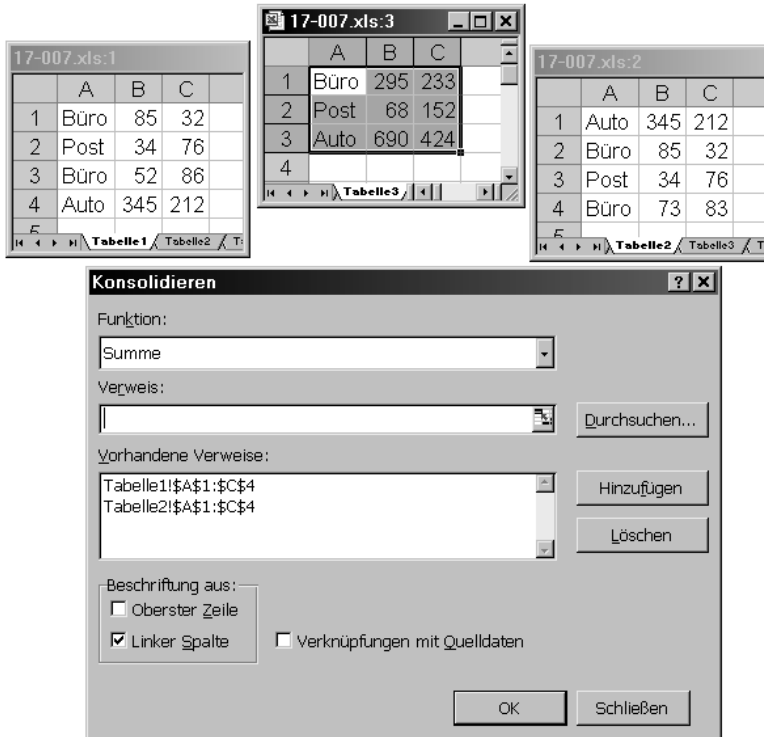
Ebenso clever verhält sich Excel bei den Bruttobeträgen und addiert auch hier nicht einfach Zellen mit identischen relativen Quellbereichspositionen, sondern jeweils die Zellen mit der Überschrift »Brutto«.

Die Option *linker Spalte* bewirkt das Gleiche, wenn sich die Überschriften nicht oberhalb der zu konsolidierenden Werte befinden, sondern links davon (Abbildung 15.7).

Linke Spalte

Diese Abbildung zeigt eine weitere interessante Eigenschaft der Orientierung an Überschriften bei der Konsolidierung: Die Kategorie »Büro« ist überflüssigerweise doppelt vorhanden. Bei der Konsolidierung fasst Excel die beiden gleichnamigen Kategorien zu einer einzigen Kategorie zusammen, in der alle vier Werte summiert sind!

Abbildung 15.7:
Zeilenüberschriften



Wenn die Reihenfolge der Kategorien in allen Bezugsdateien identisch ist, ist es nicht notwendig, dass sich Excel an den Kategoriennamen orientiert.

Dennoch sollten Sie auch in diesem Fall bei der Bereichsselektion die Kategoriennamen integrieren und *oberster Zeile* bzw. *linker Spalte* aktivieren. Denn zweifellos ist es recht nützlich, dass Excel dann nicht nur Zahlen, sondern auch die zugehörigen Bezeichnungen in den Konsolidierungsbereich einsetzt.

15.3 Dynamische Verknüpfungen

Es wäre nett, wenn Excel bei der Konsolidierung keine konstanten Werte, sondern externe Bezüge in den Konsolidierungsbereich eintragen würde, damit dieser auch bei Änderungen der Quellbereichsdaten ständig aktuell ist.

Genau das ist die Aufgabe von *Verknüpfungen mit Quelldaten*. Aktivieren Sie dieses Kontrollkästchen, wird eine dynamische Verbindung zu den Quellbereichen aufgebaut und nach jeder Änderung in einem dieser Bereiche der Zielbereich automatisch aktualisiert.

Sie können diese Verknüpfung auch nachträglich für eine bereits bestehende Konsolidierung aufbauen, indem Sie die Zielbereichstabelle aktivieren, DATEN|KONSOLIDIEREN... wählen und im zugehörigen Dialogfeld *Verknüpfungen mit Quelldaten* aktivieren.

Bei dynamischen Verknüpfungen gibt es jedoch eine gewichtige Einschränkung: Der Zielbereich darf sich nicht in der gleichen Tabelle befinden wie einer der Quellbereiche!



Statische Konsolidierungen, bei denen sich der Zielbereich in einer der Quellbereichstabellen befindet, lassen sich daher nicht in dynamische Konsolidierungen umwandeln.

Das folgende Beispiel verwendet einen Zielbereich, der sich nicht nur in einer anderen Tabelle, sondern sogar in einer anderen Arbeitsmappe als die Quellbereiche befindet (Abbildung 15.8).

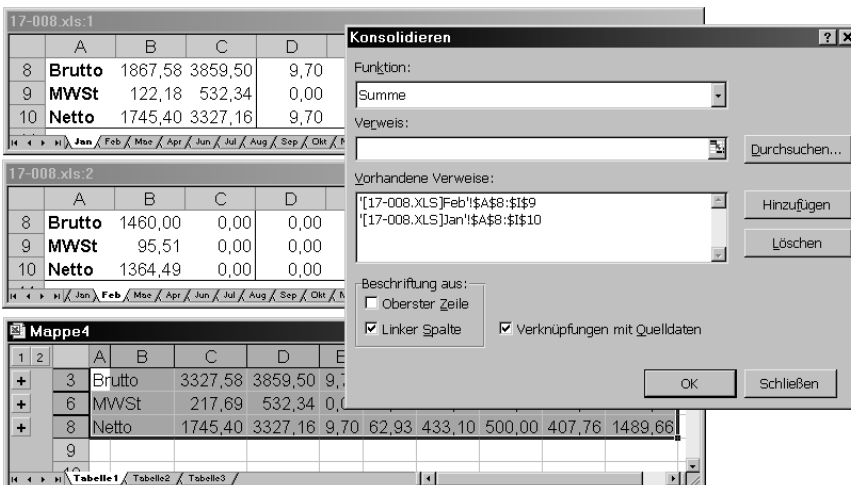


Abbildung 15.8:
Dynamische
Konsolidierung

Die Abbildung zeigt außer der Buchführungsmappe mit den zwölf Monats-Tabellen eine weitere Mappe, die eine dynamische Konsolidierung der beiden Monats-Tabellen »Jan« und »Feb« enthält.

Zunächst ging ich genauso vor wie bei statischen Konsolidierungen: Die Zelle A1 der neu angelegten Tabelle wurde als obere linke Ecke des Konsolidierungsbereichs selektiert und danach DATEN|KONSOLIDIEREN... gewählt.

Anschließend aktivierte ich *Verweis*, zeigte in der Buchführungsmappe auf den Bereich A8:I10 der Tabelle »Jan«, der jeweils die Brutto-, MWSt- und Nettosummen der einzelnen Rubriken für den betreffenden Monat enthält, und aktivierte *Hinzufügen*.

Danach zeigte ich in der Buchführungsmappe auf den Bereich A8:I10 der Tabelle »Feb« und aktivierte wieder *Hinzufügen*.

Nach dem Hinzufügen dieser beiden Bezüge aktivierte ich das Kontrollkästchen *linker Spalte*, um Excel darauf hinzuweisen, dass die linken Bereichsspalten Rubrikennamen enthalten und nicht etwa zu konsolidierende Daten.

Vor allem aber aktivierte ich *Verknüpfen mit Quelldaten*, bevor ich das Dialogfeld mit OK schloss! Daher werden diesmal keine konstanten Werte in den Konsolidierungsbereich eingefügt, sondern externe Bezüge.

Wenn Sie das Ganze nachvollziehen, stellen Sie fest, dass bei Testeingaben in die Tabellen »Jan« oder »Feb« die Summen in der Konsolidierungstabelle sofort entsprechend aktualisiert werden.

Automatische Gliederung

Die dazu verwendeten externen Bezüge im Konsolidierungsbereich bekommen Sie nur zu sehen, wenn Sie in der von Excel automatisch erstellten Gliederung die tiefste Gliederungsebene (2) aktivieren (Abbildung 15.9).

Abbildung 15.9:
Die Gliederungs-
funktion

	A	B	C	D	E	F	G	H	I	J
3	Brutto		1867,58	3859,50	9,70	73,00	502,40	580,00	473,00	1728,00
4		Buchfrg	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
5		Buchfrg	122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
6	MWSt		122,18	532,34	0,00	10,07	69,30	80,00	65,24	238,34
7		Buchfrg	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
8		Buchfrg	1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66
9	Netto		1745,40	3327,16	9,70	62,93	433,10	500,00	407,76	1489,66

C1 enthält beispielsweise die Formel

=`[BUCHFRG.XLS]Feb!$B$8,`

also einen externen Bezug auf die Zelle B8 der Tabelle »Feb« der Mappe BUCHFRG.XLS. Dieser Bezug ist auf der höchsten Ebene (1) normalerweise ebenso ausgeblendet wie der Bezug

=`[BUCHFRG.XLS]Jan!$B$8,`

den Zelle C2 enthält und der auf den Inhalt von Zelle B8 der Tabelle »Jan« zugreift.

Stattdessen ist auf der höchsten Ebene nur die in C3 enthaltene Formel

=`SUMME(C1:C2)`

sichtbar, die die Ergebnisse dieser beiden externen Bezüge mit der *SUMME*-Funktion verknüpft.

Allgemein: Zeile 1 enthält für jede Zelle der ersten Zeile des Bezugsbereichs A8:I10 der Monats-Tabelle »Jan« einen entsprechenden externen Bezug. Zeile 2 enthält entsprechende externe Bezüge auf die einzelnen Zellen des A8:I10 der Monats-Tabelle »Feb«.

Diese beiden Zeilen greifen somit nur auf die einzelnen Werte im Quellbereich zu, führen aber noch keine Konsolidierung dieser Werte durch. Das erfolgt erst in Zeile 3 mit der *SUMME*-Funktion.

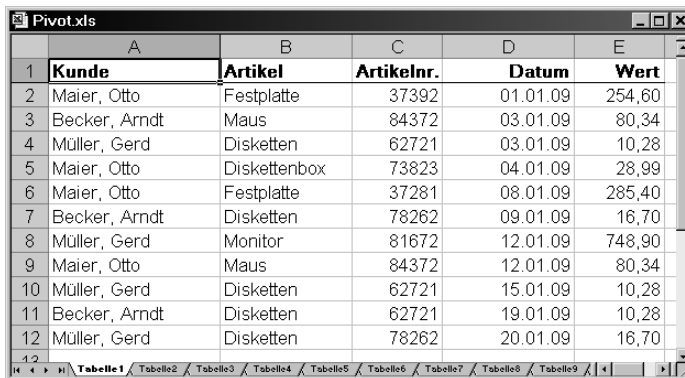
Entsprechend folgt darunter ein Block aus drei Zeilen, der die Konsolidierung für die zweite Zeile der Quellbereiche durchführt, und ein weiterer dreizeiliger Block, der für die Konsolidierung der dritten Zeile der Quellbereiche zuständig ist.

Hätten Sie auch Bezüge auf die restlichen zehn Monats-Tabellen im *Konsolidieren*-Dialogfeld eingefügt, würde jeder dieser Blöcke entsprechend aus 13 Zeilen bestehen, einer Zeile für je eine Zeile eines Quellbereichs und darunter einer dreizehnten Konsolidierungszeile.

16 Pivot-Tabellen

Sollte Ihnen der Begriff Pivot-Tabelle nichts sagen, denken Sie einfach an die vielleicht eher bekannten Kreuz-Tabellen. Pivot-Tabellen ermöglichen jedoch im Gegensatz zu Kreuz-Tabellen nicht nur eine komfortablere Datenanalyse, sondern vor allem das einfache nachträgliche Umstellen und Ändern der Analysen.

Bei einer Kreuz-Tabelle handelt es sich im Grunde um eine einfache Häufigkeitsauszählung. Excel zählt in Ihrer Tabelle zusammen, wie oft das Produkt X bestellt wurde, wie oft das Produkt Y, das Produkt Z und so weiter (Abbildung 16.1).



	A	B	C	D	E
1	Kunde	Artikel	Artikelnr.	Datum	Wert
2	Maier, Otto	Festplatte	37392	01.01.09	254,60
3	Becker, Arndt	Maus	84372	03.01.09	80,34
4	Müller, Gerd	Disketten	62721	03.01.09	10,28
5	Maier, Otto	Diskettenbox	73823	04.01.09	28,99
6	Maier, Otto	Festplatte	37281	08.01.09	285,40
7	Becker, Arndt	Disketten	78262	09.01.09	16,70
8	Müller, Gerd	Monitor	81672	12.01.09	748,90
9	Maier, Otto	Maus	84372	12.01.09	80,34
10	Müller, Gerd	Disketten	62721	15.01.09	10,28
11	Becker, Arndt	Disketten	62721	19.01.09	10,28
12	Müller, Gerd	Disketten	78262	20.01.09	16,70

Abbildung 16.1:
Pivot.xls, »Tabelle1«

In der abgebildeten Tabelle wäre es sicher interessant zu wissen, welche Kunden bevorzugt welche Artikel bestellen. Natürlich können Sie per Hand abzählen, wie oft Herr Maier Disketten bestellt, Müller Monitore und so weiter – Sie können dieses Abzählen jedoch auch Excel überlassen.

Um das Prinzip zu verstehen, müssen Sie in Variablen und Ausprägungen dieser Variablen denken: Spalte A enthält die Variable »Kunde«, die verschiedene Zustände oder Ausprägungen besitzen kann, eben die verschiedenen Kundennamen »Maier, Otto«, »Becker, Arndt« und so weiter.

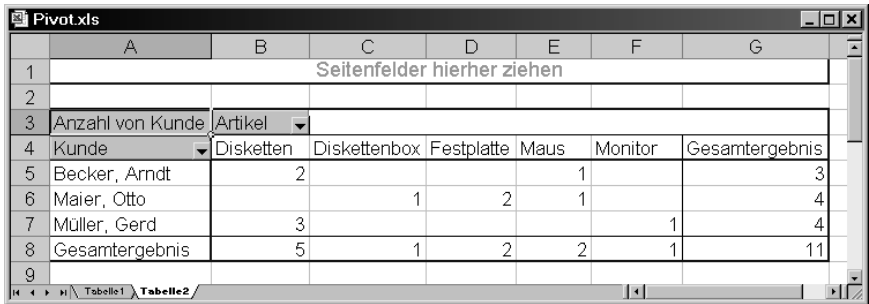
*Variablen und
Ausprägungen*

Spalte B enthält die Variable »Artikel« mit den Ausprägungen »Festplatte«, »Maus« etc. Excel kann auszählen, wie oft bestimmte Ausprägungskombinationen in der Tabelle vorkommen; wie oft beispielsweise Spalte A die

Ausprägung »Becker, Arndt« und gleichzeitig Spalte B die Ausprägung »Disketten« enthält oder anders ausgedrückt: wie oft Herr Becker bei Ihnen Disketten bestellte.

Eine Kreuz-Tabelle besteht in einer Auszählung all dieser Ausprägungskombinationen (Abbildung 16.2).

Abbildung 16.2:
Pivot.xls, »Tabelle2«



	A	B	C	D	E	F	G
1	Seitenfelder hierher ziehen						
2							
3	Anzahl von Kunde	Artikel					
4	Kunde	Disketten	Diskettenbox	Festplatte	Maus	Monitor	Gesamtergebnis
5	Becker, Arndt	2			1		3
6	Maier, Otto		1	2	1		4
7	Müller, Gerd	3				1	4
8	Gesamtergebnis	5	1	2	2	1	11
9							

Sie sehen auf einen Blick, wie oft eine bestimmte Ausprägungskombination vorkommt, beispielsweise dass Herr Becker zweimal Disketten bei Ihnen bestellte und Herr Müller sogar dreimal.

Zeile 1 dieser Pivot-Tabelle enthält alle vorkommenden Ausprägungen der Variablen »Artikel« und Spalte A die verschiedenen Ausprägungen der Variablen »Kunde«, sodass sich eine NxM-Matrix ergibt mit:

- ➔ N = Anzahl der Ausprägungen von Variable »Artikel«
- ➔ M = Anzahl der Ausprägungen von Variable »Kunde«

Im folgenden Kapitel erläutere ich zunächst den Pivot-Tabellen-Assistenten. Er ermöglicht die interaktive Erzeugung derartiger Tabellen per Mausklick und Drag&Drop.

Mit seiner Hilfe ist es im Gegensatz zu einfachen Kreuz-Tabellen möglich, die erstellten Analysen nachträglich auf einfachste Art und Weise nahezu beliebig umzustellen und zu variieren. Sie können beispielsweise jederzeit nachträglich

- ➔ die Spalten und Zeilen der Tabelle miteinander vertauschen,
- ➔ Pivot-Tabellen individuell sortieren, sowohl die Spalten- als auch die Zeilenbeschriftungen, wahlweise in auf- oder in absteigender Reihenfolge,
- ➔ Felder (Variablen) entfernen oder durch andere Variablen ersetzen
- ➔ und die reine Häufigkeitsauszählung durch andere Funktionen wie Mittelwertsbildung etc. ersetzen.

Danach gehe ich auf einige Spezialitäten ein. Ich erläutere, wie Sie multiple Kreuz-Tabellen, die auf mehr als zwei Variablen basieren, in einer einzigen Tabelle mit Unterspalten und Unterzeilen zusammenfassen.

Vor allem aber zeige ich Ihnen, wie Sie mit Pivot-Tabellen auf externe Datenquellen zugreifen, beispielsweise auf eine dBase- oder Access-Datenbank, sodass die resultierende Pivot-Tabelle eine Zusammenfassung der darin enthaltenen Daten liefert.

16.1 Der Pivot-Tabellen-Assistent

Um eine Pivot-Tabelle zu erstellen, wählen Sie DATEN|PIVOTTABLE- UND PIVOTCHART-BERICHT... oder klicken auf das zugehörige Symbol (Kategorie *Daten* des *Anpassen*-Dialogfelds).



In beiden Fällen meldet sich der zugehörige Assistent (Abbildung 16.3).

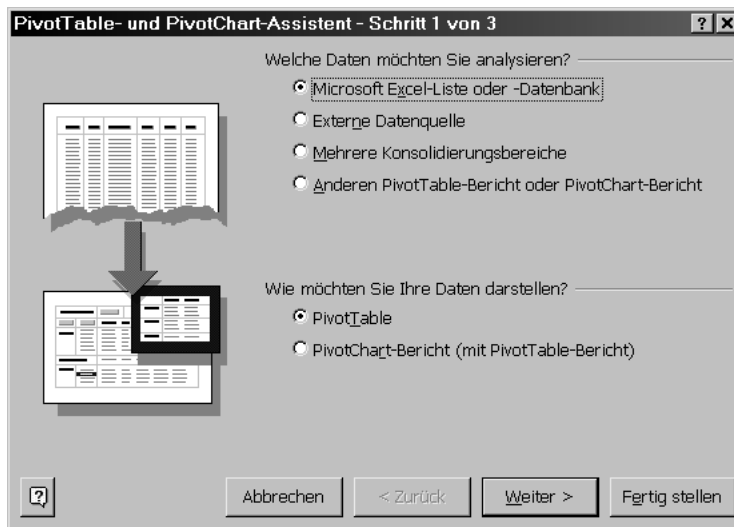
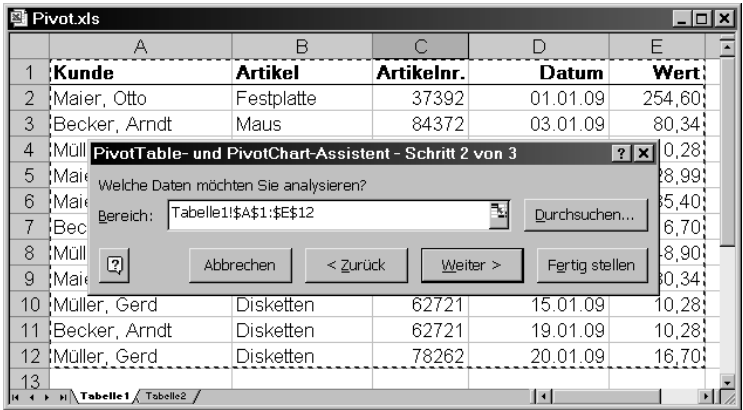


Abbildung 16.3:
Der Assistent

Lassen Sie die vorselektierte Option bitte aktiviert, die besagt, dass als Datenbasis eine Excel-Datenbank oder -liste verwendet und die Daten als Pivot-Tabelle dargestellt werden, und klicken Sie auf *Weiter >* (Abbildung 16.4).

Sie sollen nun die zu verwendende Datenbasis angeben. Sie können einen Bereich selektieren. Sie können sogar mit *Durchsuchen...* das Dateiselektionsfeld öffnen, darin eine andere Mappe mit der in Frage kommenden Datenbasis auswählen und den zu analysierenden Bereich jener Mappe angeben.

Abbildung 16.4:
Datenbasis
angeben



Sie müssen eine Mappe also nicht zwangsläufig öffnen, um eine Pivot-Tabellen-Auswertung mit darin enthaltenen Daten durchzuführen.

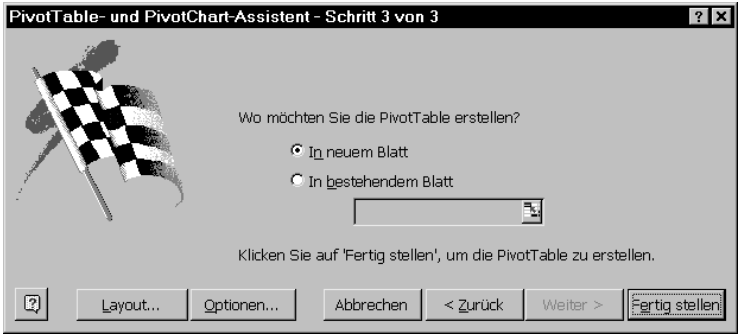
Am einfachsten ist es, wenn sich der Cursor momentan im betreffenden Datenbereich befindet, im Beispiel also irgendwo im Bereich A1:E12 von »Tabelle1« der Mappe PIVOT.XLS. Dann wird dieser Bereich automatisch *inklusive eventuell vorhandener Überschriften* selektiert und vorgegeben und Sie klicken einfach auf *Weiter >*.

f
INFO

Sie werden nun gefragt, ob die Pivot-Tabelle auf den Daten einer bereits bestehenden Pivot-Tabelle erstellt werden soll (die zuvor erstellt wurde und die Daten bereits enthält), um Speicherplatz zu sparen. Antworten Sie am besten mit Ja und selektieren Sie im folgenden Dialogfeld eine beliebige der zu diesem Zweck angebotenen Tabellenblätter, die auf den gleichen Daten basierende Pivot-Tabellen enthalten. Selbst wenn Sie das nun ausgewählte Tabellenblatt später löschen, bleiben die Daten der jetzt zu erzeugenden Tabelle erhalten. Kurz gesagt: Ja spart Speicherplatz und schadet in keiner Weise.

Im letzten Schritt entscheiden Sie, wo die Pivot-Tabelle gespeichert wird (Abbildung 16.5).

Abbildung 16.5:
Erstellungs-
Optionen



- ➔ *In neuem Blatt* legt ein neues Tabellenblatt an und speichert die Pivot-Tabelle darin.
- ➔ Aktivieren Sie *In bestehendem Blatt*, so können Sie darunter eine Zellreferenz eingeben bzw. auf eine Zelle eines beliebigen Blatts zeigen. Die Pivot-Tabelle wird dann in jenes Blatt eingefügt werden und die betreffende Zelle bildet die linke obere Ecke der einzufügenden Auswertung.

Klicken Sie auf *Fertig stellen*, wird die Pivot-Tabelle nun an der betreffenden Position eingefügt, enthält jedoch noch keine Daten (Abbildung 16.6).

16.2 Layout- und Tabellen-Optionen

Fast alle im folgenden beschriebenen Symbole und Befehle sind auch im Kontextmenü verfügbar, das erscheint, wenn Sie mit der rechten Maustaste auf eine Zelle der erzeugten Pivot-Tabelle klicken!

Das im Folgenden beschriebene Ziehen von Feldern aus der Feldliste erfordert, dass eine Zelle des Pivot-Tabellenbereichs selektiert ist. Nur dann ist die Feldliste eingeblendet!

Um in die neue Pivot-Tabelle die gewünschten Daten einzufügen, benutzen Sie die automatisch erscheinende PivotTable-Feldliste. Sie enthält alle Variablenamen, die der benutzte Datenbereich enthält (siehe Abbildung 16.1), im Beispiel »Kunde«, »Artikel«, »Artikelnr.«, »Datum« und »Wert« (Abbildung 16.6).



Per Drag&Drop



Abbildung 16.6:
Variablen angeben

Ziehen Sie die ausgewählten Felder von der Feldliste dorthin ins Pivot-Tabellenschema, wo die betreffenden Felder angezeigt werden sollen.

Da in Bild 16.2 die verschiedenen Ausprägungen von »Artikel« als Spaltenüberschriften verwendet werden, ziehen Sie das Feld *Artikel* zum Abschnitt *Spaltenfelder nach hier ziehen*. Und da die verschiedenen Ausprägungen von »Kunde« als Zeilenüberschriften dargestellt werden sollen, ziehen Sie *Kunde* in den Abschnitt *Zeilenfelder nach hier ziehen*.

Zusätzlich müssen Sie das zu verwendende Datenfeld angeben, jene Variable, deren Ausprägungen ausgezählt werden sollen. Ziehen Sie dazu bitte entweder *Kunde* oder *Artikel* in den Abschnitt *Datenfelder nach hier ziehen*.

Als Resultat erhalten Sie die in Bild 16.2 dargestellte Pivot-Tabelle (Abbildung 16.7).

Abbildung 16.7:
Eingefügte
Variablen

	A	B	C	D	E	F	G
1	Seitenfelder hierher ziehen						
2							
3	Anzahl von Kunde	Artikel					
4	Kunde	Disketten	Diskettenbox	Festplatte	Maus	Monitor	Gesamtergebnis
5	Becker, Arndt	2			1		3
6	Maier, Otto		1	2	1		4
7	Müller, Gerd	3				1	4
8	Gesamtergebnis	5	1	2	2	1	11
9							



Statt ein Feld per Drag&Drop einzufügen, können Sie es in der Feldliste anklicken, danach im Listenfeld darunter den Schemabereich wählen, in den es eingefügt werden soll und es zum Abschluss auf Hinzufügen zu dort einfügen.

Mit dem Layout-
Dialogfeld

Statt mit der Feldliste können Sie das Layout der Pivot-Tabelle bereits während der Erstellung in einem eigenen Dialogfeld festlegen, indem Sie im letzten Schritt des Assistenten auf die Schaltfläche *Layout...* klicken (siehe Bild 16.5).

Im zugehörigen Dialogfeld gibt Excel ganz rechts alle Variablennamen vor, die der Tabellenbereich enthält, »Kunde«, »Artikel« etc. (Abbildung 16.8).

Sie ziehen diese Felder wie bei der Benutzung der Feldliste im Tabellenschema dorthin, wo sie dargestellt werden sollen. Im Beispiel würden Sie entsprechend der Abbildung das Feld *Artikel* zum Schema-Abschnitt *Spalte* ziehen und *Kunde* in den Abschnitt *Zeile*. Zuletzt ziehen Sie *Kunde* ein zweitesmal in das Schema, diesmal zum Abschnitt *Daten*.

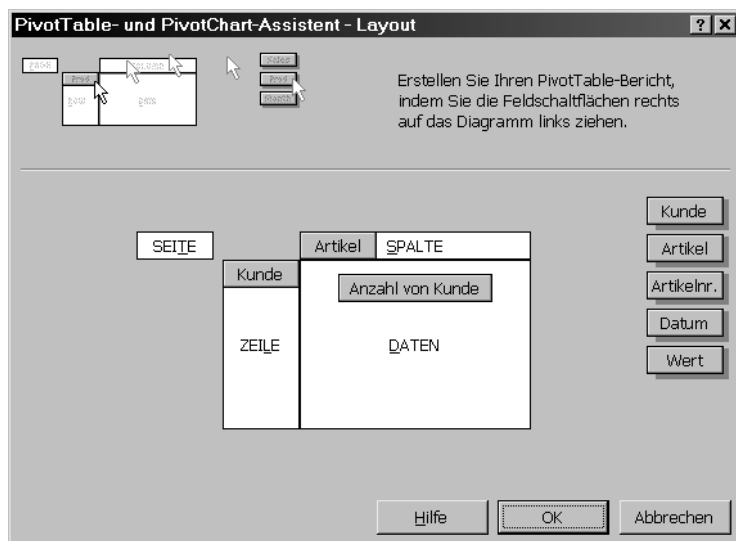


Abbildung 16.8:
Layout-Dialogfeld

Den auch in der erstellten Pivot-Tabelle enthaltenen Abschnitt Seite (dort heißt er Seitenfelder) erläutere ich in Kapitel 16.7, »Anzeige-Einschränkung und Seitenanzeige«.



Zusätzlich können Sie im letzten Schritt verschiedenste Tabellen-Optionen festlegen, indem Sie auf die Schaltfläche *Optionen...* des Assistenten klicken (Abbildung 16.9).

Tabellen-Optionen

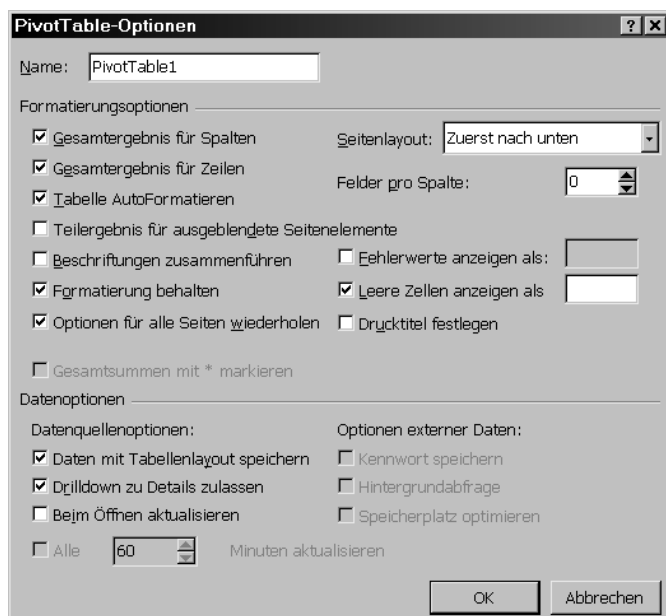


Abbildung 16.9:
Tabellen-Optionen

Unter *Formatierungs-Optionen* können Sie die Optik der Pivot-Tabelle beeinflussen. Deaktivieren Sie beispielsweise das Kontrollkästchen *Gesamtergebnis für Spalten*, so entfällt in Bild 16.2 die Zeile 8, die Gesamtergebnisse für die Spalten B bis F enthält.

Analog dazu würden Sie das Kontrollkästchen *Gesamtergebnis für Zeilen* deaktivieren, wenn Sie die in Spalte G enthaltenen Gesamtergebnisse für die Zeilen 3 bis 6 nicht interessieren.

Bei den Einstellungen der Daten-Optionen ist vor allem das Kontrollkästchen *Daten mit Tabellen-Layout speichern* wichtig. Ist es aktiviert, werden zusammen mit der Pivot-Tabelle auch die Quelldaten gespeichert. Dann ist die resultierende Pivot-Tabelle ständig aktuell, auch nach Änderungen der Quelldaten.

Deaktivieren Sie das Kontrollkästchen, so müssen Sie eine notwendige Aktualisierung später mit DATEN|DATEN AKTUALISIEREN bzw. dem entsprechenden Symbol der PivotTable-Symboleiste manuell einleiten.

16.3 Layouteinstellungen

Einer der größten Vorteile von Pivot-Tabellen besteht darin, dass die Datenanalyse sehr leicht nachträglich verändert werden kann, ohne eine neue Pivot-Tabelle erzeugen zu müssen. Stattdessen bauen Sie einfach die bestehende Pivot-Tabelle um und entfernen Felder daraus oder fügen andere Felder ein.

Momentan zeigt unsere Pivot-Tabelle, wie oft beispielsweise ein Kunde einen bestimmten Artikel bestellte: Excel zählt, wie oft in der Ausgangstabelle identische Ausprägungskombinationen der verwendeten Variablen »Kunde« und »Artikel« vorkommen und stellt beispielsweise fest, dass es darin genau zweimal Kombination »Becker, Arndt« und »Disketten« gibt. Entsprechend erscheint dieser Wert in Zelle B3 der Pivot-Tabelle, dem Schnittpunkt von »Becker, Arndt« und »Disketten«.

Analog dazu besagt der Wert 3 in Zelle B5, dem Schnittpunkt von »Müller, Gerd« und »Disketten«, dass diese Ausprägungskombination genau dreimal in der Ausgangstabelle enthalten ist; oder anders ausgedrückt, dass Herr Müller dreimal Disketten bestellte.

Möglicherweise interessiert Sie jedoch nicht, wie oft Herr Becker oder Herr Müller ein Produkt bestellte, sondern wie hoch der zugehörige Bestellwert war.

Sie wollen also erneut eine Aufteilung der Zeilen in »Kunde« und der Spalten in »Artikel«. Im Datenbereich interessiert Sie nun jedoch nicht die Häufigkeit des Auftretens einer Kombination. Stattdessen soll Excel darin die zugehörige Ausprägung der Variablen »Wert« addieren.

Dazu müssen Sie die Variable »Kunden« aus dem Datenbereich entfernen und durch die Variable »Wert« ersetzen. Sie haben mehrere Möglichkeiten, das Layout der Pivot-Tabelle entsprechend zu ändern:

- ➔ Mit der Feldliste: In der Pivot-Tabelle klicken Sie auf das graue Feld *Anzahl von Kunde*, ziehen es aus dem Pivot-Tabellenbereich heraus und lassen es außerhalb davon fallen, um es daraus zu entfernen. Anschließend klicken Sie in der Feldliste auf das Feld *Wert* und ziehen es in den nun leeren Datenbereich.
- ➔ Mit dem Layout-Dialogfeld: Sie öffnen in der Symbolleiste das Menü PIVOTTABLE und wählen *Assistent...* (oder fügen das abgebildete Symbol, das den Assistenten direkt aufruft, in diese oder eine andere Symbolleiste ein).



Daraufhin erscheint Schritt 3 des Assistenten (siehe Bild 16.5). Sie wählen darin *Layout...*, um das in Bild 16.8 gezeigte *Layout*-Dialogfeld zu öffnen. Im Dialogfeld ziehen Sie *Anzahl von Kunde* aus dem Tabellenschema heraus und fügen dafür *Wert* in den Datenbereich ein (Abbildung 16.10), schließen das Dialogfeld mit OK und wählen im Assistenten *Fertig stellen*.

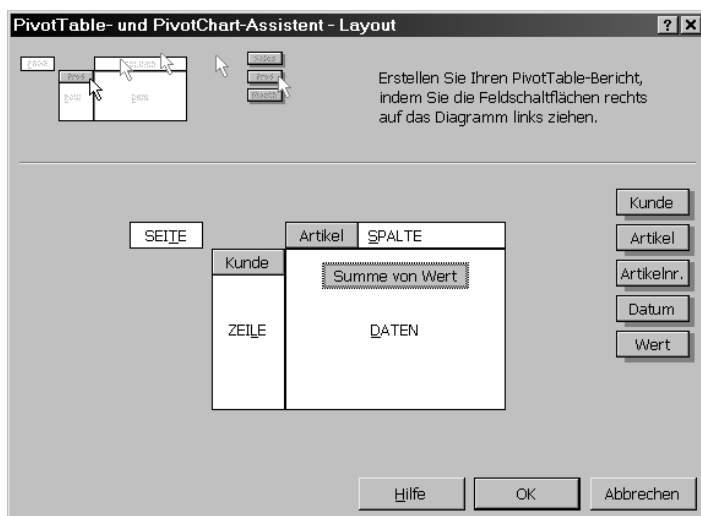


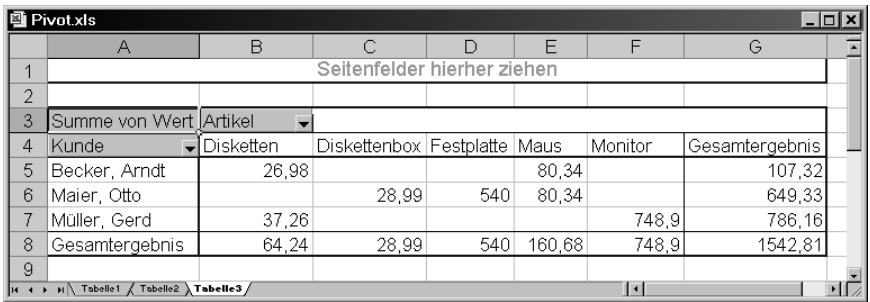
Abbildung 16.10:
Layout verändern

In beiden Fällen ersetzt Excel das Feld *Wert* nicht durch *Anzahl von Wert*, sondern durch *Summe von Wert*, wendet also auf das Feld *Wert* nicht die Funktion *ANZAHL* an, sondern die *SUMME*-Funktion.

Um das zu verstehen, müssen Sie berücksichtigen, dass das zuvor verwendete Datenfeld *Kunde* Texte enthielt. Dann können nur Häufigkeiten ausgezählt werden, das heißt bei jedem weiteren Auftreten einer gerade analysierten Kombination wird ein interner Zähler um 1 erhöht.

Das neue Datenfeld *Wert* enthält jedoch Zahlen und die sollen meist summiert werden. Entsprechend wird der Zähler nun bei jedem Auftreten einer bestimmten Kombination nicht um 1, sondern um den zugehörigen Wert der Variablen »Wert« erhöht und wir erhalten das gewünschte Resultat (Abbildung 16.11).

Abbildung 16.11:
Pivot.xls, »Tabelle3«



	A	B	C	D	E	F	G
1	Seitenfelder hierher ziehen						
2							
3	Summe von Wert	Artikel					
4	Kunde	Disketten	Diskettenbox	Festplatte	Maus	Monitor	Gesamtergebnis
5	Becker, Arndt	26,98			80,34		107,32
6	Maier, Otto		28,99	540	80,34		649,33
7	Müller, Gerd	37,26				748,9	786,16
8	Gesamtergebnis	64,24	28,99	540	160,68	748,9	1542,81
9							

B3 (Kombination »Becker, Arndt« mit »Disketten«) enthält nun den Wert 26,98, weil Excel bei jedem Auftreten der Ausprägungskombination »Becker, Arndt« und »Disketten« die zugehörige Ausprägung der Variablen »Wert« addierte; also die beiden Zahlen 16,70 und 10,28, die sich in den Zeilen 7 und 11 der Ausgangstabelle befinden, jenen Zeilen, die die betreffende Ausprägungskombination enthalten.

Auf die erläuterte Art und Weise können Sie beliebige ins Tabellenschema eingefügte Felder durch andere ersetzen und beispielsweise durch entsprechendes Vertauschen der Felder *Kunde* und *Artikel* die Spalten und Zeilen der Pivot-Tabelle miteinander vertauschen.

Auf die gleiche Weise können Sie die Spalten und Zeilen der Tabelle vertauschen. Dazu ziehen Sie im *Layout*-Dialogfeld *Artikel* in den Abschnitt *Zeile* und *Kunde* genau umgekehrt in den Abschnitt *Spalte*.

Das Ganze funktioniert auch in der Pivot-Tabelle selbst, indem Sie das betreffende Feld einfach zur gewünschten neuen Position ziehen: *Artikel* zum Feld *Kunde* und danach *Kunde* dorthin, wo sich zuvor *Artikel* befand.

Anschließend sind in beiden Fällen die Kundennamen zeilenweise und die Artikelbezeichnungen spaltenweise angeordnet.

16.4 Feldeinstellungen

Außer dem Layout der Pivot-Tabelle können Sie zusätzlich verschiedenste Eigenschaften der darin verwendeten Felder verändern.

Angenommen, Ihnen passen die Feldbezeichnungen *Summe von Wert*, *Kunden* und *Artikel* nicht: Selektieren Sie das betreffende Feld durch Anklicken und editieren Sie seine Bezeichnung anschließend in der Bearbeitungsleiste, ebenso wie Sie das von den Beschriftungen normaler Zellen wie »Maier, Otto«, »Becker, Arndt« etc. her gewohnt sind.

Die höherwertigen Einstellungen erschließen sich erst nach einem Doppelklick auf eines der grauen Felder, beispielsweise auf *Anzahl – Kunde* oder *Artikel* (Alternative: durch einfachen Klick selektieren und danach auf das abgebildete Symbol klicken, das dem Befehl *Feldeigenschaften...* des Menüs PIVOTTABLE der Symbolleiste entspricht).



Dabei muss jedoch zwischen den Einstellungen für Datenfelder (im Beispiel ursprünglich *Anzahl von Kunde*, inzwischen *Summe von Wert*) und für Beschriftungsfelder (*Kunde* und *Anzahl*) unterschieden werden.

16.4.1 Datenfelder

Gehen wir von der in Bild 16.11 gezeigten Pivot-Tabelle aus. Vielleicht interessiert Sie statt der Summe von Herrn Beckers Bestellungen der durchschnittliche Bestellwert. Doppelklicken Sie auf das Datenfeld *Summe von Wert*, um im folgenden Dialogfeld seine Eigenschaften entsprechend zu ändern (Abbildung 16.12).

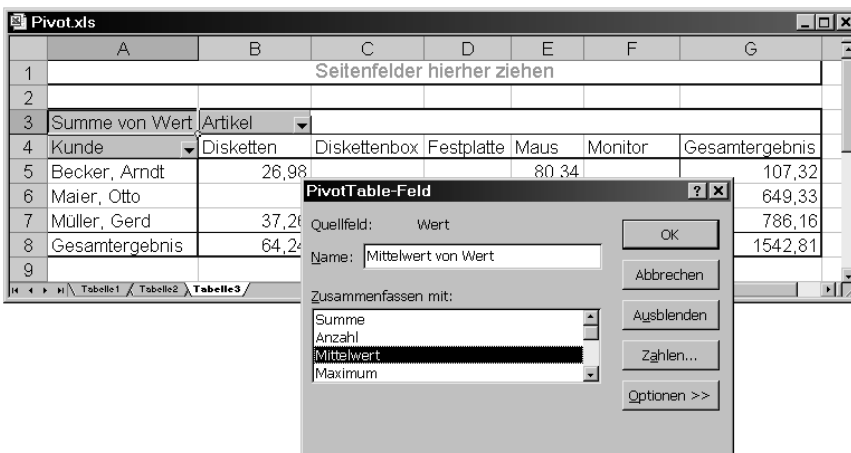


Abbildung 16.12:
Wertefunktion
auswählen

Unter *Name* können Sie die vorgegebene Beschriftung des Felds ändern.

Wichtiger: Selektieren Sie unter *Zusammenfassen mit* entsprechend der Abbildung »Mittelwert« und schließen Sie das Dialogfeld mit OK, so wird die Pivot-Tabelle sofort entsprechend geändert und für dieses Datenfeld werden nun Mittelwerte statt Summen angezeigt (Abbildung 16.13).

Abbildung 16.13:
Pivot.xls, »Tabelle4«

	A	B	C	D	E	F	G
1	Seitenfelder hierher ziehen						
2							
3	Mittelwert von Wert	Artikel					
4	Kunde	Disketten	Diskettenbox	Festplatte	Maus	Monitor	Gesamtergebnis
5	Becker, Arndt	13,49			80,34		35,77333333
6	Maier, Otto		28,99	270	80,34		162,3325
7	Müller, Gerd	12,42				748,9	196,54
8	Gesamtergebnis	12,848	28,99	270	80,34	748,9	140,2554545
9							

#DIV/0

Die resultierende Kreuz-Tabelle enthält mehrfach keinerlei Eintrag, da es in den betreffenden Fällen keine einzige Bestellung und daher auch keinen mittleren Bestellwert gab.

B3 enthält nun den Wert 13,49, entsprechend einem durchschnittlichen Auftragswert von 13,49 DM für Herrn Beckers Diskettenbestellungen.

Der Gesamtdurchschnitt von 12,848 DM für Diskettenbestellungen in Zelle B6 zeigt übrigens, dass Excel nicht verfügbare Werte bei den Globalauswertungen ignoriert und nicht einfach 0 dafür einsetzt (sonst ergäbe sich für Disketten ein weitaus niedrigerer Gesamtdurchschnitt von rund 9 DM).

Weitere Möglichkeiten:

- ➔ Mit der Schaltfläche *Ausblenden* können Sie das aktuelle Datenfeld aus dem Schema entfernen.
- ➔ *Zahlen...* öffnet das Register *Zahlen* des Befehls *FORMAT|ZELLEN...*, mit dem Sie den Wertebereich der Pivot-Tabelle formatieren können.
- ➔ *Optionen >>* erweitert das Dialogfeld (Abbildung 16.14).

Wählen Sie beispielsweise unter *Zusammenfassen mit* »Summe« und unter *Daten zeigen als* »% der Zeile« (Abbildung 16.15).

Die Pivot-Tabelle gibt nun Auskunft über die prozentuale Verteilung der Bestellwerte auf die einzelnen Artikel. Beispielsweise machen die von Herrn Becker bestellten Disketten 25,14% seines Gesamtbestellwerts aus.

Vergleichs-
Optionen

Sollen die in der Pivot-Tabelle enthaltenen Werte mit einer bestimmten Variablen prozentual verglichen werden, selektieren Sie unter *Daten zeigen als* den Eintrag »% von«.



Abbildung 16.14:

Wertfelder-
Optionen

Abbildung 16.15:

Pivot.xls, »Tabelle5«

Unter *Basisfeld* selektieren Sie die Vergleichsvariable, zum Beispiel »Kunde«, und unter *Basiselement* die interessierende Ausprägung der Variablen, zum Beispiel »Müller, Gerd« (Abbildung 16.16).

Das Ergebnis 72,41% in Zelle B3 der folgenden Abbildung bedeutet beispielsweise, dass die Diskettenbestellungen von Herrn Becker wertmäßig 72,41% der Diskettenbestellungen von Herrn Müller ausmachten, dem Vergleichselement (Abbildung 16.17).

Der Fehlerwert #DIV/0! in einigen Zellen bedeutet, dass die zur Ermittlung des prozentualen Anteils notwendige Berechnung (genauer: die zugehörige Division) für diese Zellen nicht durchgeführt werden konnte.

Der Grund ist einfach: Gibt es von Herrn Müller keine Bestellung beispielsweise von »Diskettenboxen«, kann diese Bestellung auch nicht mit jenen der anderen beiden Kunden prozentual verglichen werden.

Abbildung 16.16:
Vergleichs-Optionen

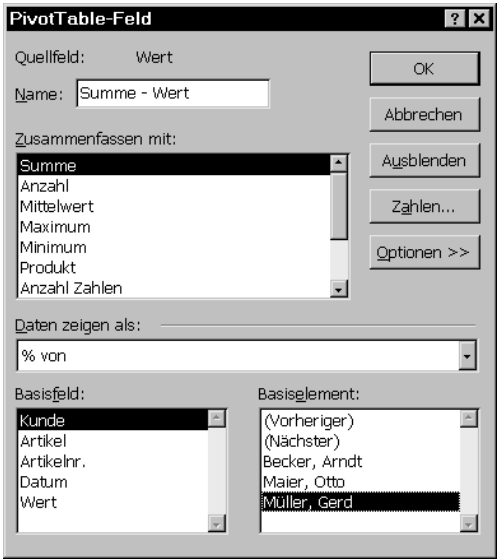


Abbildung 16.17:
Pivot.xls, »Tabelle6«

The screenshot shows a PivotTable in 'Pivot.xls' with the following data:

	A	B	C	D	E	F	G
1	Seitenfelder hierher ziehen						
2							
3	Summe von Wert	Artikel					
4	Kunde	Disketten	Diskettenbox	Festplatte	Maus	Monitor	Gesamtergebnis
5	Becker, Arndt	72,41%	#DIV/0!	#DIV/0!	#DIV/0!	0,00%	13,65%
6	Maier, Otto	0,00%	#DIV/0!	#DIV/0!	#DIV/0!	0,00%	82,60%
7	Müller, Gerd	100,00%	#DIV/0!	#DIV/0!	#DIV/0!	100,00%	100,00%
8	Gesamtergebnis						
9							

16.4.2 Beschriftungs- und Seitenfelder

Doppelklicken Sie auf eines der beiden Beschriftungsfelder, beispielsweise auf *Kunde*, können Sie im folgenden Dialogfeld die Eigenschaften dieses Felds verändern (Abbildung 16.18).

Abbildung 16.18:
Beschriftungsfeld-
Optionen



- ➔ *Ausblenden* entfernt das Beschriftungsfeld aus dem Schema.
- ➔ Unter *Name* können Sie wieder die vorgegebene Feldbeschriftung ändern.

Die Schaltfläche *Weitere...* öffnet ein Dialogfeld zur näheren Festlegung der Sortierung und verschiedener Anzeige-Optionen.

Bei Seitenfeldern können Sie auf die gleiche Weise (per Doppelklick auf ein Seitenfeld) das zugehörige Eigenschafts-Dialogfeld öffnen.



16.5 Mehr als zwei Variablen

Excel kann multiple Kreuz-Tabellen, die auf mehr als zwei Variablen basieren, in einer einzigen Tabelle mit Unterspalten und Unterzeilen zusammenfassen.

Angenommen, Sie wollen nicht nur wissen, wie sich die Bestellungen Ihrer Kunden wertmäßig nach Artikelarten aufteilen, sondern benötigen eine genauere Unterteilung nach Diskettenart und Festplattentyp (im Blatt »Tabelle1« der Mappe PIVOT.XLS existieren jeweils zwei Diskettensorten und Festplattentypen mit unterschiedlichen Artikelnummern).

Sie beginnen wie in den vorhergehenden Beispielen, ziehen jedoch außer *Artikel* zusätzlich auch das Feld *Artikelnr.* in den Abschnitt *Spalte* (Abbildung 16.19).

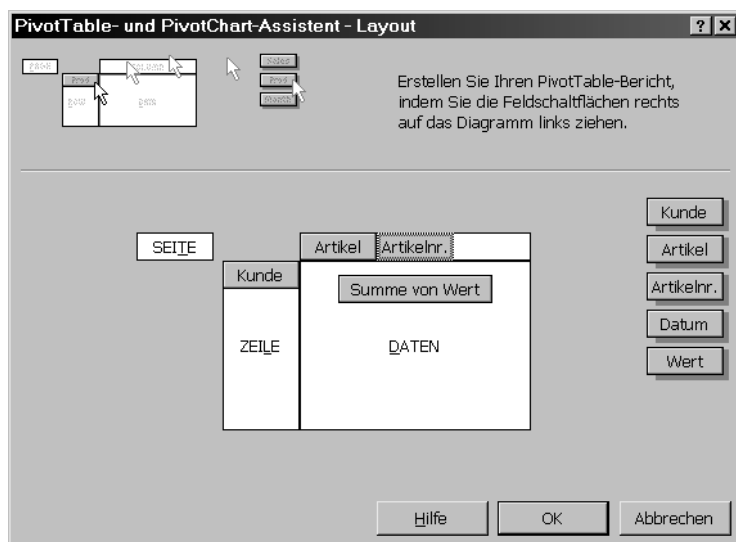
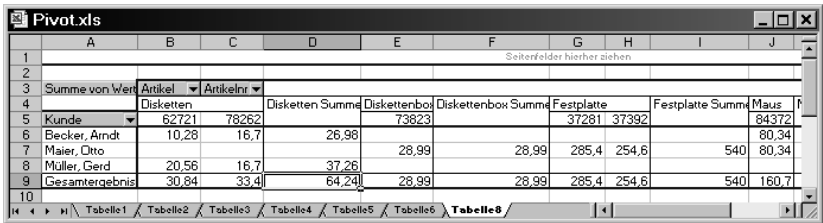


Abbildung 16.19:
Untergeordnete
Spaltenrubriken

Excel kennt nun zwei Spaltenrubriken: die Hauptrubrik »Artikel«, die dazu führt, dass in der resultierenden Kreuz-Tabelle die Ausprägungen »Disketten«, »Diskettenbox« etc. dieser Variablen die Spaltenüberschriften bilden, und die Unterrubrik »Artikelnr.«, in die diese Spaltenrubriken weiter unterteilt werden (Abbildung 16.20).

Abbildung 16.20:
Pivot.xls, »Tabelle8«



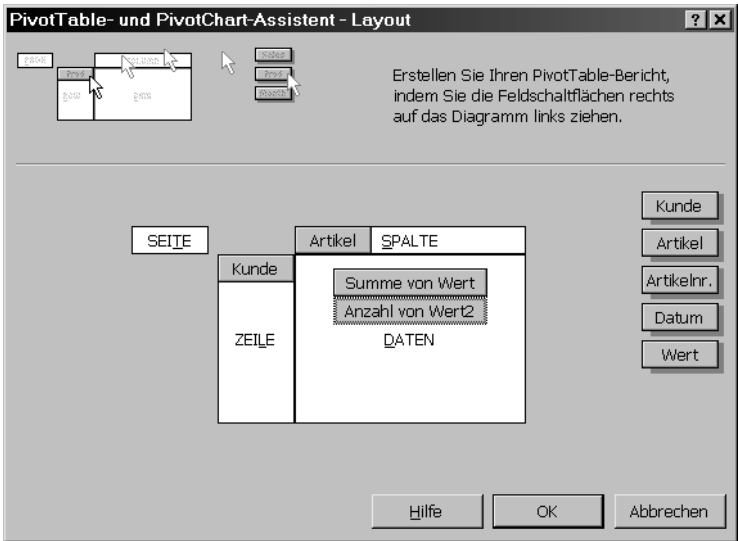
	Artikel	Artikelnr.							
Summe von Wert	Disketten	Disketten Summe	Diskettenbox	Diskettenbox Summe	Festplatte	Festplatte Summe	Maus		
Kunde	62721	78262		73823	37281	37392	84372		
Becker, Arndt	10,28	16,7	26,98				80,34		
Maier, Otto				28,99	28,99	285,4	254,6	540	80,34
Müller, Gerd	20,56	16,7	37,26						
Gesamtergebnis	30,84	33,4	64,24	28,99	28,99	285,4	254,6	540	160,7

Beispielsweise ist die Spaltenüberschrift »Disketten« in die beiden Disketten-Artikelnummern 62721 und 78262 unterteilt. Die Spaltenüberschrift »Festplatte« ist analog dazu unterteilt.

Sie können beliebig viele solcher Unterrubriken sowohl für die Zeilen als auch für die Spalten der Pivot-Tabelle bilden. Die Reihenfolge der Variablen in den Listenfeldern entscheidet darüber, ob eine Variable die Haupt- oder eine Unterrubrik einer darüber angeordneten Variablen bildet.

Auch in den Datenbereich können Sie mehrere Felder einfügen, beispielsweise zweimal das Feld *Wert*. Beide Male wird wie gewohnt die *SUMME*-Funktion angewendet. Per Doppelklick auf eins der beiden Felder können Sie das jedoch ändern und für das betreffende Feld beispielsweise die *ANZAHL*-Funktion auswählen (Abbildung 16.21).

Abbildung 16.21:
Datenbereich aus mehreren Variablen



Sie erhalten eine Pivot-Tabelle, in der die Bestellungen nach Artikeln und Kunden unterteilt sind und die gleich zwei Auswertungen für diese Bestellungen enthält: sie informiert über die Summen der betreffenden Bestellwerte und gleichzeitig über die Anzahl der jeweils bestellten Artikel (Abbildung 16.22).

Kunde	Daten	Artikel	Disketten	Diskettenbox	Festplatte	Maus	Monitor	Gesamtergebnis
Becker, Arndt	Summe von Wert	26,98				80,34		107,32
	Anzahl von Wert2	2				1		3
Maier, Otto	Summe von Wert		28,99		540	80,34		649,33
	Anzahl von Wert2		1		2	1		4
Müller, Gerd	Summe von Wert	37,26					748,9	786,16
	Anzahl von Wert2	3					1	4
Gesamt: Summe von Wert		64,24	28,99		540	160,68	748,9	1542,81
Gesamt: Anzahl von Wert2		5	1		2	2	1	11

Abbildung 16.22:
Pivot.xls, »Tabelle9«

16.6 Sortieren und Gliedern

Sie können Pivot-Tabellen mit den bereits früher erläuterten Mitteln sortieren und gliedern.

Excel sortiert die Spalten- und Zeilenbeschriftungen einer Pivot-Tabelle zunächst automatisch in aufsteigender Reihenfolge. Daher befindet sich »Disketten« vor »Diskettenbox«, dieser Eintrag wiederum vor »Festplatte« und so weiter.

Sortieren

Entsprechend bildet »Becker« die oberste Tabellenzeile, »Maier« ist darunter und »Müller« befindet sich in der letzten Zeile.

Wäre es Ihnen lieber, wenn zuerst genau umgekehrt »Müller«, dann »Maier« und zuletzt »Becker« so kommt, selektieren Sie irgendeine dieser Beschriftungen und klicken in der Standardleiste auf das Symbol zum Sortieren in absteigender Reihenfolge (Abbildung 16.23).



Ebenso einfach können Sie die Tabelle statt nach Beschriftungen nach Werten sortieren: Sie selektieren irgendeine Wertezelle jener Spalte, nach deren Inhalt sortiert werden soll, beispielsweise in der Spalte »Disketten«, und klicken das Symbol zum Sortieren in aufsteigender Reihenfolge an: Die Zeile mit dem kleinsten Wert in der aktiven Spalte bildet anschließend die oberste Tabellenzeile, die Zeile mit dem zweitkleinsten Wert die zweite Spalte und so weiter.



Abbildung 16.23:
Sortierreihenfolge
ändern

	A	B	C	D	E	F	G
1	Seitenfelder hierher ziehen						
2							
3	Anzahl von Kunde	Artikel					
4	Kunde	Disketten	Diskettenbox	Festplatte	Maus	Monitor	Gesamtergebnis
5	Müller, Gerd	3				1	4
6	Maier, Otto		1	2	1		4
7	Becker, Arndt	2			1		3
8	Gesamtergebnis	5	1	2	2	1	11

Ausprägungen
aus-/einblenden

Eine weitere Möglichkeit besteht darin, uninteressante Ausprägungen auszu-
blenden. Sollen im Blatt »Tabelle3« von PIVOT.XLS weder die Ausprägungen
»Diskettenbox« noch »Maus« der Variablen »Artikel« angezeigt werden, kli-
cken Sie auf den abwärts zeigenden Pfeil im Feld *Artikel* (Abbildung 16.24).

Abbildung 16.24:
Elemente von
»Artikel«
ausblenden

	A	B	C	D	E
1	Seitenfelder hierher ziehen				
2					
3	Summe von Wert	Artikel			
4	Kunde				Gesamtergebnis
5	Becker, Arndt				26,98
6	Maier, Otto				540
7	Müller, Gerd				48,9
8	Gesamtergebnis				1353,14

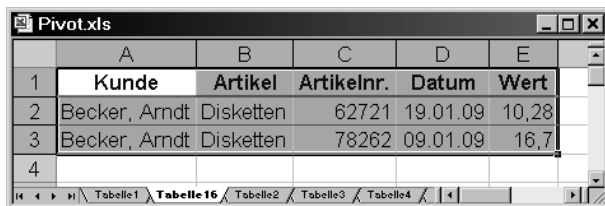
☐ (Alle anzeigen)
☒ Disketten
☐ Diskettenbox
☒ Festplatte
☐ Maus
☒ Monitor
OK Abbrechen

Deaktivieren Sie im zugehörigen Dialogfeld die Einträge »Diskettenbox«
und »Maus«, werden diese beiden Spalten nicht mehr angezeigt, können
aber jederzeit mit der gleichen Technik wieder eingeblendet werden.

Details ein-/
ausblenden

Ebenso einfach können Sie zusätzliche Details einblenden. Angenommen,
Sie interessieren sich für einen ganz bestimmten Wert einer Pivot-Tabelle.
Sie wollen zum Beispiel wissen, wie in der letzten Abbildung der Wert 2 für
»Becker, Arndt« und »Disketten« zu Stande kommt.

Kein Problem: Ein Doppelklick auf diese Zelle genügt und eine neue Tabelle
wird erzeugt, die alle Datensätze enthält, auf denen der betreffende Wert
basiert, im Beispiel also die beiden Diskettenbestellungen von Herrn Arndt
(Abbildung 16.25).



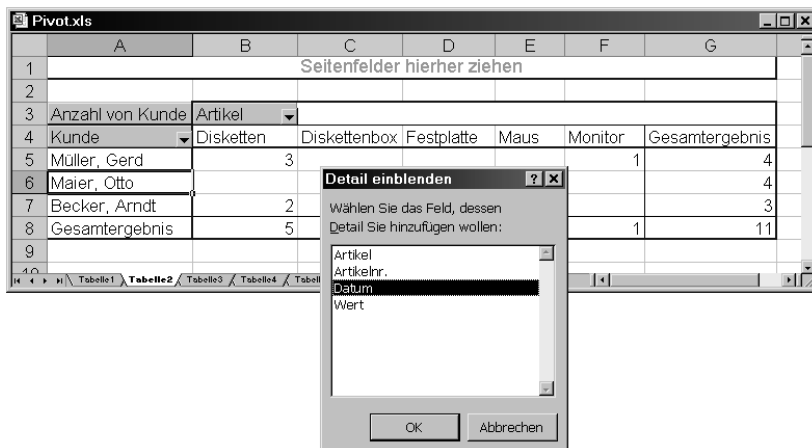
	A	B	C	D	E
1	Kunde	Artikel	Artikelnr.	Datum	Wert
2	Becker, Arndt	Disketten	62721	19.01.09	10,28
3	Becker, Arndt	Disketten	78262	09.01.09	16,7
4					

Abbildung 16.25:
Zu Grunde liegende
Datensätze selektieren

Zusätzlich können Sie auch in der Pivot-Tabelle selbst weitere Details ein- oder ausblenden. Möglicherweise interessiert Sie das Datum, an dem eine Bestellung vorgenommen wurde.

Gliedern

Kein Problem: Doppelklicken Sie auf irgendeinen Kunden, beispielsweise auf »Maier, Otto«, oder selektieren Sie die Zelle per einfachem Klick und klicken Sie danach auf das abgebildete Symbol (weitere Alternative: mit rechter Maustaste anklicken und im Kontextmenü GRUPPIERUNG UND DETAIL ANZEIGEN|DETAIL ANZEIGEN wählen) (Abbildung 16.26).

	A	B	C	D	E	F	G
1	Seitenfelder hierher ziehen						
2							
3	Anzahl von Kunde	Artikel					
4	Kunde	Disketten	Diskettenbox	Festplatte	Maus	Monitor	Gesamtergebnis
5	Müller, Gerd	3				1	4
6	Maier, Otto						4
7	Becker, Arndt	2					3
8	Gesamtergebnis	5				1	11
9							

Detail einblenden

Wählen Sie das Feld, dessen Detail Sie hinzufügen wollen:

- Artikel
- Artikelnr.
- Datum
- Wert

OK Abbrechen

Abbildung 16.26:
Detail einblenden

Selektieren Sie als einzublegendes Detail das gewünschte Feld »Datum« und schließen Sie das Dialogfeld mit OK (Abbildung 16.27).

In der Pivot-Tabelle erscheint das zusätzliche Feld *Datum*, allerdings wird diese zusätzliche Detailinformation nur für den Kunden »Maier, Otto« angezeigt.



Interessiert Sie nicht mehr, wann Herr Maier die einzelnen Artikel bestellte, doppelklicken Sie erneut auf »Maier, Otto« (oder benutzen – nach einfachem Klick – das abgebildete Symbol) und dieses Detail wird wieder ausgeblendet (weitere Alternative: mit rechter Maustaste anklicken und im Kontextmenü GRUPPIERUNG UND DETAIL ANZEIGEN|DETAIL AUSBLENDEN wählen).

Abbildung 16.27:
Eingeblendetes
Kundendetail

Pivot.xls							
	A	B	C	D	E	F	G
1	Seitenfelder hierher ziehen						
2							
3	Anzahl von Kunde		Artikel				
4	Kunde	Datum	Disketten	Diskettenbox	Festplatte	Maus	Monitor
5	Müller, Gerd		3				1
6	Maier, Otto	01.01.09			1		
7		04.01.09		1			
8		08.01.09			1		
9		12.01.09				1	
10	Maier, Otto Summe			1	2	1	
11	Becker, Arndt		2			1	
12	Gesamtergebnis		5	1	2	2	1
13							
14	Tabelle1 / Tabelle2 / Tabelle3 / Tabelle4 / Tabelle5 / Tabelle6 / Tabelle9						

Doppelklicken Sie auf »Müller, Gerd«, wird das Datums-Detail entsprechend für jenen Kunden eingeblendet – und zwar ohne vorhergehende Frage nach dem hinzuzufügenden Detailfeld, da *Datum* der Pivot-Tabelle nun ja bereits hinzugefügt ist.

Analog dazu können Sie auf einen beliebigen Artikel wie »Maus« klicken und danach auch den Artikeln Details hinzufügen, beispielsweise die Artikelnummer.

Anschließend genügt ein Doppelklick auf ausgewählte Artikel wie »Diskettenbox« oder »Maus«, um jederzeit die zugehörigen Artikelnummern einzublenden (Abbildung 16.28).

Abbildung 16.28:
Eingeblendetes
Artikel-Detail

Pivot.xls							
	A	B	C	D	E	F	G
1	Seitenfelder hierher ziehen						
2							
3	Summe von Wert		Artikel	Artikelnr.			
4			Disketten	Diskettenbox	Diskettenbox Summe	Festplatte	Maus
5	Kunde	Datum		73823			
6	Becker, Arndt		26,98				84372
7	Maier, Otto	01.01.09				254,6	
8		04.01.09		28,99	28,99		
9		08.01.09				285,4	
10		12.01.09					80,34
11	Maier, Otto Summe			28,99	28,99	540	80,34
12	Müller, Gerd	03.01.09	10,28				
13		12.01.09					
14		15.01.09	10,28				
15		20.01.09	16,7				
16	Müller, Gerd Summe		37,26				
17	Gesamtergebnis		64,24	28,99	28,99	540	160,68
18	Tabelle1 / Tabelle2 / Tabelle3 / Tabelle4 / Tabelle5 / Tabelle6 / Tabelle9						

16.7 Anzeige-Einschränkung und Seitenanzeige

Mit dem Schema-Abschnitt *Seite* können Sie eine Art Anzeigekriterium definieren (Abbildung 16.29).

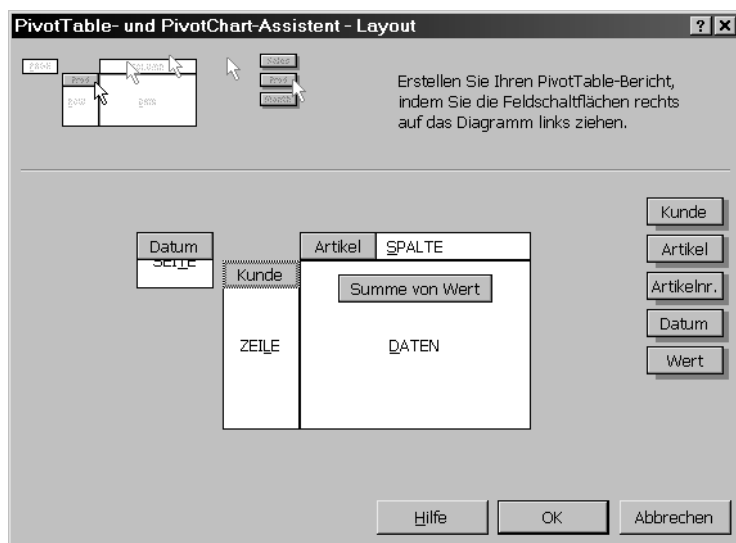


Abbildung 16.29:
Abschnitt »Seite«

Diese Definitionen entsprechen prinzipiell jenen, die zur Erzeugung des Blatts »Tabelle3« von PIVOT.XLS verwendet wurden: Die Datenbasis wird nach Artikeln und Kunden gruppiert und eine Übersicht der zugehörigen Bestellwerte erzeugt.

Allerdings wurde zusätzlich in den Abschnitt *Seite* das Feld *Datum* eingefügt. Anschließend befindet sich entsprechend ganz oben in der Pivot-Tabelle, im Abschnitt *Seite*, das Feld *Datum*, und direkt daneben der Eintrag »(Alle)« und ein Listenfeldpfeil.

Die zugehörige Liste enthält alle verfügbaren Elemente der Variablen »Datum« (Abbildung 16.30).

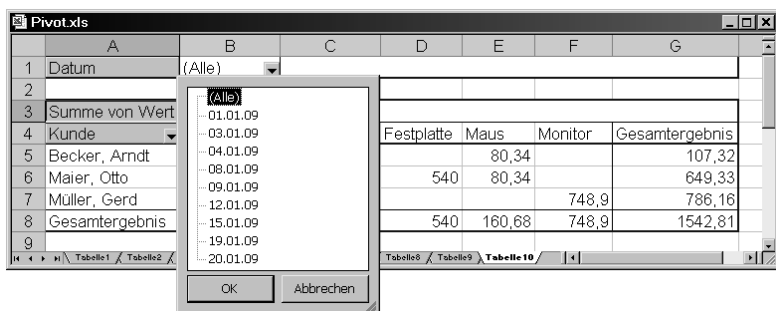


Abbildung 16.30:
Kriterium
auswählen

Selektieren Sie eines davon, zum Beispiel den 8.1.09, werden nur die unter diesem Datum notierten Bestellungen angezeigt (Abbildung 16.31).

Abbildung 16.31:
Anzeige-
beschränkung

	A	B	C
1	Datum	08.01.09	
2			
3	Summe von Wert	Artikel	
4	Kunde	Festplatte	Gesamtergebnis
5	Maier, Otto	285,4	285,4
6	Gesamtergebnis	285,4	285,4
7			

»(Alle)« würde wieder alle Bestellungen anzeigen.



»Datum« oder ein beliebiges anderes als Kriterium zu verwendendes Feld können Sie auch ohne das Layout-Dialogfeld einfügen, indem Sie es aus der Feldliste in den Bereich Seitenfelder hierher ziehen ziehen und dort fallen lassen.



Dieses Symbol (Befehl SEITEN ANZEIGEN... im Listenfeld PIVOT TABLE der Symbolleiste) besitzt eine sehr interessante Aufgabe. Es listet alle Felder auf, die sich im Abschnitt *Seite* befinden, im Beispiel also das soeben dort eingefügte Feld »Datum« (Abbildung 16.32).

Abbildung 16.32:
Seitenvariable
auswählen



Selektieren Sie einen Eintrag und wählen Sie **OK**, werden für jedes Element der betreffenden Variablen separate Auswertungen erstellt.

Für jede Auswertung wird eine zusätzliche Tabelle in der aktiven Arbeitsmappe angelegt, die den Namen des betreffenden Elements erhält (Abbildung 16.33).

	A	B	C
1	Datum	19.01.09	
2			
3	Summe von Wert	Artikel	
4	Kunde	Disketten	Gesamtergebnis
5	Becker, Arndt	10,28	10,28
6	Gesamtergebnis	10,28	10,28
7			

Abbildung 16.33:
Erzeugte Tabellen

Die abgebildete Tabelle bezieht sich auf das Element »19.1.2009« und besitzt daher auch den Namen dieses Elements. Sie zeigt nur die unter diesem Datum vermerkten Bestellungen an. Entsprechend zeigen die beiden benachbarten Tabellen nur die Bestellungen vom 20.1.2009 bzw. vom 15.1.2009 an.

16.8 Zugriff auf externe Datenquellen

Mit Pivot-Tabellen können Sie auch auf externe Datenbanken zugreifen. Ich setze voraus, dass Sie das Kapitel 14, »Microsoft Query – Zugriff auf externe Datenbanken«, lasen. Dann sind Sie mit Abfragen und Datenquellen vertraut und können die nun erläuterten Schritte nachvollziehen.



16.8.1 MS Query

Um aus externen Daten eine Pivot-Tabelle zu erstellen, rufen Sie mit DATEN|PIVOT-TABELLE- UND PIVOTCHART-BERICHT... den Pivot-Tabellen-Assistenten auf.

Unter *Welche Daten möchten Sie analysieren?* selektieren Sie die Option *Externe Datenquelle*. Der zweite Schritt sieht so aus (Abbildung 16.34).

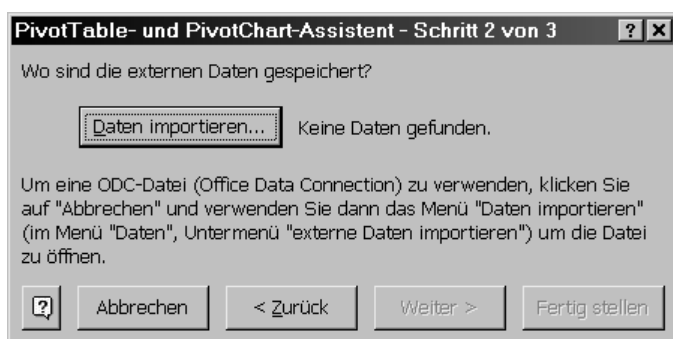


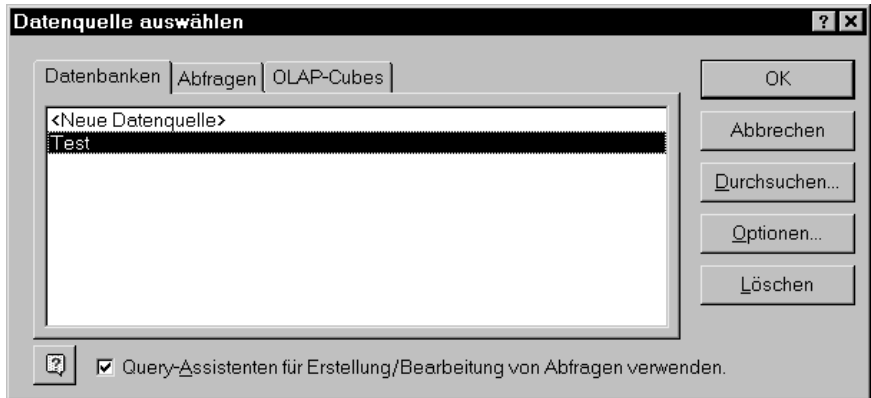
Abbildung 16.34:
Daten abrufen

Daten importieren... startet Query und öffnet das Dialogfeld *Datenquelle auswählen*. Existiert bereits eine zuvor erstellte und gespeicherte Abfrage, deren Daten Sie in die Pivot-Tabelle übernehmen wollen, wählen Sie darin *Durchsuchen...* und öffnen die betreffende Abfrage.



Gibt es noch keine Abfrage, die die gewünschten Daten auswählt, müssen Sie zunächst die gewünschte Datenquelle selektieren bzw. definieren. Mehr dazu in Kapitel 14, »Microsoft Query – Zugriff auf externe Datenbanken« (Abbildung 16.35).

Abbildung 16.35:
Daten abrufen



Nach Auswahl der Datenquelle selektieren Sie im Query-Assistenten die benötigten Tabellen und erstellen eine Abfrage, die die gewünschten Daten aus diesen Tabellen extrahiert.

Für das folgende Beispiel verwende ich die in Kapitel 14 beschriebene Abfrage, die Daten aus den dBase-Tabellen LIEFERAN.DBF, PRODUKT.DBF und LIEFPROF.DBF auswählt und Auskunft darüber gibt, von welchen Lieferanten Sie welche Produkte beziehen und wie viel Stück Sie davon jeweils auf Lager haben (Abbildung 16.36).

Haben Sie die drei dBase-Tabellen in die Abfrage eingefügt, sollten Sie im letzten Schritt des Query-Assistenten *Daten in Microsoft Query bearbeiten oder ansehen* wählen, um Query selbst aufzurufen und dort die Beziehungen zwischen den Tabellen durch Ziehen der entsprechenden Beziehungslinien bekannt zu geben.

Danach wählen Sie in Query *DATEI|DATEN AN MICROSOFT EXCEL ZURÜCKGEBEN* (Abbildung 16.37).

Excel wird wieder aktiviert und im Dialogfeld erscheint entsprechend dieser Abbildung der Hinweis »Daten gefunden«.

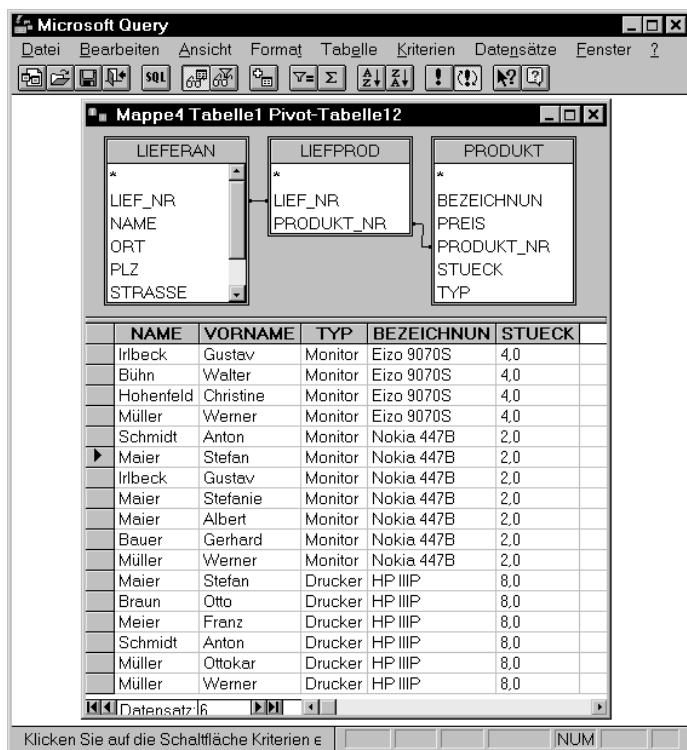


Abbildung 16.36:
Abfrage

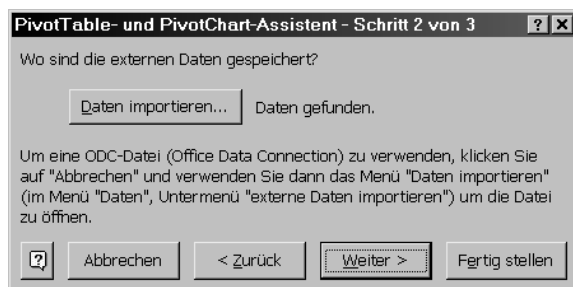


Abbildung 16.37:
Geglückte Daten-
übergabe

Weiter > führt zum letzten Schritt, in dem Sie wie üblich mit *Layout...* die in der Abfrage verwendeten Felder in das Pivot-Schema einfügen können (Abbildung 16.38).

Als Ergebnis erhalten Sie die folgende Pivot-Tabelle (Abbildung 16.39).

Um die Abfrage nachträglich zu verändern, aktivieren Sie einfach wieder den Pivot-Tabellen-Assistenten, gehen zum zweiten Schritt zurück (siehe Bild 16.34) und wählen *Daten importieren...*: Query wird aufgerufen, Sie können die Filter- und Sortierkriterien verändern und die geänderte Abfrage speichern und/oder *Daten an Microsoft Excel zurückgeben* wählen.

Abbildung 16.38:
Abfragefelder
einfügen

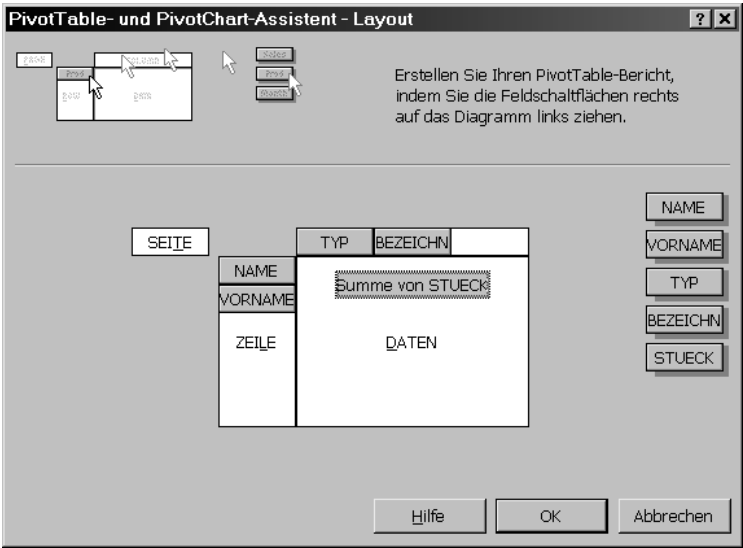


Abbildung 16.39:
Resultierende
Pivot-Tabelle

NAME	VORNAME	Drucker	Drucker Summe	Monitor	Monitor Summe	Gesamtergebnis
Bauer	Gerhard	HP IIIP		Eizo 9070S	2	2
Bauer	Summe		2	Nokia 447B	2	2
Braun	Otto		8			8
Braun	Summe		8			8
Bühn	Walter			4	4	4
Bühn	Summe			4	4	4
Hohenfeld	Christine			4	4	4
Hohenfeld	Summe			4	4	4
Irlbeck	Gustav			4	2	6
Irlbeck	Summe			4	2	6
Maier	Albert			2	2	2
Maier	Stefan	8	8	2	2	10
Maier	Stefanie			2	2	2
Maier	Summe	8	8	6	6	14
Meier	Franz	8	8			8
Meier	Summe	8	8			8
Müller	Ottokar	8	8			8
Müller	Werner	8	8	4	2	14
Müller	Summe	16	16	4	2	22
Schmidt	Anton	8	8	2	2	10
Schmidt	Summe	8	8	2	2	10
Gesamtergebnis		48	48	16	14	78



Dieses Symbol aktualisiert die Pivot-Tabelle, indem die Datenquelle erneut abgefragt wird.



Beachten Sie bitte, dass Sie Daten aus externen Datenbanken nicht nur als Pivot-Tabelle, sondern auch im Rohzustand in eine Excel-Tabelle einfügen können (siehe Kapitel 14.2, »Externe Daten in Excel importieren«).

16.8.2 OLAP-Cubes

Statt Daten aus einer relationalen Datenbank wie Access oder dBase wie erläutert direkt als Pivot-Tabelle zu importieren, können Sie einen OLAP-Cube zwischenschalten.

Ein OLAP-Cube (OLAP = On-Line Analytical Processing) stellt praktisch eine Untermenge der in einer Datenbank enthaltenen Daten dar.

Er enthält nur die zuvor per Query ausgewählten Daten, und zwar nicht im Rohformat, sondern in bereits vorausgewerteter Form. Bei einer Artikel-Tabelle enthält er beispielsweise nicht jeden einzelnen Datensatz dieser Tabelle und somit auch nicht den Verkaufspreis und die Stückzahl jedes Artikels, sondern nur die Summen oder Mittelwerte dieser Datensatzfelder.

Der große Vorteil dieser Vorauswertung besteht in der Reduzierung der zu übertragenden Datenmengen.

Angenommen, eine Datenbank enthält Artikel, Aufträge und Kunden und Sie benötigen einen Überblick über die Höhe der Aufträge Ihrer Kunden. Dann interessiert Sie nicht jede einzelne Bestellung von Herrn Becker und Herrn Maier, sondern die Summen der nach »Kunde« gruppierten Bestellungen. Solche gruppierten Auswertungen können Sie wie zuvor erläutert mit Pivot-Tabellen vornehmen, die über Query direkt auf die betreffende Datenbank zugreifen.

Der Haken: Die Datenbank ist möglicherweise riesig und befindet sich noch dazu nicht direkt auf Ihrem Rechner, sondern auf einem Server, mit dem Ihr Rechner über ein relativ langsames Netzwerk verbunden ist.

Geschwindigkeit

Um die benötigten Auftragssummen zu bilden, muss jedoch jeder einzelne Auftragsdatensatz an Excel übertragen werden, damit Excel die darin enthaltenen Bestellwerte nach »Kunde« gruppiert summieren kann.

Mit einem zwischengeschalteten OLAP-Cube ist die Auswertung erheblich schneller. Der OLAP-Cube enthält ja bereits die vom OLAP-Server (der Datenbank) vorausgewerteten Daten, also die benötigten Summen. Statt der einzelnen Datensätze werden daher nur diese in der Pivot-Tabelle darzustellenden Summen an Excel übertragen, was die zu übertragenden Datenmengen drastisch reduziert.

Ein weiterer Vorteil besteht in der Möglichkeit des Offline-Zugriffs auf die Daten. Ein normaler OLAP-Cube ist eine Abfragedefinition, die wie jede Query- oder Webabfrage gespeichert werden kann. Wird die Abfrage ausgeführt, werden die aktuellen Daten aus der Datenbank geholt.

Offline-Zugriff

Zusätzlich können Sie jedoch auch offline Cubes erstellen – vorausgesetzt, die Datenquelle, auf die Sie zugreifen, unterstützt dieses Feature!

Ein Offline-Cube stellt eine Art Mini-Datenbank dar, in der die benötigten Daten selbst gespeichert sind. Das ist möglich, da OLAP-Cubes ja wie erläutert keine einzelnen Datensätze enthalten, sondern bereits die gruppierten Summen, Mittelwerte oder andere Auswertungen. Der Platzbedarf eines Offline-Cubes hält sich daher in Grenzen.

Der Vorteil: Sie können den Cube auch dann zur Erstellung von Pivot-Tabellen verwenden, wenn Sie gerade nicht per Netzwerk oder Modem mit der Datenquelle verbunden sind. Die Daten sind dann zwar möglicherweise teilweise veraltet, aber immerhin können Sie den Offline-Cube jederzeit aktualisieren lassen, wenn Sie wieder mit der Datenquelle verbunden sind.

Erstellung

Die Erstellung eines OLAP-Cubes setzt die Definition einer Datenquelle zum Zugriff auf die betreffende Datenbank voraus.



Setzen wir voraus, Sie haben eine Datenquelle zum Zugriff auf die dBase-Dateien der Begleit-CD definiert (siehe Kapitel 14, »Microsoft Query – Zugriff auf externe Datenquellen«). Es gibt zwei Wege, daraus einen OLAP-Cube zu erstellen:



1. *Sie wählen DATEN|EXTERNE DATEN IMPORTIEREN|NEUE ABFRAGE ERSTELLEN..., selektieren die Datenquelle zum Zugriff auf die dBase-Dateien und wählen die Daten der Sie interessierenden dBase-Dateien aus. Ich beschrieb all das bereits in Kapitel 14.2, »Externe Daten in Excel importieren«. Im Gegensatz zu jenen Erläuterungen beenden Sie die Erstellung der Abfrage jedoch nicht mit Daten an Microsoft Excel zurückgeben. Stattdessen wählen Sie die Option Einen OLAP-Cube aus dieser Abfrage erstellen.*
2. *Sie wählen wie bei der Erzeugung einer neuen Pivot-Tabelle DATEN|PIVOTTABLE- UND PIVOTCHART-BERICHT... und danach in Query ebenfalls die Datenquelle und die fraglichen Daten aus. Bevor Sie Query mit DATEI|DATEN AN MICROSOFT EXCEL ZURÜCKGEBEN beenden, wählen Sie jedoch noch den Query-Befehl DATEI|OLAP-CUBE ERSTELLEN... .*

Nehmen wir als Beispiel die in Kapitel 14 beschriebene und auch im vorhergehenden Kapitel angesprochene Abfrage, mit der Daten aus den dBase-Tabellen LIEFERAN.DBF, PRODUKT.DBF und LIEFPROF.DBF ausgewählt und Auskunft darüber gegeben wird, von welchen Lieferanten Sie welche Produkte beziehen und wie viel Stück Sie davon jeweils auf Lager haben.

Haben Sie diese Abfrage erstellt und wählen Sie zum Abschluss wie erläutert die Option *Einen OLAP-Cube aus dieser Abfrage erstellen* bzw. den Query-Befehl `DATEI|OLAP-CUBE ERSTELLEN...`, so meldet sich nun der OLAP-Cube-Assistent (Abbildung 16.40).

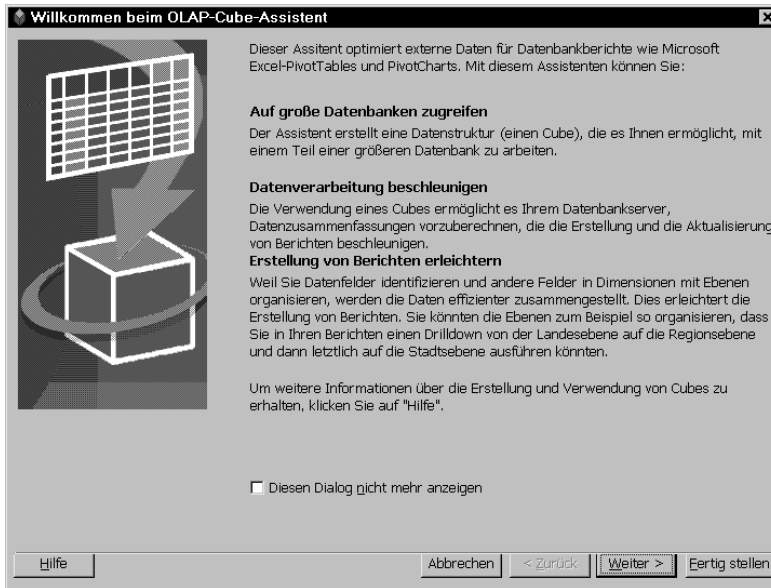


Abbildung 16.40:
OLAP-Cube-
Assistent

Im ersten Schritt wählen Sie aus, mit welchen Auswertungsfunktionen die in der Abfrage enthaltenen numerischen Felder ausgewertet werden. Der Assistent erkennt die numerischen Felder und gibt als Auswertungsfunktion jeweils »Summe« vor, summiert das betreffende Feld also, wenn Sie diese Voreinstellung übernehmen (Abbildung 16.41).

Stattdessen können Sie im Listenfeld jedoch auch eine andere Funktion wie »Anzahl«, »Min.« oder »Max.« auswählen.

Im folgenden Schritt definieren Sie die Dimensionen des Cubes, in anderen Worten: Sie legen die Gruppierungsebenen fest (Hauptgruppe, Untergruppe etc.) (Abbildung 16.42).

Die im Beispiel wichtigste Gruppierungsebene ist vermutlich »Typ«, also der Artikeltyp (Monitor, Drucker etc.). Dann ziehen Sie »Typ« aus der linken Hälfte zu »(Felder hier ablegen...)«.

Daraufhin wird diese Dimension oder Gruppierungsebene in der rechten Fensterhälfte eingefügt und Sie können darunter weitere Gruppierungsebenen einfügen, zum Beispiel »Name« (der Name des Lieferanten des Artikels) und »Bezeichnung« (die Artikelbezeichnung) (Abbildung 16.43).

Abbildung 16.41:
OLAP-Cube-
Assistent, Schritt 1

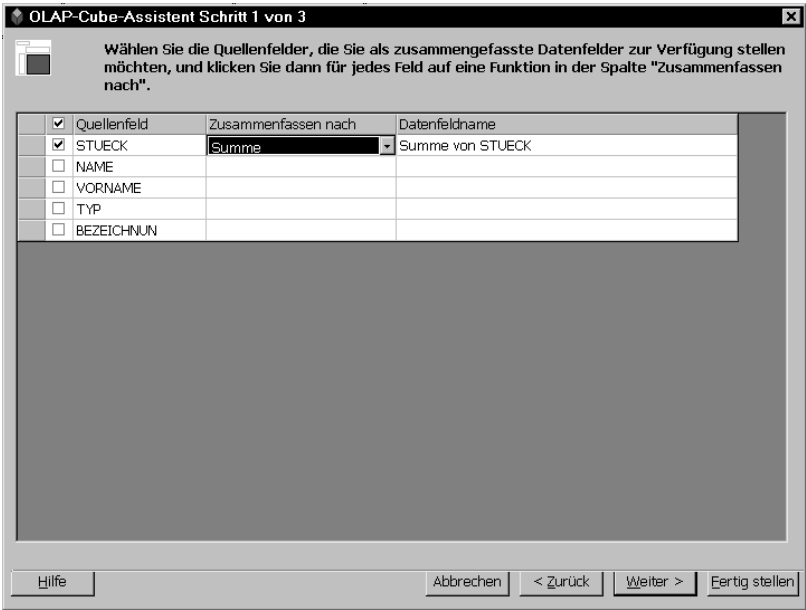
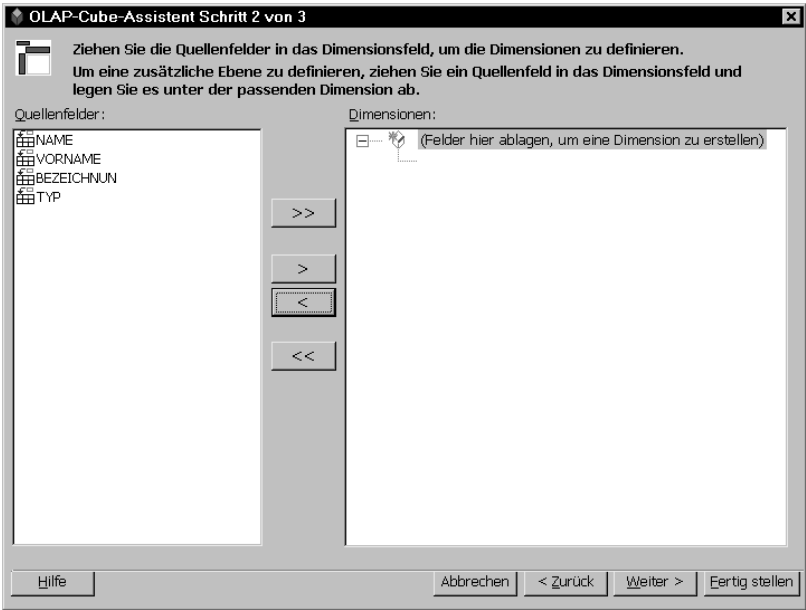


Abbildung 16.42:
OLAP-Cube-
Assistent, Schritt 2



Im letzten Schritt legen Sie den zu erstellenden Cube-Typ fest (Abbildung 16.44).

Die beiden ersten Optionen erstellen beide einen Cube, der nur aus der Cube-Definition selbst besteht und sich nur in der Art und Weise der Aktualisierung unterscheidet.

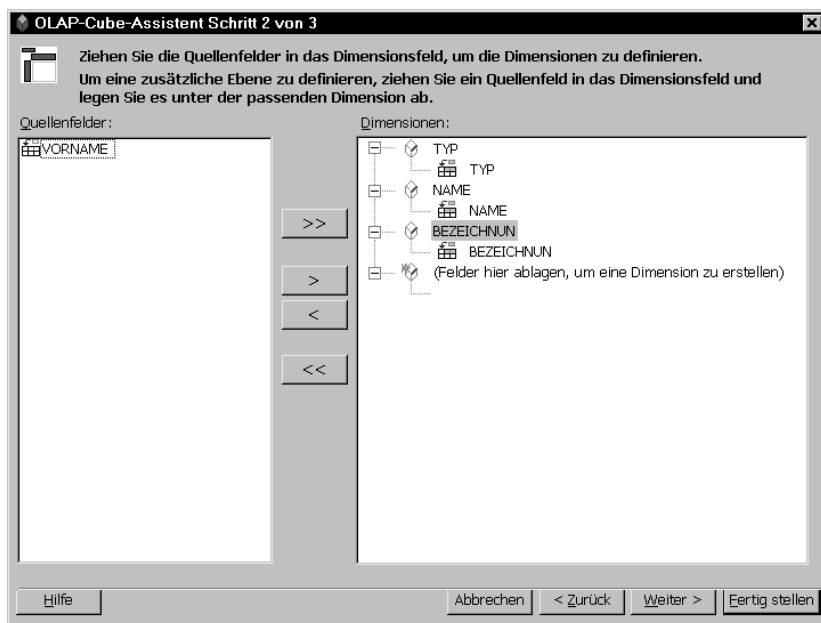


Abbildung 16.43:
Dimensionen
definieren

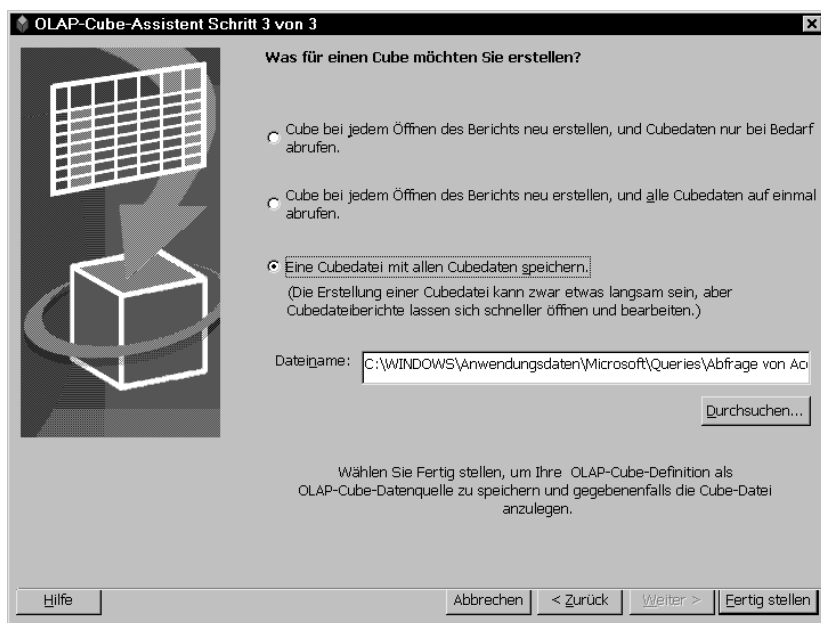


Abbildung 16.44:
OLAP-Cube-
Assistent, Schritt 3

Die letzte Option erzeugt den erläuterten Offline-Cube, dessen Daten auch ohne Verbindung zur Datenquelle verfügbar sind. Wählen Sie diese Option, so können Sie darunter auswählen, wo und unter welchem Namen die Cube-Datei (Endung .CUB) gespeichert werden soll.

Egal, welche Option Sie auswählen, anschließend werden Sie auf alle Fälle aufgefordert, den Namen und Ort anzugeben, unter dem die Cube-Definition selbst gespeichert wird, also die Abfragedefinition, die im Falle eines Cubes die Endung .OQY besitzt.

Danach wird der Cube erstellt und Sie können die dadurch ausgewählten Daten an Excel zurückgeben. Die Daten werden als Pivot-Tabelle dargestellt und Sie werden daher nun wie immer gefragt, wo Sie die Pivot-Tabelle erstellen sollen, ob in einem neuen oder in einem bereits bestehenden Blatt. Das Resultat ist eine Pivot-Tabelle, in deren Abschnitte Sie die verfügbaren Daten einfügen können (Abbildung 16.45).

Abbildung 16.45:
Resultierende Pivot-
Tabelle

	NAME	Bauer	Braun	Böhn	Hohenfeld	Irlbeck	Maier	Meier	Müller	Schmidt	Gesamtergebnis
Drucker			8				8	8	16	8	48
Monitor		2		4	4	6	6		6	2	30
Gesamtergebnis		2	8	4	4	6	14	8	22	10	78

Was in dieser Pivot-Tabelle jedoch nicht zur Verfügung steht (auch nicht im *Layout*-Dialogfeld des Pivot-Assistenten) das Feld »Stück«. Stattdessen enthält der Cube wie erläutert nur die Summen der einzelnen Artikelstückzahlen, abhängig von der von Ihnen gewählten Gruppierung (hier: Gruppierung nach »Typ« und »Name« (des Lieferanten)).

Die resultierende Pivot-Tabelle können Sie natürlich wie gewohnt manipulieren, aktualisieren etc.



Zusätzlich können Sie mit DATEN|EXTERNE DATEN IMPORTIEREN|DATEN IMPORTIEREN... die zuvor gespeicherte Cube-Abfrage (Endung OQY) jederzeit erneut ausführen, um die darin enthaltenen Daten zur Erstellung einer völlig neuen Pivot-Tabelle zu verwenden.



Um einen bestehenden OLAP-Cube zu verändern, rufen Sie Query auf und öffnen mit DATEI|ÖFFNEN... die Cube-Abfragedatei (die bei der Erstellung des Cubes gespeicherte Datei mit der Endung OQY). Daraufhin meldet sich automatisch der OLAP-Cube-Assistent, mit dem Sie den bestehenden Cube nun verändern können.

16.9 Pivot-Diagramme und Berichte

Dieses Symbol erzeugt aus einer Pivot-Tabelle ein Pivot-Diagramm (ebenso wie der Befehl PIVOTCHART im Kontextmenü bzw. in der Symbolleiste) (Abbildung 16.46).

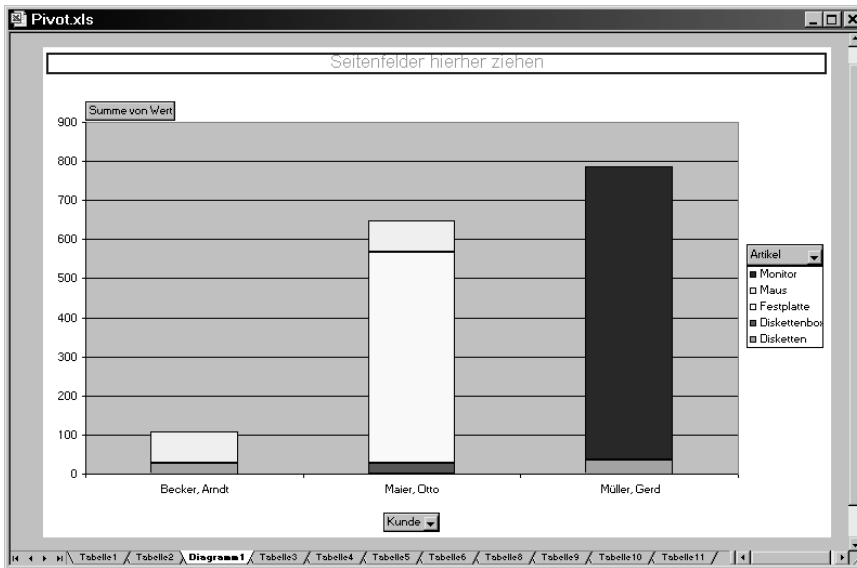


Abbildung 16.46:
Pivot-Diagramm

Den Umgang mit Diagrammen erläutere ich noch ausführlich. Wichtig ist auf alle Fälle, dass die in der Abbildung sichtbaren Felder *Summe von Wert* etc. im Gegensatz zu normalen Diagrammen die gewohnte Funktion besitzen, ein Doppelklick darauf also die erläuterten Dialogfelder öffnet.

Dieses Symbol ermöglicht die in Kapitel 6.11, »AutoFormatieren«, erläuterte Formatierung eines Datenbereichs – hier: der Pivot-Tabellenbereich – auf Knopfdruck (ebenso wie der Befehl BERICHT FORMATIEREN... der Symbolleiste) (Abbildung 16.47).



Abbildung 16.47:
Bericht

AutoFormat

Qrt	Ernte	Raum	Zone	Eink.
Q3				888
	Korn			333
		USA		
			NO	111
			NW	222
		JPN		555
			SO	555

Bericht 1

Qrt	Ernte	Raum	Zone	Eink.
Q3				888
	Korn			888
		USA		333
			NO	111
			NW	222
		JPN		555
			SO	555

Bericht 3

Qrt	Ernte	Raum	Zone	Eink.
Q3				888
	Korn			888
		USA		333
			NO	111
			NW	222
		USA	Summe	333
	Korn	Summe		333

Bericht 5

Qrt	Ernte	Raum	Zone	Eink.
Q3				888
	Korn			888
		USA		888
			NO	111
			NW	222
		USA	Summe	333
	JPN			555
		SO		555

Bericht 6

Qrt	Ernte	Raum	Zone	Eink.
Q3				888
	Korn			888
		USA		333
			NO	111
			NW	222
		USA	Summe	333
	JPN			555
		SO		555

Bericht 4

OK

Abbrechen

17 Mehrfachoperationen

Im folgenden Kapitel erläutere ich Mehrfachoperationen, die sich ähnlich wie der Szenario-Manager für Analysen der Art »Was wäre wenn...« eignen. Auch Mehrfachoperationen variieren einen oder mehrere Parameter in den von Ihnen angegebenen Schritten und speichern die Auswirkungen auf die interessierenden Formeln im Arbeitsblatt.

Interessiert Sie beispielsweise der Wert, den eine in der Tabelle gespeicherte Formel zur Renditeberechnung bei verschiedenen Zinssätzen ergibt, können Sie alle in Frage kommenden Zinssätze im Arbeitsblatt eintragen, beispielsweise 6%, 7%, 8%, 9% und 10%, und Excel mitteilen, dass es beim Durchrechnen der Formel der Reihe nach diese Zinssätze verwenden soll.

Excel rechnet die Formel daraufhin der Reihe nach mit diesen Zinssätzen durch und speichert die verschiedenen Resultate in entsprechend vielen Zellen der Tabelle, sodass Sie anschließend alle Ergebnisse auf einen Blick sehen und miteinander vergleichen können.

Wie ich zeigen werde, kann Excel auch die Auswirkungen auf beliebig viele Formeln gleichzeitig testen und entsprechend die verschiedenen Ergebnisse jener Formeln bei unterschiedlichen Zinssätzen in der Tabelle speichern.

Statt mit nur einer können derartige Mehrfachoperationen sogar mit zwei Variablen durchgeführt werden, beispielsweise mit unterschiedlichen Zinssätzen und unterschiedlichem Anlagekapital.

Excel rechnet die maßgebliche Formel daraufhin mit allen möglichen Variationen dieser beiden Parameter durch und speichert die Ergebnisse erneut dauerhaft in der Tabelle, sodass Sie sie jederzeit wieder einsehen können.

17.1 Mit einem Eingabefeld

Nehmen wir an, Sie wollen 10000 DM festverzinslich anlegen und haben die Wahl unter verschiedenen Anleihen mit einer Laufzeit von jeweils zehn Jahren, aber verschiedenem Risiko (unterschiedliche Bonität der Schuldner) und entsprechend höherem oder niedrigerem Zinssatz.

*Kapital-
berechnung*

Sie wollen wissen, welches Kapital sich nach zehn Jahren bei den verschiedenen Zinssätzen angesammelt hat. Ohne Berücksichtigung des Zinseszins-effekts genügt dazu eine sehr primitive Formel:

$$\text{Endkapital} = \text{Startkapital} + \text{Laufzeit} * (\text{Startkapital} * \text{Zins})$$

Bei einer Anleihe mit einem Zinssatz von 8,5% ergäbe sich:

$$\text{Endkapital} = 10000 + 10 * (10000 * 8,5\%) = 18500$$

Um auch den Zinseszins-effekt zu berücksichtigen, sollte man den Taschenrechner jedoch besser beiseite legen und auf Excel zurückgreifen, genauer: auf die Funktion ZW. Vereinfacht dargestellt lautet der Aufruf dieser Funktion

`ZW(Zins;Jahre;0;-Startkapital).`

Entsprechend würde die Formel

`=ZW(8,5%;10;0;-10000)`

das Endkapital (22609,83 DM) ermitteln, das sich aus einem Startkapital von 10000 DM nach zehn Jahren bei jährlichen Zinszahlungen von 8,5% ergibt.

Angewandt auf unser Problem ergibt sich folgende Tabelle, die noch auf herkömmliche Art und Weise arbeitet, ohne Mehrfachoperation (Abbildung 17.1).

Abbildung 17.1:
Mehrfach.xls,
»Tabelle1«

	A	B	C
1	Startkapital		10000
2	Laufzeit in Jahren		10
3	Zinssatz		8,50%
4	Endkapital		22609,83

C4 enthält die Formel

`=ZW(C3;C2;0;-C1)`

Sie errechnet das Endkapital, das bei dem in C3 enthaltenen Zinssatz und einer Laufzeit von C2 Jahren aus einem Startkapital C1 resultiert – vorausgesetzt, die jährlich ausgezahlten Zinsen werden mit dem gleichen Zinssatz wieder angelegt.

Um das Ganze mit verschiedenen Zinssätzen durchzuspielen, müssen Sie in C3 immer wieder einen anderen Zinssatz eintragen und sich anschauen, welches Endkapital jeweils in C4 resultiert.

Zum Glück ist es recht einfach, dieses Blatt auf eine Mehrfachoperation umzustellen, die dieses Durchtesten *selbstständig* übernimmt.

Getestet werden soll der Wert, den die Formel in C4 bei verschiedenen Zinssätzen ergibt. Dazu müssen Sie die zu testenden Zinssätze unterhalb dieser Formelzelle und in der Spalte links daneben eingeben, zum Beispiel Zinssätze von 8% bis 10% in 0,25%-Schritten (siehe Bild 17.2).

Anschließend selektieren Sie den Bereich B4:C13, der diese Testwerte (B5:B13), den momentanen Formelwert (C4) und die von Excel zu ermittelnden Resultate bei anderen Zinssätzen (C5:C13) enthält, und wählen DATEN|TABELLE... (Abbildung 17.2).

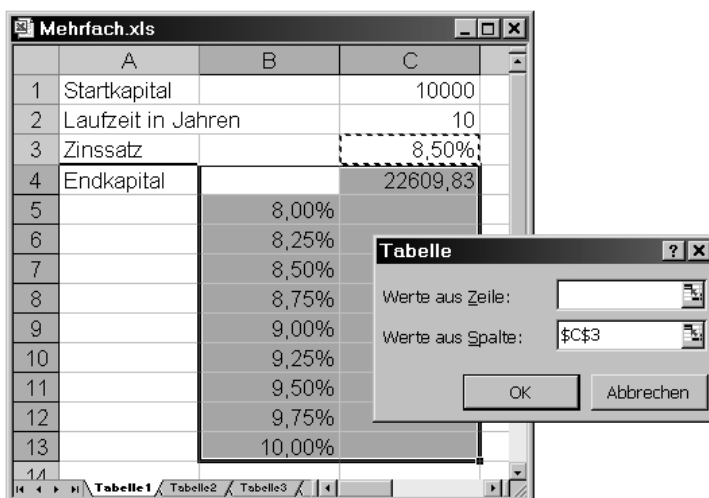


Abbildung 17.2:
Vorbereitung der
Mehrfachoperation

Nun müssen Sie Excel mitteilen, wo sich der Wert befindet, der Schritt für Schritt durch die Testwerte ersetzt wird, und ob sich diese Werte in der linken Spalte oder in der obersten Zeile des selektierten Bereichs befinden.

Der zu ersetzende Wert, der Zinssatz 8,5%, befindet sich in C3, und die Testwerte sollen aus der linken Bereichsspalte übernommen werden. Also aktivieren Sie *Werte aus Spalte*, geben den Bezug C3 durch Zeigen ein, wählen OK und sehen danach auf einen Blick, welches Endkapital sich bei einem bestimmten Zinssatz ergibt (Abbildung 17.3).

Der interne Ablauf: Excel geht davon aus, dass sich die Formel, um die es geht, in der oberen rechten Ecke des markierten Bereichs befindet, in C4. Sie gaben an, dass der in dieser Formel enthaltene Bezug C3 durch die Testwerte ersetzt werden soll und – weil Sie *Werte aus Spalte* wählten – dass sich diese Werte in der linken Spalte des markierten Bereichs befinden.

Abbildung 17.3:
Mehrfach.xls,
»Tabelle2«

	A	B	C
1	Startkapital		10000
2	Laufzeit in Jahren		10
3	Zinssatz		8,50%
4	Endkapital		22609,83
5		8,00%	21589,25
6		8,25%	22094,2391
7		8,50%	22609,8344
8		8,75%	23136,2334
9		9,00%	23673,6367
10		9,25%	24222,2486
11		9,50%	24782,2761
12		9,75%	25353,9301
13		10,00%	25937,4246

Excel setzt in die Formel nun Schritt für Schritt die einzelnen Werte aus der linken Spalte des selektierten Bereichs statt des Bezugs C3 ein, berechnet das Ergebnis und speichert die Resultate in der rechten Bereichsspalte.

Bei der spaltenweisen Anordnung der Testwerte gilt:

- ➔ Die linke Spalte enthält ab der zweiten Zeile des markierten Bereichs die Testwerte.
- ➔ In der ersten Zeile sowie eine Spalte rechts neben den Testwerten befindet sich die Formelzelle.
- ➔ Die rechte Spalte nimmt unterhalb der Formelzelle die Resultate dieser Formel mit den verschiedenen Testwerten auf.

Für die zeilenweise Anordnung gilt:

- ➔ Die obere Zeile enthält ab der zweiten Spalte des markierten Bereichs die Testwerte.
- ➔ In der ersten Spalte sowie eine Zeile unter den Testwerten befindet sich die Formelzelle.
- ➔ Die untere Zeile nimmt neben der Formelzelle die Resultate dieser Formel mit den verschiedenen Testwerten auf.

Angewandt auf das Beispiel sind bei zeilenweiser Anordnung die neun Zinssätze im Bereich D3:L3 einzugeben.

Vor Anwahl von TABELLE... muss der Bereich C3:L4 selektiert werden, der die Testwerte, die Resultatzellen und die Formel selbst enthält.

Im Dialogfeld ist die Bezugszelle C3 unter *Werte aus Zeile:* anzugeben, um Excel klarzumachen, dass die Testwerte diesmal zeilenweise angeordnet sind (Abbildung 17.4).

	A	B	C	D	E	F	G	H	I	J	K	L
1	Startkapital		10000									
2	Laufzeit in Jahren		10									
3	Zinssatz		8,50%	8,00%	8,25%	8,50%	8,75%	9,00%	9,25%	9,50%	9,75%	10,00%
4	Endkapital		22610	21589	22094	22610	23136	23674	24222	24782	25354	25937
5												

Abbildung 17.4:
Mehrfach.xls,
»Tabelle3«

17.2 Testen mehrerer Formeln

Excel kann die Auswirkungen auf beliebig viele Formeln gleichzeitig testen. Sind die Testwerte in einer Spalte enthalten, müssen Sie die Testformeln nebeneinander anordnen, sodass jeweils eine Spalte die Resultate einer Formel aufnimmt.

Ein Beispiel: Sie erweitern Ihre Anleihen-Analyse um eine Rubrik »Gewinn (absolut)«, die die Differenz zwischen End- und Startkapital enthalten soll. Dazu verwenden Sie folgende Formel

=C4-C1.

Außerdem fügen Sie eine Rubrik »Gewinn (prozentual)« ein, die den Gewinn in Prozent vom Startkapital ausweist und zwar mit der Formel

=100*C4/C1.

Die Testwerte, also die Zinssätze von 8% bis 10%, ordnen Sie wie zuvor in Spalte B untereinander an und die drei Formeln in der obersten Zeile nebeneinander (Abbildung 17.5).

Die drei Formeln befinden sich in C4, D4 und E4. Die zugehörigen Spalten verwendet Excel für die Anzeige der resultierenden Werte, wenn in diesen Formeln der Bezug C3 der Reihe nach durch die Testwerte ersetzt wird.

Vor Anwahl von TABELLE... muss der gesamte Bereich B4:E13 mit den Testwerten, den Formeln und dem Leerraum für die Resultate selektiert werden. Als Bezugsfeld wird wieder C3 angegeben. Das Resultat sieht so aus wie Bild 17.6.

Zum Beispiel ergibt sich bei einem Zinssatz von 10% ein Endkapital von 25937,42 DM, ein absoluter Gewinn von 15937,42 DM und eine Erhöhung des Startkapitals auf 259,37% vom Ausgangswert.

Abbildung 17.5:
Mehrere zu
testende Formeln
nebeneinander

The screenshot shows an Excel window titled 'Mehrfach.xls'. The active sheet is 'Tabelle4'. The data table is as follows:

	A	B	C	D	E
1	Startkapital		10000		
2	Laufzeit in Jahren		10		
3	Zinssatz		8,50%	Gewinn (absolut)	Gewinn (prozentual)
4	Endkapital		22609,83	12609,83	226,10
5		8,00%			
6		8,25%			
7		8,50%			
8		8,75%			
9		9,00%			
10		9,25%			
11		9,50%			
12		9,75%			
13		10,00%			
14					

The 'Tabelle' dialog box is open, showing 'Werte aus Zeile:' and 'Werte aus Spalte:' fields. The 'Werte aus Spalte:' field contains '\$C\$3'. The 'OK' and 'Abbrechen' buttons are visible.

Abbildung 17.6:
Mehrfach.xls,
»Tabelle4«

The screenshot shows the same Excel window as before, but now the data table is fully populated with calculated values:

	A	B	C	D	E
1	Startkapital		10000		
2	Laufzeit in Jahren		10		
3	Zinssatz		8,50%	Gewinn (absolut)	Gewinn (prozentual)
4	Endkapital		22609,83	12609,83	226,10
5		8,00%	21589,25	11589,25	215,89
6		8,25%	22094,24	12094,24	220,94
7		8,50%	22609,83	12609,83	226,10
8		8,75%	23136,23	13136,23	231,36
9		9,00%	23673,64	13673,64	236,74
10		9,25%	24222,25	14222,25	242,22
11		9,50%	24782,28	14782,28	247,82
12		9,75%	25353,93	15353,93	253,54
13		10,00%	25937,42	15937,42	259,37
14					

Sie können die Anordnung der Formeln und Testwerte auch umkehren; die Testwerte also in der obersten Zeile und die Formeln in der linken Spalte des Mehrfachoperationsbereichs anordnen. Den durch die Testwerte zu ersetzenden Bezug geben Sie dann entsprechend unter *Werte aus Zeile* an.

17.3 Mehrfachoperationsbereiche editieren

Der Bereich, in den Excel die Resultate der Formeln einträgt, ist ein so genannter Mehrfachoperationsbereich (auch Matrix genannt).

Er bildet für Excel eine Einheit und es ist nicht möglich, ein einzelnes Feld des Bereichs zu editieren, es beispielsweise zu löschen oder einen anderen Wert einzugeben. Mehrfachoperationsbereiche können Sie nur am Stück editieren, also alle von Excel durch die Mehrfachoperation mit Inhalten gefüllten Zellen *gleichzeitig* löschen, kopieren oder verschieben.

Vor solchen Editierungen müssen Sie daher entweder den Mehrfachoperationsbereich selektieren oder aber einen größeren Bereich, in dem er *vollständig* enthalten ist.

Sie sehen an der Bearbeitungszeile, dass alle Resultatfelder eine Einheit bilden. Egal in welchem Resultatfeld Sie sich befinden, immer wird als Inhalt etwa folgende Formel angezeigt:

```
{=MEHRFACHOPERATION(;C3)}
```

Matrixformel

Die geschweiften Klammern zeigen an, dass dieser Bereich eine zusammengehörende Einheit darstellt, eine Matrix, die nur als Einheit manipuliert werden kann.

*Wissen Sie nicht mehr genau, welche Zellen der Mehrfachoperationsbereich umfasst, so selektieren Sie eine der zugehörigen Zellen und drücken **[Strg]+[⇧]+[↵]** (entspricht der Option **Aktueller Bereich des Befehls BEARBEITEN|GEHE ZU...**, die nach Klicken auf die Schaltfläche **Inhalte...** verfügbar ist): Excel markiert daraufhin den Mehrfachoperationsbereich, zu dem die aktive Zelle gehört.*

**:-)
TIPP**

17.4 Mit zwei Eingabefeldern

Wenn sich Ihre Anleihen außer im Zinssatz auch in der Laufzeit unterscheiden, ist die in Zelle C4 der folgenden Abbildung enthaltene Formel

```
=ZW(C3;C2;0;-C1)
```

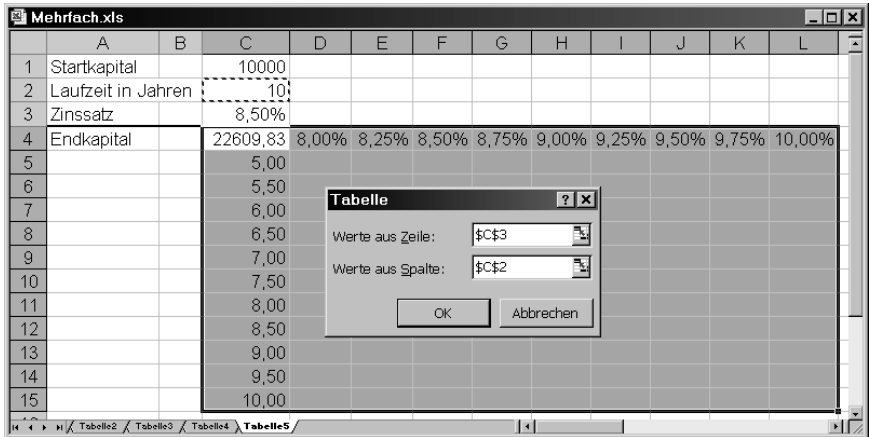
von zwei Zellen abhängig, von C3 (Zinssatz) und C2 (Laufzeit).

Excel ist in der Lage, Mehrfachoperationen mit bis zu zwei Eingabefeldern durchzuführen, deren Werte variieren. Entsprechend benötigen Sie nun zwei Gruppen von Testwerten: die verschiedenen Zinssätze, die für den Bezug C3 in die Formel einzusetzen sind, und die unterschiedlichen Laufzeiten, die den Bezug C2 ersetzen sollen. Zur Anordnung der beiden Gruppen:

- ➔ Die zu testende Formel befindet sich in der linken oberen Ecke eines rechteckigen Bereichs.
- ➔ Die eine Gruppe der Testwerte befindet sich rechts neben dieser Formel, also in der obersten Zeile des Bereichs.
- ➔ Die zweite Gruppe an Testwerten befindet sich darunter, also in der linken Spalte des Bereichs.

Der in der folgenden Abbildung selektierte Bereich C4:L15 entspricht diesen Anforderungen (Abbildung 17.7).

Abbildung 17.7:
Vorbereitung einer
Mehrfachoperation
mit zwei Eingabe-
feldern



Die Formel befindet sich in C4, in der linken oberen Ecke. Die eine Gruppe an Testwerten, die Zinssätze von 8% bis 10%, befinden sich rechts daneben. Die zweite Testgruppe, die verschiedenen Laufzeiten von fünf bis zehn Jahren in Halbjahresschritten, befinden sich unterhalb der Formel.

Die beiden Testgruppen bilden somit einen rechteckigen Bereich aus N (Anzahl der Zins-Testwerte) mal M (Anzahl der Laufzeit-Testwerte) Feldern. Jede Zelle dieses Bereichs wird nach der Mehrfachoperation genau einen Wert enthalten, der sich aus der Formel bei einer der zu testenden Laufzeiten und einem der zu testenden Zinssätze ergibt.

Im Dialogfeld sind nun *beide* Eingabefelder auszufüllen, da zwei Bezüge in der zu testenden Formel durch Testwerte ersetzt werden sollen:

- ➔ Der Bezug C3 (Zinssatz) soll Schritt für Schritt durch die zeilenweise angeordneten Laufzeiten ersetzt werden und muss daher unter *Werte aus Zeile* angegeben werden.
- ➔ Der Bezug C2 (Laufzeit in Jahren) wird durch die spaltenweise angeordneten Zinssätze ersetzt und daher unter *Werte aus Spalte* angegeben.

Mit diesen beiden Angaben (siehe vorhergehende Abbildung) erhalten Sie folgendes Ergebnis (siehe Bild 17.8).

Beispielsweise enthält I15 die Zahl 24222. Da Spalte I den Testwert 9,25% und Zeile 15 den Testwert 10 Jahre enthält, bedeutet das, dass eine Anleihe mit einem Zins von 9,25% und einer Laufzeit von zehn Jahren zu einem Endkapital von 24222 DM führt.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Startkapital		10000									
2	Laufzeit in Jahren		10									
3	Zinssatz		8,50%									
4	Endkapital		22609,83	8,00%	8,25%	8,50%	8,75%	9,00%	9,25%	9,50%	9,75%	10,00%
5			5,00	14693	14864	15037	15211	15386	15563	15742	15923	16105
6			5,50	15270	15465	15663	15862	16064	16267	16473	16681	16891
7			6,00	15869	16090	16315	16542	16771	17003	17238	17475	17716
8			6,50	16491	16741	16994	17250	17509	17772	18038	18308	18580
9			7,00	17138	17418	17701	17989	18280	18576	18876	19179	19487
10			7,50	17811	18122	18438	18759	19085	19416	19752	20092	20438
11			8,00	18509	18855	19206	19563	19926	20294	20669	21049	21436
12			8,50	19235	19617	20006	20401	20803	21212	21628	22052	22482
13			9,00	19990	20410	20839	21275	21719	22171	22632	23102	23579
14			9,50	20774	21236	21706	22186	22675	23174	23683	24202	24730
15			10,00	21589	22094	22610	23136	23674	24222	24782	25354	25937

Abbildung 17.8:

Mehrfach.xls,
»Tabelle5«

18 Matrixformeln

Die im letzten Kapitel erläuterten Mehrfachoperationen beruhen auf so genannten Matrixformeln, die ich nun erläutern werde.

Matrixformeln liefern als Funktionsergebnis keinen einfachen Wert, sondern eine Wertematrix, das heißt mehrere Werte gleichzeitig. Ich zeige Ihnen, wie Sie sich mithilfe dieser Eigenschaft häufig die wiederholte Eingabe bzw. das Kopieren mehrerer identischer Formeln ersparen können.

Allgemein ausgedrückt: Matrixformeln sind immer dann praktisch, wenn Sie eine Formel mehrfach nach unten, oben, rechts oder links kopieren. Sie können alternativ dazu eine einzige Matrixformel verwenden und sich dadurch teilweise eine Menge Arbeit ersparen!

18.1 Eingabe und Anwendung

Wie gesagt, eine einzige dieser Formeln ersetzt mehrere normale Formeln und spart zudem Speicherplatz. Dafür ist diese Formelart jedoch nicht ganz einfach zu verstehen, ebenso wenig wie das spätere Editieren einer Matrixformel.

Matrixformeln beruhen auf einem Teilgebiet der linearen Algebra, in dem es um den Umgang mit Matrizen geht. Abgebildet auf einer Tabelle ist eine Matrix ein rechteckiger Bereich, der sich über eine bestimmte Anzahl an Spalten und Zeilen erstreckt. Besteht der Bereich zum Beispiel aus sieben in einer Spalte untereinander angeordneten Zellen, spricht man von einer »7x1-Matrix«.

Allgemein: in einer »NxM-Matrix« kennzeichnet N die Anzahl der Zeilen und M die Anzahl der Spalten.

Mit Matrizen können verschiedene Operationen ausgeführt werden, bei denen die Matrix mit einer einzelnen Zahl oder gar mit einer anderen Matrix verknüpft wird. Beispielsweise können zwei Matrizen addiert werden.

In jedem Fall ist das Ergebnis erneut eine Matrix. Vor allem aber wirkt sich die betreffende Operation niemals nur auf einen Teil der Matrix aus: Jede einzelne Zelle ist davon betroffen!

Das ist auch der Grund, warum eine Matrixformel viele einfache Formeln ersetzt: Statt jeweils einer Formel pro Zelle des Bereichs wird bei Behandlung des Bereichs als Matrix nur noch eine einzige Formel benötigt, die sich auf die gesamte Matrix bezieht, eben eine Matrixformel.

Praktisch immer, wenn Sie eine Formel mehrfach nach unten, oben, rechts oder links kopieren, können Sie stattdessen eine einzige Matrixformel verwenden. Zum Beispiel, wenn sich in Spalte A sechs Nettobeträge untereinander befinden und Sie 30% eines jeden dieser Werte ermitteln wollen (siehe Bild 18.1).

Befinden sich die Beträge in A2:B7, würden Sie normalerweise in B2 die Formel

=A2*30%

verwenden, in B3 die Formel

=A3*30%

und so weiter.

**Matrix und
Konstante**

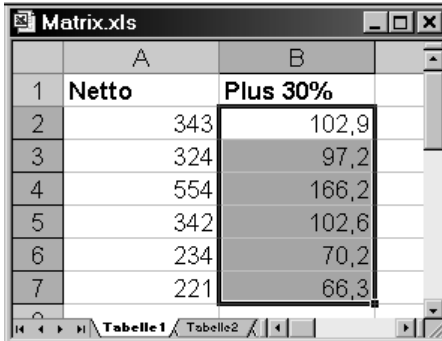
Eine einzige Matrixformel ersetzt diese sechs normalen Formeln: Selektieren Sie B2:B7 und geben Sie ein:

=A2:A7*30%

Drücken Sie bitte nicht , sondern  +  + .

Abbildung 18.1:
Matrix.xls,
»Tabelle1«

{=A2:A7*30%}



	A	B
1	Netto	Plus 30%
2	343	102,9
3	324	97,2
4	554	166,2
5	342	102,6
6	234	70,2
7	221	66,3

In der Bearbeitungsleiste (Abbildung 18.1) zeigt Excel nun folgende Formel an:

{=A2:A7*30%}

Egal welche Zelle des Bereichs B2:B7 Sie selektieren: Immer wird diese von geschweiften Klammern umschlossene Matrixformel angezeigt. Ihre Eigenschaften sind:

- ➔ Wo Sie in einer einfachen Formel eine einzelne Zelle angeben, geben Sie in einer Matrixformel einen Bereich an: Die Matrixformel liefert genauso viele Ergebnisse, wie der angegebene Bereich Zellen enthält.
- ➔ Das ist so, weil Excel die gewünschte Verknüpfung der Matrix mit der Konstanten *mehrfach* ausführt: zunächst mit der ersten Zelle des Bereichs, dann mit der zweiten Zelle und so weiter.

Eine Matrixformel kann mehrere Werte besitzen, vielwertig sein. Dann liefert sie als Ergebnis statt einer einzelnen Zahl einen rechteckigen Wertebereich, eine Wertematrix (siehe Kapitel 17, »Mehrfachoperationen«).



Beispielsweise verknüpft die Matrixformel

`{=A2:A7*30%}`

die Konstante 30% mit der Matrix A2:A7, das heißt mit jeder einzelnen Zelle dieses Bereichs. Die Formel besitzt daher sechs Ergebnisse.

Das Resultat, die Wertematrix, ist ebenfalls ein Bereich, der sich über eine Spalte und sechs Zeilen erstreckt. Jede Zelle dieses Bereichs enthält genau ein Resultat: In B2 wird das Ergebnis der Operation $A2 * 30\%$ gespeichert, in B3 das Resultat der zweiten Operation $A3 * 30\%$ etc.

Statt mit einer Konstanten wie 30% können Sie eine Matrix auch mit einer weiteren Matrix verknüpfen.

**Matrizen
miteinander
verknüpfen**

Angenommen, Sie benötigen in Spalte C die Summen der Spalten A und B. Beide Bereiche können als je eine 6×1 -Matrix interpretiert und diese beiden Matrizen miteinander addiert werden: Selektieren Sie den Bereich C2:C7, in dem das Ergebnis dieser Verknüpfung, die resultierende Wertematrix, gespeichert werden soll, und geben Sie folgende Matrixformel ein (Eingabe mit `[Strg] + [⇧] + [↵]` beenden!):

`=A2:A7+B2:B7`

Werden zwei Matrizen miteinander verknüpft, verknüpft Excel jeweils die Zellen, die sich an den gleichen relativen Positionen in den beiden Bereichen befinden, ausgehend von der jeweiligen oberen linken Bereichsecke.

Somit werden zunächst A2 und B2 addiert und das Ergebnis in C2 gespeichert, der ersten Zelle der Ergebnismatrix. Danach werden A3 und B3 addiert und das Ergebnis in C3 gespeichert und so weiter (Abbildung 18.2).

Abbildung 18.2:
Matrix.xls,
»Tabelle2«

{=A2:A7+B2:B7}

	A	B	C
1	Netto	Plus 30%	Gesamt
2	343	102,9	445,9
3	324	97,2	421,2
4	554	166,2	720,2
5	342	102,6	444,6
6	234	70,2	304,2
7	221	66,3	287,3

Werden wie hier zwei 6x1-Matrizen miteinander verknüpft, ist das Ergebnis ebenfalls eine 6x1-Matrix. Matrizen können jedoch ebenso gut mehrere Zeilen *und gleichzeitig* mehrere Spalten umfassen.

Nehmen wir an, Ihre Tabelle enthält in drei Spalten nebeneinander jeweils sechs Nettobeträge, zum Beispiel im Bereich A2:C7. Diese drei Spalten bilden eine 6x3-Matrix.

Nehmen wir weiter an, Sie benötigen von jedem dieser Beträge 30%. Diese Werte werden ebenfalls in drei Spalten mit je sechs Feldern gespeichert, also ebenfalls in einer 6x3-Matrix, zum Beispiel im Bereich E2:G7.

Selektieren Sie den Bereich, der die Ergebnismatrix enthalten soll, also E2:G7, und geben Sie für diesen Bereich folgende Formel ein:

=A2:C7*30%

Beenden Sie die Eingabe wieder mit **[Strg]+[↵]+[↵]**, da es sich ja wieder um eine Matrixformel handeln soll, die Excel anschließend in der Bearbeitungsleiste als {=A2:C7*30%} darstellt (Abbildung 18.3).

Abbildung 18.3:
Matrix.xls,
»Tabelle3«

{=A2:C7*30%}

	A	B	C	D	E	F	G	H	I	J	K
1	Netto1	Netto2	Netto3		Plus 30%	Plus 30%	Plus 30%		Brutto1	Brutto2	Brutto3
2	343	342	373		102,9	102,6	111,9		445,9	444,6	484,9
3	324	834	954		97,2	250,2	286,2		421,2	1084,2	1240,2
4	554	826	947		166,2	247,8	284,1		720,2	1073,8	1231,1
5	342	284	293		102,6	85,2	87,9		444,6	369,2	380,9
6	234	957	957		70,2	287,1	287,1		304,2	1244,1	1244,1
7	221	127	273		66,3	38,1	81,9		287,3	165,1	354,9

Die Matrixformel besitzt 18 Ergebnisse. Die Operation »*30%« wird der Reihe nach mit allen Zellen des Bereichs A2:C7 durchgeführt, also 18-mal. Somit wurden 18 einfache Formeln durch eine einzige Matrixformel ersetzt.

Wie der rechte Teil der Abbildung zeigt, werden zusätzlich die beiden 6x3-Matrizen der Bereiche A2:C7 und E2:G7 addiert und die resultierende 6x3-Ergebnismatrix im Bereich I2:K7 gespeichert. Dazu wurde dieser Bereich von mir selektiert und folgende Matrixformel eingegeben:

`{=A2:C7+E2:G7}`

Sie addiert die erste Zelle A2 des ersten Matrixbereichs A2:C7 und die erste Zelle E2 des zweiten Matrixbereichs E2:G7 und speichert die resultierende Summe in I2.

Danach addiert diese Formel die jeweils zweite Zelle der beiden Matrizen, A3 und E3, und speichert den resultierenden Bruttobetrag in der zweiten Zelle I3 der Ergebnismatrix etc.

Matrixformeln müssen nicht unbedingt mehrere Ergebnisse besitzen. Eine Matrixformel kann auch für eine 1x1-Ergebnismatrix gültig sein, also für eine einzelne Zelle.

Nehmen wir an, Sie wollen die drei Bruttowerte in Zeile 2 miteinander multiplizieren, also I2, J2 und K2.

Anschließend wollen Sie die entsprechenden drei Bruttowerte in Zeile 3 miteinander multiplizieren, dann die in Zeile 4 und zuletzt die drei Werte in I7, J7 und K7.

Alle sechs Ergebnisse sollen addiert und die Summe in I10 angezeigt werden.

Ohne Matrixformeln benötigen Sie in I10 folgenden Bandwurm:

`=I2*J2*K2+I3*J3*K3+I4*J4*K4+I5*J5*K5+I6*J6*K6+I7*J7*K7`

Den können Sie jedoch durch eine einzige Matrixformel ersetzen (Abbildung 18.4):

`{=SUMME(I2:I7*J2:J7*K2:K7)}`

Vergessen Sie zunächst die *SUMME*-Funktion und sehen Sie nur eine Matrixformel, die drei 6x1-Matrizen miteinander multipliziert: die Matrix I2:I7, die Matrix J2:J7 und die Matrix K2:K7.

Das Ergebnis ist wieder eine 6x1-Matrix: Zunächst werden die ersten Zellen der drei Matrizen miteinander multipliziert, I2, J2 und K2. Das Resultat wird in der ersten Zelle der Ergebnismatrix gespeichert. Diese Operation wird anschließend auch auf die anderen fünf Zeilen der Matrizen angewandt.

*Mehrdimen-
sionale
Matrizen*

Abbildung 18.4:
Matrix.xls,
»Tabelle4«

I10 {=SUMME(I2:I7*J2:J7*K2:K7)}

	A	B	C	D	E	F	G	H	I	J	K
1	Netto1	Netto2	Netto3		Plus 30%	Plus 30%	Plus 30%		Brutto1	Brutto2	Brutto3
2	343	342	373		102,9	102,6	111,9		445,9	444,6	484,9
3	324	834	954		97,2	250,2	286,2		421,2	1084,2	1240,2
4	554	826	947		166,2	247,8	284,1		720,2	1073,8	1231,1
5	342	284	293		102,6	85,2	87,9		444,6	369,2	380,9
6	234	957	957		70,2	287,1	287,1		304,2	1244,1	1244,1
7	221	127	273		66,3	38,1	81,9		287,3	165,1	354,9
8											
9											
10	Summe der zeilenweise miteinander multiplizierten Bruttowerte								2164751667		
11											

Wie gesagt, das Resultat ist eine 6x1-Ergebnismatrix. Dieser Matrixbereich wird zwar nirgendwo im Arbeitsblatt gespeichert, dennoch ist er vorübergehend virtuell vorhanden und auf ihn wird nun die *SUMME*-Funktion losgelassen. Sie addiert wie üblich alle Werte des angegebenen Bereichs, hier also die sechs Werte der Ergebnismatrix. Nur dieses endgültige Resultat wird im Arbeitsblatt gespeichert, in I10.

18.2 Editieren

Matrixbereiche können nur als Einheit editiert werden: Sie selektieren irgendeine Zelle des Bereichs, ändern die Matrixformel und schließen Ihre Änderungen statt mit  mit  +  +  ab.

Diese Tastenkombination erkennt Excel als Versuch, die für den gesamten Bereich gültige Matrixformel zu editieren, statt eine Formel in die aktuelle Zelle einzugeben, was bei Zellen eines Matrixbereichs nun mal nicht möglich ist.



Wissen Sie nicht mehr genau, über welchen Bereich sich eine Matrixformel erstreckt, so selektieren Sie einfach irgendeine Zelle des Bereichs und drücken  +  +  (Option Aktueller Bereich des Befehls BEARBEITEN|GEHE ZU..., Schaltfläche Inhalte...): Excel selektiert daraufhin den gesamten Matrixbereich, den Sie nun beispielsweise mit  löschen können.

Die Inhalte von Matrixbereichen können Sie in konstante Werte umwandeln, sodass die Matrixformel gelöscht und dafür in jeder Zelle der aus ihr resultierende Wert selbst gespeichert wird: Sie selektieren den Matrixbereich, wählen BEARBEITEN|KOPIEREN und danach – der Matrixbereich muss immer noch selektiert sein – BEARBEITEN|INHALTE EINFÜGEN... mit der Option *Werte*: In den Matrixbereich werden nur die momentan darin vorhandenen Werte eingefügt, nicht jedoch die Matrixformel selbst!

Da nun keine Matrixformel mehr existiert, gibt es auch keinen Matrixbereich mehr, sondern nur noch eine Ansammlung einzelner Zellen mit darin enthaltenen Zahlen.

18.3 Matrixkonstanten und Matrixexpansion

Außer Matrixbereichen gibt es auch Matrixkonstanten, also Konstanten, die *mehrere* Werte umfassen: Dazu listen Sie alle Werte durch Semikola voneinander getrennt auf, umgeben von einer geschweiften Klammer.

Beispielsweise ist {1;2;3;4} eine 4x1-Matrixkonstante, also eine Matrix, die sich über vier Zeilen und eine Spalte erstreckt:

```
1
2
3
4
```

Dagegen handelt es sich mit einem Punkt als Trennzeichen ({1.2.3.4}) gerade umgekehrt um eine 1x4-Matrix mit einer Zeile und vier Spalten, entsprechend einem Bereich wie A1:D1:

```
1 2 3 4
```

Bei mehrdimensionalen Matrizen leitet das Semikolon praktisch den Beginn einer neuen Matrixzeile ein. Daher entspricht die Matrixkonstante

```
{1.2.3;4.5.6}
```

folgender Werteanordnung:

```
1 2 3
4 5 6
```

Das entspricht also einer 2x3-Matrix. Ein weiteres Beispiel ist dieses (Abbildung 18.5):

```
{=A1:A3*{4;5;6}}
```

	A	B
1	1	4
2	2	10
3	3	18
4		

Abbildung 18.5:
Matrix.xls,
»Tabelle5«

Nach Selektion des Bereichs B1:B3 gab ich (mit **Strg** + **↕** + **↵**) folgende Matrixformel ein:

$\{=A1:A3*{4;5;6}\}$

Der erste Wert A1 im Matrixbereich A1:A3 wird dadurch mit dem ersten Wert der Matrixkonstanten multipliziert, mit 4. Danach wird der zweite Wert A2 mit dem zweiten Wert 5 der Matrixkonstanten multipliziert und zuletzt der dritte Wert A3 mit dem dritten Wert 6 der Matrixkonstanten.



Bei Verwendung des zeilenorientierten Bereichs A1:C3 müsste die Matrixkonstante wie zuvor erläutert übrigens {4.5.6} lauten, also ebenfalls zeilenorientiert sein. Nur bei identischer Form von Matrixkonstanten und Matrixbereich ist gewährleistet, dass jeder Zahl des Bereichs eine entsprechende Konstante zugeordnet ist!

Sie können mit mehreren Matrixkonstanten gleichzeitig operieren:

$=SUMME(\{1;2;3\}+\{4;5;6\})$

Die beiden Matrizen werden zeilenweise addiert: Zuerst 1 und 4, dann 2 und 5 und zuletzt 3 und 6. Das Ergebnis ist wieder eine 3x1-Matrixkonstante: {5;7;9}.

Die *SUMME*-Funktion behandelt diese Matrixkonstante als Bereich und addiert die darin enthaltenen Zahlen zum Resultat 21.

Ein letztes Beispiel (Abbildung 18.6):

Abbildung 18.6:
Matrix.xls,
»Tabelle6«

$\{=A2:B7+\{1.2;3.4;5.6;7.8;9.10;11.12\}\}$

	A	B	C	D	E
1	Netto1	Netto2			
2	52	12		53	14
3	85	16		88	20
4	94	72		99	78
5	73	83		80	91
6	64	42		73	52
7	87	25		98	37

Zu jeder Zahl im Bereich A2:B7 soll eine Konstante addiert werden: zu A2 der Wert 1, zu B2 der Wert 2, zu A3 eine 3 und so weiter. Also benutzen wir eine Matrixkonstante gleicher Form und Ausdehnung:

$\{=A2:B7+\{1.2;3.4;5.6;7.8;9.10;11.12\}\}$

Durch Punkte getrennte Werte befinden sich in der gleichen Zeile der Matrixkonstanten, jedem Semikolon folgen die Werte der darunter liegenden Zeile.

Wenn Sie eine Matrixkonstante als Operand verwenden und die Größe dieser Matrix nicht zu der damit zu verknüpfenden Matrix passt, expandiert Excel die Matrixkonstante *selbstständig*. Ein Beispiel:

Matrixexpansion

```
=SUMME(C2:E4+{4.8,3})
```

Der Matrixbereich C2:E4 und die Matrixkonstante {4.8,3} werden addiert. Die Matrixkonstante enthält jedoch nur zwei Werte, ist also kleiner als der Matrixbereich. Excel expandiert die Matrixkonstante *selbstständig* und verhält sich so, als enthielte sie außer der 4 und der 8,3 zusätzlich eine 0, also so, als hätten Sie tatsächlich folgende Formel eingegeben:

```
=SUMME(C2:E4+{4.8,3.0})
```

Matrixexpansion bedeutet somit, dass Excel bei unvollständigen Matrizen *selbstständig* Nullen am Ende ergänzt.

19 Datenexport und Datenimport

Sie können auf unterschiedlichste Art und Weise die Daten einer Excel-Tabelle exportieren, um sie beispielsweise mit einer Textverarbeitung oder Dateiverwaltung weiterzuverarbeiten, oder umgekehrt Daten eines Fremdprogramms in eine Excel-Tabelle importieren.

Ich zeige Ihnen, wie Sie mithilfe des Befehls DATEI|SPEICHERN UNTER... Daten in unterschiedlichsten Formaten speichern können, beispielsweise im Textformat oder als dBase-Datei – vor allem aber in den verschiedenen Formaten der vorhergehenden Excel-Versionen, beispielsweise als Excel 3.0-Arbeitsblatt, sodass auch Kollegen mit älteren Excel-Versionen Ihre Daten weiterbearbeiten können.

Alternativ dazu können Sie eine Excel-Tabelle in einem Format exportieren, das eine Textverarbeitung wie Word lesen kann, sodass Sie die Tabelle ohne Abtippen eines Ausdrucks direkt in einen Text einbinden können.

Umgekehrt können Sie eine Tabelle aus einer Textverarbeitung wie Win-Word in eine Excel-Tabelle importieren oder Datensätze importieren, die zum Beispiel mit dBase erzeugt wurden. Excel kann problemlos Dateien all dieser Formate öffnen.

Erkennt Excel dabei ein bestimmtes Textformat nicht, meldet sich automatisch ein Assistent, der Sie durch den Importprozess führt und die zum erfolgreichen Import benötigten Angaben erfragt.

So mancher wird meinen, dass ich im vorliegenden Kapitel Textdateien viel zu intensiv behandle, und wird es auch überflüssig finden, dass ich ausführlich auf »Altformate« wie dBase eingehe.

Beides hat einen Grund: Es gibt Tausende Spezialprogramme, mit denen der eine oder andere eventuell mit Excel Daten austauschen will. »Der eine oder andere« bedeutet, dass ich unmöglich auf jedes dieser Spezialprogramme eingehen kann.

Genau hier kommt das Textformat ins Spiel, da *nahezu jedes Programm* die allgemeinste Form des Exports/Imports von Daten beherrscht, nämlich den Export/Import in irgendeiner der verschiedenen Textformat-Varianten –



oder eben in einem Altformat wie dBase, das nun mal Standards gesetzt hat! Diese Formate bieten daher nahezu mit Sicherheit zumindest Kompromisslösungen, auf welches Datenaustauschproblem auch immer Sie in der Zukunft treffen mögen.



Übrigens stelle ich im Rahmen der Erläuterungen zu den Textformaten ein kleines Programmchen vor, das sich auf der Begleit-CD zum Buch befindet und das beim Textdatei-Import manchmal auftretende Problem variabler Leerzeichenfolgen löst, indem es beliebig lange Leerzeichen-Sequenzen durch je ein Tabulatorzeichen ersetzt. Diese Vorbehandlung ermöglicht auch den Import äußerst problematischer Textdateien.

Nach Behandlung der Textformate führe ich Sie in die Besonderheiten des Datenaustauschs mit Multiplan, Lotus, Symphony und dBase ein.

Danach erläutere ich die Möglichkeiten von Excel zum Datenaustausch mit modernen Windows-Programmen. Das beginnt beim Kopieren von Daten in die Zwischenablage und Einfügen in das Fremdprogramm.

Stattdessen können Sie jedoch auch mit dem dynamischen Datenaustausch (DDE = Dynamic Data Exchange) dauerhafte Verknüpfungen zwischen Excel und einem Fremdprogramm erzeugen, die wahlweise automatisch oder auf Befehl aktualisiert werden, sodass beispielsweise die in ein Word-Dokument eingefügten Excel-Daten immer auf dem neuesten Stand sind.

Bei modernen Windows-Programmen können Sie Daten auch einbetten, beispielsweise eine Excel-Tabelle in ein Word-Dokument. Dann genügt ein Doppelklick auf das eingebettete Objekt, um den Objektlieferanten aufzurufen, beispielsweise Excel, und die Excel-Tabelle zu bearbeiten, ohne Word verlassen und Excel aufrufen zu müssen.

19.1 Datenbankprogramme und ODBC



Eine spezielle Form des Datenimports erläutere ich hier nicht, da das bereits geschah: In Kapitel 14, »Microsoft Query – Zugriff auf externe Datenbanken«, erläutere ich den Umgang mit Microsoft Query.

Mithilfe dieses Programms können Sie auf dBase-, FoxPro-, Access- und weitere Datenbanken zugreifen oder gar Datenbanken in einem dieser Formate erstellen.



Um Daten aus einer solchen Datenbank in eine Excel-Tabelle zu importieren, können Sie sie entweder mit dem betreffenden Programm in einem der nachfolgend erläuterten Dateiformate speichern oder aber die Daten wesentlich komfortabler und flexibler mit den in Kapitel 14.2, »Externe Daten in Excel importieren«, erläuterten Techniken importieren.

Sie können auch mit den in Kapitel 16.8, »Zugriff auf externe Datenquellen«, erläuterten Methoden daraus eine Pivot-Tabelle erstellen.



19.2 Dateiformate

Um Excel-Tabellen in Formaten zu speichern, die für andere Programme lesbar sind, benutzen Sie die *Dateityp*-Optionen des Befehls DATEI|SPEICHERN UNTER... (Abbildung 19.1).

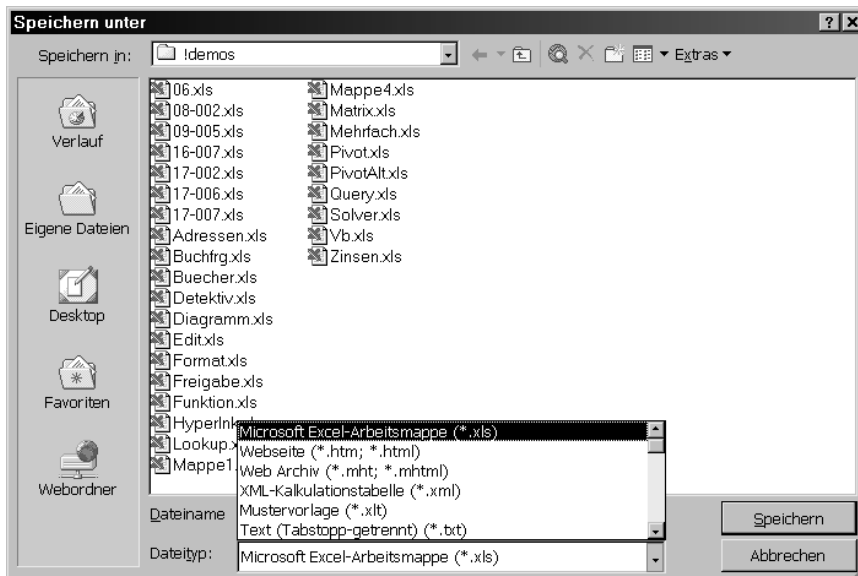


Abbildung 19.1:
Dateiformate

- ➔ Microsoft Excel-Arbeitsmappe: Excels eigenes Format, in dem nur Excel selbst ab jener Version, mit der Sie arbeiten oder höher die gespeicherten Dateien wieder lesen kann
- ➔ Die verschiedenen »Microsoft Excel x.x-Arbeitsmappe«-Formate: Speichern die Mappe im Format der betreffenden Excel-Version
- ➔ Die verschiedenen »Microsoft Excel x.x-Tabelle«-Formate: Speichern nur die aktive Tabelle im Format der betreffenden Excel-Version
- ➔ Formatierter Text (Leerzeichen getrennt): Textformat mit Leerzeichen als Trennzeichen zwischen den einzelnen Spalten einer Zeile
- ➔ Text (Tabstopp getrennt): Textformat mit Tabulatoren als Trennzeichen
- ➔ CSV (Trennzeichen getrennt): Textformat mit Kommata als Trennzeichen (*deutsche Version: Semikola!*)

- ➔ Microsoft Excel 4.0-Arbeitsmappe: Speichert die gesamte Mappe als Excel-4.0-Arbeitsbereichsdatei
- ➔ WQ1 (Quattro Pro/DOS): Speichert nur die aktive Tabelle im Quattro Pro für DOS-Format
- ➔ »Text (Macintosh)« und »Text (MS-DOS)«: Textformate für den Mac und DOS
- ➔ »CSV (Macintosh)« und »CSV (MS-DOS)«: Textformate mit Trennzeichen für den Mac und DOS
- ➔ DIF (Data Interchange Format): Speichert im Visicalc-»Data Interchange Format«
- ➔ SYLK (Symbolische Verbindung): Speichert Dateien im »Symbolic Link Format«, das früher von vielen DOS-Microsoft-Produkten wie Multiplan und Chart benutzt wurde
- ➔ Microsoft Excel-Add-In: Speichert die Mappe als Add-In
- ➔ Die verschiedenen »WK x (1-2-3)«-Formate: Speichern die Mappe (Tabellen und Diagramme) bzw. nur die aktive Tabelle im Format der betreffenden Lotus-Version
- ➔ Die verschiedenen »DBFx (dBASE x)«-Formate: Speichern den aktuellen Datenbankbereich im Format der betreffenden dBase-Version



Webseite, Webarchiv und XML-Kalkulationstabelle: Speichert eine Tabelle als Webseite, wobei entweder eine Hauptdatei und zusätzlich viele Hilfsdateien gespeichert werden (z.B. für jede Grafik in der Tabelle eine eigene Datei) oder aber nur eine einzige Datei, die die komplette Tabelle als Webseite enthält, dafür jedoch höhere Mindestanforderungen an den Webbrowser stellt, mit dem sie betrachtet werden soll (siehe Kapitel 27.2, »Als Webseite speichern«, und Kapitel 27.3, »Interaktive Webseiten«)



Mustervorlage: Speichert eine Tabelle als Mustervorlage (siehe Kapitel 8.8, »Arbeitsmappen als Vorlagen verwenden und automatisch öffnen«)



Beim Speichern in Formaten der Excel-Vorgängerversionen gehen alle Features Ihrer Mappen verloren, über die die betreffende Version noch nicht verfügte.

Besonders krass ist das, wenn die Mappe ein Dialog- oder ein Modulblatt enthält: In einem der 4.0-, 3.0- oder 2.1-Tabellenformate kann ein solches Blatt nicht gespeichert werden!

Als Excel 4.0-Arbeitsmappe gespeichert sind die betreffenden Blätter in der resultierenden .XLW-Datei ganz einfach nicht enthalten – allerdings könnten die vorhergehenden Excel-Versionen ja sowieso nichts mit einem der beiden Blatt-Typen anfangen.

Beim Speichern müssen Sie das gewünschte Dateiformat selbst wählen. Öffnen Sie jedoch eine in einem der verfügbaren Formate gespeicherte Datei, erkennt Excel das betreffende Format selbstständig.



Beachten Sie jedoch, dass Sie die Dateimaske »Alle Dateien(*.*)« verwenden müssen, um unabhängig von der oft formatspezifischen Erweiterung auch wirklich *alle* Dateien angezeigt zu bekommen, also auch Dateien wie TEST.TXT, TEST.ASC oder TEST.SLK.

Sie können auch Dateien im HTML-Format öffnen, das heißt Webseiten. Wie Sie Excel-Tabellen genau umgekehrt in diesem Format speichern, erläutere ich in Kapitel 25.4.2, »Webseiten«.



19.3 Im Textformat exportieren

Die unverbindlichste Form des Datenaustauschs ist die Verwendung des Textformats in seinen Varianten »Formatierter Text«, »durch Tabulatoren getrennter Text« und »CSV«.

Alle drei Formate werden praktisch von jedem Programm zumindest halbwegs verstanden. Dafür berücksichtigt keines irgendwelche Besonderheiten des Zielprogramms.

In allen Fällen werden ausschließlich Texte und Zahlen gespeichert, aber keine Formeln.

Jede Zeile der Tabelle wird in genau einer Zeile der erzeugten Datei gespeichert, die mit einem Zeilenvorschub abgeschlossen wird, einer Kombination der Sonderzeichen Carriage Return (ASCII-Code 13) und Linefeed (ASCII-Code 10).

Die drei Formate unterscheiden sich durch die Art und Weise, in der die in einer Zeile enthaltenen Werte voneinander getrennt sind. Gehen wir von folgender Tabelle aus (Abbildung 19.2).

Im Format »Formatierter Text« gespeichert bleiben die Originalspaltenbreiten auch in der Textdatei erhalten. Text wie der Name »Hohenfeld, Christine«, der – da nun die Proportionalschrift fehlt – nicht in die vorgegebenen Spalten hineinpasst, wird einfach abgeschnitten. Leerräume zwischen den einzelnen Spalten sind mit Leerzeichen gefüllt (Abbildung 19.3).

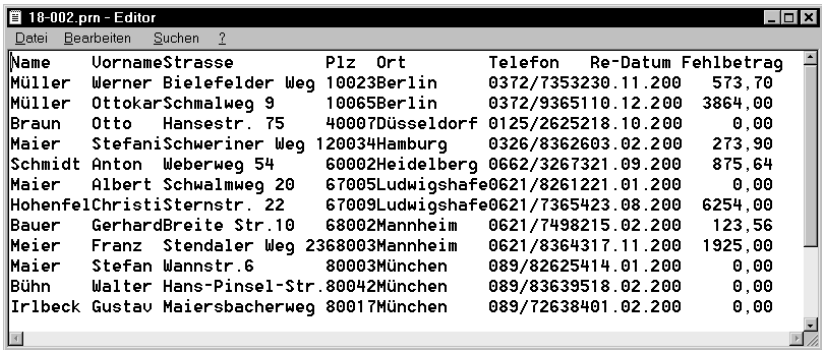
Typ »Formatierter Text«

Abbildung 19.2:
Demo-Tabelle



	A	B	C	D	E	F	G	H
	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	30.11.2009	573,70
3	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.2009	3864,00
4	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.10.2009	0,00
5	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.02.2009	273,90
6	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.09.2009	875,64
7	Maier	Albert	Schwalweg 20	67005	Ludwigshafen	0621/826125	21.01.2009	0,00
8	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.08.2009	6254,00
9	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.02.2009	123,56
10	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.11.2009	1925,00
11	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.2009	0,00
12	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.2009	0,00
13	Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/726384	01.02.2009	0,00

Abbildung 19.3:
Formatierter Text



Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/7353230	11.200	573,70
Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/9365110	10.12.200	3864,00
Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/2625218	10.200	0,00
Maier	Stefani	Schweriner Weg 12	20034	Hamburg	0326/8362603	02.200	273,90
Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/3267321	09.200	875,64
Maier	Albert	Schwalweg 20	67005	Ludwigshafe	0621/8261221	01.200	0,00
Hohenfeld	Christi	Sternstr. 22	67009	Ludwigshafe	0621/7365423	08.200	6254,00
Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/7498215	02.200	123,56
Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/8364317	11.200	1925,00
Maier	Stefan	Wannstr.6	80003	München	089/82625414	01.200	0,00
Bühn	Walter	Hans-Pinsel-Str.80042	München		089/83639518	02.200	0,00
Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/72638401	02.200	0,00

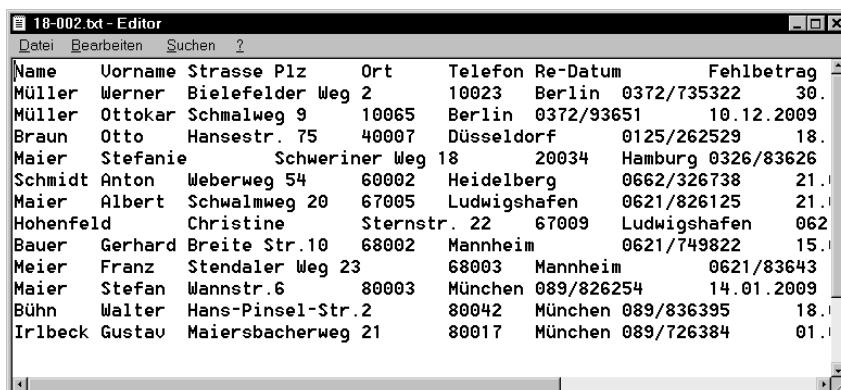
Um das Problem der abgeschnittenen Zeichenketten zu vermeiden, erhöhen Sie in der Excel-Tabelle die Spaltenbreiten und speichern die Tabelle anschließend erneut.

Typ »Text« Im normalen Textformat »Text« gespeichert wird der Inhalt jeder Zelle durch je ein Tabulatorzeichen vom Inhalt der Nachbarzellen getrennt. Nach dem Einlesen in eine Textverarbeitung ergibt sich in etwa die folgende Darstellung (Abbildung 19.4).

Zellinhalte werden auf keinen Fall abgeschnitten und bleiben vollständig erhalten.

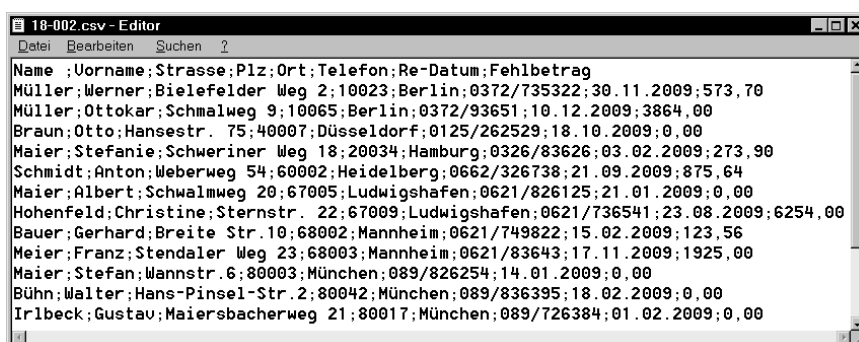
Dass die Tabelle leicht verrutscht ist, liegt an der ungeeigneten Tabulatoreinstellung im zur Anzeige verwendeten DOS-Programm. Durch Veränderung der Tabulatoreinstellungen in der verwendeten Textbearbeitung können Sie dieses Problem sehr einfach lösen und die Daten wieder spaltengerecht untereinander anordnen.

Typ »CSV« Im Format »CSV« erfolgt die Trennung der Zellinhalte durch je ein Semikolon (Abbildung 19.5).



Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
Müller	Werner	Bielefelder Weg 2		10023	Berlin	0372/735322	30.
Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.2009	
Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.	
Maier	Stefanie	Schweriner Weg 18		20034	Hamburg	0326/83626	
Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.	
Maier	Albert	Schwalweg 20	67005	Ludwigshafen	0621/826125	21.	
Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	062		
Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.	
Meier	Franz	Stendaler Weg 23		68003	Mannheim	0621/83643	
Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.2009	
Bühn	Walter	Hans-Pinsel-Str.2		80042	München	089/836395	18.
Irlbeck	Gustav	Maiersbacherweg 21		80017	München	089/726384	01.

Abbildung 19.4:
Tabulatorformat



Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	30.11.2009	573,70
Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.2009	3864,00
Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.10.2009	0,00
Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.02.2009	273,90
Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.09.2009	875,64
Maier	Albert	Schwalweg 20	67005	Ludwigshafen	0621/826125	21.01.2009	0,00
Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.08.2009	6254,00
Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.02.2009	123,56
Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.11.2009	1925,00
Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.2009	0,00
Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.2009	0,00
Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/726384	01.02.2009	0,00

Abbildung 19.5:
CSV-Format

Analog zum unformatierten Textformat bleiben die Zellinhalte unabhängig von der Einstellung der Spaltenbreiten vollständig erhalten.

Die Textformate besitzen einen Haken: DOS-Programme verwenden die ASCII-Zeichen-Kodierung, Windows dagegen die ANSI-Zeichen-Kodierung. Öffnen Sie die erzeugte Textdatei mit einem DOS-Programm (Norton Commander etc.), werden darin Umlaute und Sonderzeichen wie »ß« falsch interpretiert.

ASCII-Kodierung

19.4 Textdatei-Import – der Textkonvertierungs-Assistent

Importieren Sie im Textformat gespeicherte Dateien, indem Sie die Textdatei mit DATEI|ÖFFNEN... öffnen, müssen Sie Excel manchmal ein wenig helfen.

Am unproblematischsten ist das Einlesen einer im CSV-Format gespeicherten Datei: Erstellen Sie mit einer Textverarbeitung eine Datei, in der die in den einzelnen Zeilen enthaltenen Werte durch Semikola voneinander getrennt sind und speichern Sie diese Datei mit der Endung .CSV, so wird jeder Wert in Excel wie gewünscht in eine eigene Spalte übernommen.

Fehlinterpretationen

In allen anderen Fällen braucht Excel ein wenig Hilfe, um die in den einzelnen Dateizeilen enthaltenen Werte korrekt voneinander zu unterscheiden. Sonst erhalten Sie katastrophale Ergebnisse wie das folgende (Abbildung 19.6):

Abbildung 19.6:
Fehlende Spalten-
trennung

Excel kann die einzelnen Elemente einer Dateizeile nicht voneinander unterscheiden und liest in der Spalte B mehrere Felder in *eine einzige Zelle* ein!

Dieses Problem können Sie nachträglich mit dem Textkonvertierungs-Assistenten beheben, indem Sie *die erste Spalte dieses Bereichs* selektieren und DATEN|TEXT IN SPALTEN... wählen.

Diese nachträgliche Analyse von Spalte A erübrigt sich normalerweise, da sich dieser Assistent beim Einlesen einer zweifelhaften Textdatei automatisch meldet (Abbildung 19.7).



Ausnahme: Die Datei besitzt die Endung .CSV, ohne das dadurch definierte Format auch tatsächlich zu besitzen: Excel nimmt an, dass es sich um eine im CSV-Format gespeicherte Datei handelt und liest sie mit den dafür zuständigen Methoden – fehlerhaft – ein.

19.4.1 Trennzeichen

Der Textkonvertierungs-Assistent analysiert die einzulesende Datei und erkennt meist korrekt, ob sich zwischen den einzelnen Werten einer Zeile ein Trennzeichen (»«, Tabulator etc.) befindet oder aber eine unterschiedliche Anzahl von Leerzeichen, wie das beim zuvor besprochenen formatierten Textformat der Fall ist.

Entsprechend ist entweder die Option *Getrennt* oder *Feste Breite* aktiviert. Achten Sie unbedingt darauf, dass die zutreffende Option aktiviert ist!

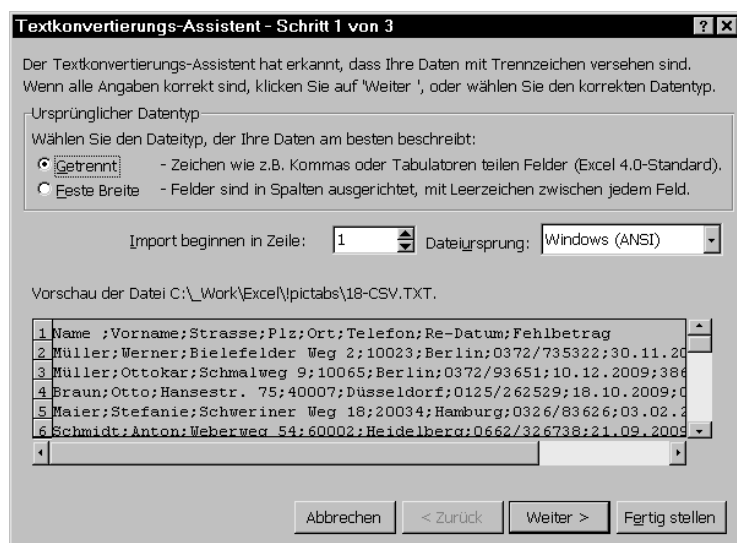


Abbildung 19.7:
Der Textkonvertierungs-Assistent

Mit *Import beginnen in Zeile* können Sie wählen, ab welcher Zeile der Import der Datei beginnen soll und sich dazu an der Dateivorschau orientieren, in der Sie die gesamte Textdatei durchblättern können.

Dateiursprung entscheidet darüber, ob Umlaute und andere Sonderzeichen korrekt interpretiert werden.

Umlaute

Bei DOS-Dateien muss unbedingt »MS-DOS (PC-8)« selektiert werden. Um eine Windows-Datei einzulesen, zum Beispiel eine zuvor unter Excel im CSV-Format gespeicherte Tabelle, müssen Sie »Windows (ANSI)« selektieren und zum Einlesen einer Macintosh-Datei »Macintosh«.

Der zweite Schritt hängt davon ab, ob die Textdatei mit Trennzeichen oder mit festen Spaltenbreiten eingelesen wird. Aktivierten Sie *Getrennt*, so erscheint folgendes Dialogfeld (siehe Abbildung 19.8).

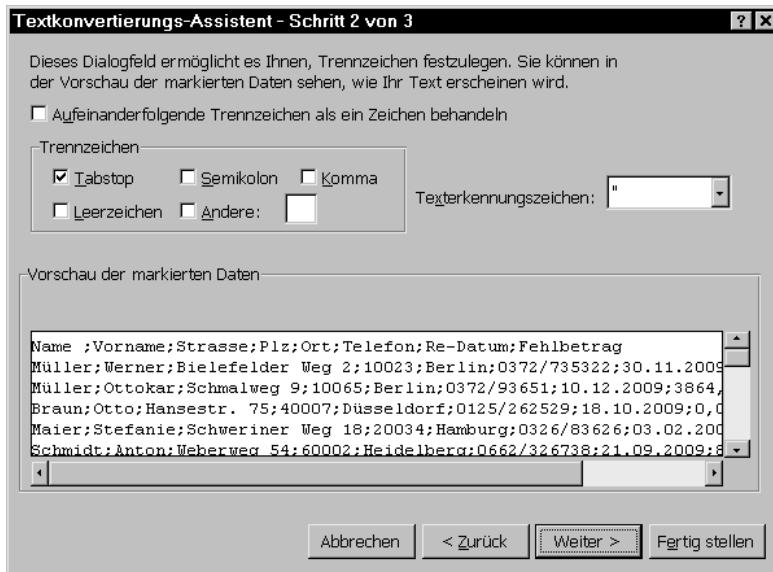
Trennzeichen

Unter *Trennzeichen* geben Sie das Trennzeichen an, das sich zwischen den einzelnen Werten befindet.

Wurde in der Datei ein anderes als die hier verfügbaren Trennzeichen verwendet, aktivieren Sie *Andere* und geben das betreffende Zeichen im zugehörigen Eingabefeld ein.

Befinden sich zwischen zwei Feldern auch nur ab und zu mehrere aufeinander folgende Trennzeichen, beispielsweise zwei Leerzeichen, wurde in Excel 4.0 das unvorhergesehene zweite Leerzeichen als eigenes Textfeld behandelt und nach dem Textimport in einer eigenen – leeren – Zelle dargestellt.

Abbildung 19.8:
Trennzeichen
angeben



Ab Excel 5.0 aktivieren Sie in diesem Fall einfach das Kontrollkästchen *Aufeinander folgende Trennzeichen als ein Zeichen behandeln*: Excel importiert nun auch solche Dateien mit variabler Trennzeichenanzahl korrekt, egal wie viele Trennzeichen sich jeweils zwischen zwei Feldern befinden.

In zu importierenden Dateien sind Texte manchmal in spezielle Zeichen wie einfache oder doppelte Anführungszeichen eingeschlossen. Ist das der Fall, wählen Sie unter *Texterkennungszeichen* das entsprechende Zeichen aus.

Vorschau

Selektieren Sie eine Option oder geben Sie ein individuelles Trennzeichen ein, so zeigt der Assistent sofort darauf im Vorschaufenster an, wie die Dateizeilen unter Verwendung dieses Zeichens als Spaltentrenner aufgetrennt würden.

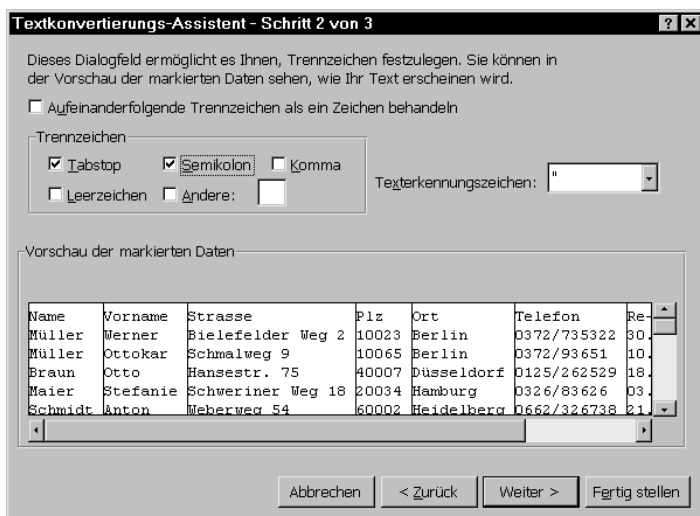
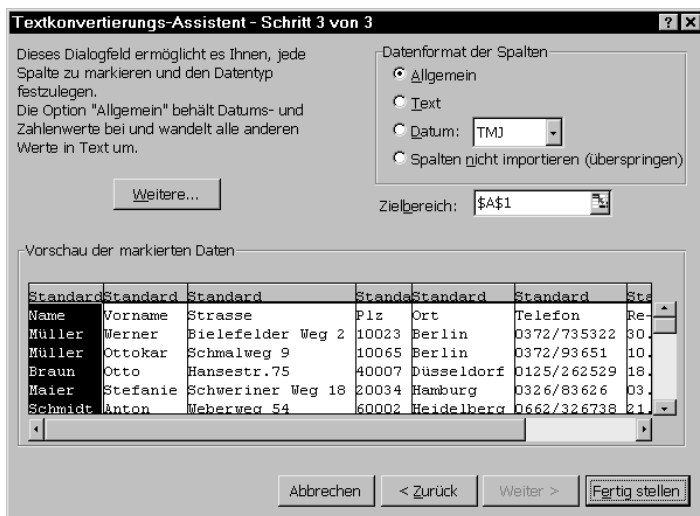
Angewandt auf die vorhergehende Abbildung wäre die Option *Semikolon* korrekt, da sich zwischen den einzelnen Werten jeder Dateizeile je ein Semikolon befindet (Abbildung 19.9).

Die vertikalen Striche deuten die Spalten an, über die die Werte einer Zeile verteilt werden.

Datentyp

Im dritten Schritt können Sie für jede Spalte getrennt den Typ der darin enthaltenen Werte bekannt geben (Abbildung 19.10).

Über jeder Spalte wird die aktuelle Einstellung angezeigt, zum Beispiel *Text* oder *Spalten nicht importieren (überspringen)*.

Abbildung 19.9:
TabellenvorschauAbbildung 19.10:
Datentyp

Normalerweise sollten Sie das vorgegebene Datenformat für alle Spalten beibehalten. Kommt es dadurch zu Fehlinterpretationen, selektieren Sie die betreffende Spalte durch Anklicken im Vorschaufeld und aktivieren die zutreffende Option, wobei Sie bei Datumsfeldern zusätzlich die Reihenfolge der drei Datuselemente »Tag, Monat, Jahr« angeben können.

Spalten nicht importieren (überspringen) bewirkt, dass die betreffende Spalte nicht in die Tabelle übernommen wird.

*Spalten
überspringen*

Fertig stellen importiert die Textdatei gemäß den aktuellen Einstellungen und legt dazu eine neue Arbeitsmappe an, die genau eine Tabelle enthält (Abbildung 19.11).

Abbildung 19.11:
Erfolgreicher Import



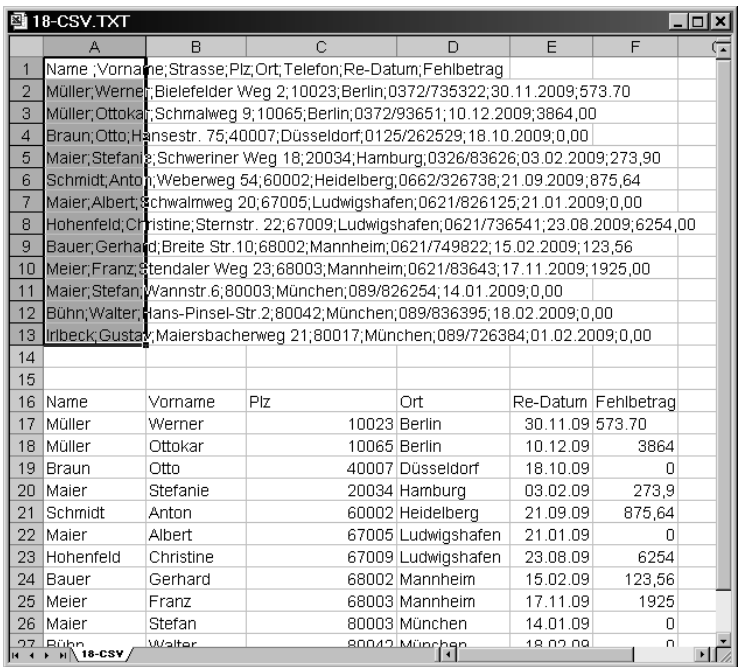
	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Müller	Werner	Bielefelder	10023	Berlin	0372/7353	30.11.09	573,70
3	Müller	Ottokar	Schmalweg	10065	Berlin	0372/9365	10.12.09	3864
4	Braun	Otto	Hansestr.	40007	Düsseldorf	0125/2625	18.10.09	0
5	Maier	Stefanie	Schwerine	20034	Hamburg	0326/8362	03.02.09	273,9
6	Schmidt	Anton	Weberweg	60002	Heidelberg	0662/3267	21.09.09	875,64
7	Maier	Albert	Schwalnw	67005	Ludwigsha	0621/8261	21.01.09	0
8	Hohenfeld	Christine	Sternstr. 2	67009	Ludwigsha	0621/7365	23.08.09	6254
9	Bauer	Gerhard	Breite Str.	68002	Mannheim	0621/7498	15.02.09	123,56
10	Meier	Franz	Stendaler	68003	Mannheim	0621/8364	17.11.09	1925
11	Maier	Stefan	Wannstr.6	80003	München	089/82625	14.01.09	0
12	Bühn	Walter	Hans-Pins	80042	München	089/83639	18.02.09	0
13	Irlbeck	Gustav	Maiersbac	80017	München	089/72638	01.02.09	0

Textkonver-
tierungs-Assistent
manuell aufrufen

Wird der Assistent nicht beim Datei-Import, sondern manuell mit DATEN|TEXT IN SPALTEN... aufgerufen (um den Inhalt einer zuvor selektierten Tabellenspalte über mehrere Spalten zu verteilen), befindet sich unter den *Datentyp*-Optionen (siehe dritter Schritt, Abbildung 19.10) zusätzlich ein Eingabefeld *Zielbereich*.

Geben Sie darin durch Zeigen einen Zellbezug an, so wird die Originalta-
belle nicht überschrieben, sondern stattdessen eine Kopie erstellt, mit der
angegebenen Zelle als linker oberer Bereichsecke (Abbildung 19.12).

Abbildung 19.12:
Textkonvertierungs-
Assistent manuell
aufrufen



	A	B	C	D	E	F
1	Name ;Vorname;Strasse;Plz;Ort;Telefon;Re-Datum;Fehlbetrag					
2	Müller;Werner;Bielefelder Weg 2;10023;Berlin;0372/735322;30.11.2009;573.70					
3	Müller;Ottokar;Schmalweg 9;10065;Berlin;0372/93651;10.12.2009;3864,00					
4	Braun;Otto;Hansestr. 75;40007;Düsseldorf;0125/262529;18.10.2009;0,00					
5	Maier;Stefani;Schweriner Weg 18;20034;Hamburg;0326/83626;03.02.2009;273,90					
6	Schmidt;Anton;Weberweg 54;60002;Heidelberg;0662/326738;21.09.2009;875,64					
7	Maier;Albert;Schwalnweg 20;67005;Ludwigshafen;0621/826125;21.01.2009;0,00					
8	Hohenfeld;Christine;Sternstr. 22;67009;Ludwigshafen;0621/736541;23.08.2009;6254,00					
9	Bauer;Gerhard;Breite Str.10;68002;Mannheim;0621/749822;15.02.2009;123,56					
10	Meier;Franz;Stendaler Weg 23;68003;Mannheim;0621/83643;17.11.2009;1925,00					
11	Maier;Stefan;Wannstr.6;80003;München;089/826254;14.01.2009;0,00					
12	Bühn;Walter;Hans-Pinsel-Str.2;80042;München;089/836395;18.02.2009;0,00					
13	Irlbeck;Gustav;Maiersbacherweg 21;80017;München;089/726384;01.02.2009;0,00					
14						
15						
16	Name	Vorname	Plz	Ort	Re-Datum	Fehlbetrag
17	Müller	Werner		10023	Berlin	30.11.09 573.70
18	Müller	Ottokar		10065	Berlin	10.12.09 3864
19	Braun	Otto		40007	Düsseldorf	18.10.09 0
20	Maier	Stefanie		20034	Hamburg	03.02.09 273,9
21	Schmidt	Anton		60002	Heidelberg	21.09.09 875,64
22	Maier	Albert		67005	Ludwigshafen	21.01.09 0
23	Hohenfeld	Christine		67009	Ludwigshafen	23.08.09 6254
24	Bauer	Gerhard		68002	Mannheim	15.02.09 123,56
25	Meier	Franz		68003	Mannheim	17.11.09 1925
26	Maier	Stefan		80003	München	14.01.09 0
27	Bühn	Walter		80042	München	18.02.09 0

Für diese Abbildung selektierte ich A1:A13, wählte DATEN|TEXT IN SPALTEN... und in Schritt 1 *Getrennt*, selektierte in Schritt 2 das Semikolon als Trennzeichen, aktivierte in Schritt 3 für die Spalten »Straße« und »Telefon« die Option *Spalten nicht importieren (überspringen)* und gab als Zielbereich A16 an.

19.4.2 Feste Spaltenbreiten

Die Option *Feste Breite* aus Schritt 1 ist für Fälle gedacht, in denen ein Statistik- oder ähnliches Programm feste Spaltenbreiten verwendet und eine Zahlenfolge wie

12,4043,5236,78
01,0082,7422,10

komprimiert in der Form

12,4043,5236,78
01,0082,7422,10

speichert. In diesem Beispiel enthält die erste Spalte Zahlen, die immer zwei Vorkommastellen und zwei Nachkommastellen besitzen, also inklusive Komma fünf Zeichen lang sind; und die zweite Spalte immer sechs Zeichen lange Zahlen und die dritte immer vier Zeichen lange Zahlen.

Bei derart aufgebauten Dateien wählen Sie in Schritt 1 *Feste Breite*. Schritt 2 sieht dann folgendermaßen aus (Abbildung 19.13):

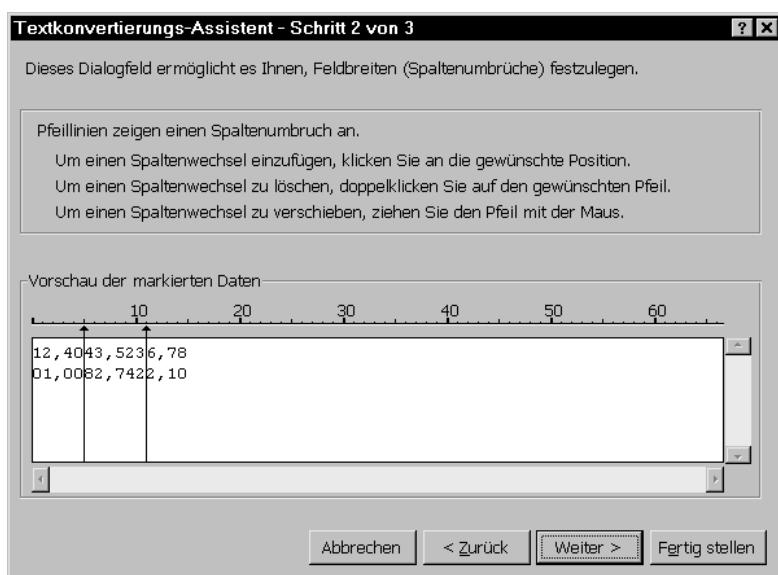


Abbildung 19.13:
Spaltenbreiten
definieren

Die vertikalen Trennlinien zwischen den einzelnen Spalten sind zunächst noch nicht vorhanden und müssen manuell eingefügt werden.

Sie geben Excel bekannt, wo jeweils eine neue Spalte beginnen soll, indem Sie in der Linealzeile auf die betreffende Position klicken, um dort eine vertikale Trennlinie einzufügen. Ein Doppelklick entfernt eine Trennlinie und per Anklicken und Ziehen können Sie sie verschieben.

Werden die Spaltenbreiten auf diese Weise angepasst, wird die Datei anschließend fehlerlos importiert.

19.4.3 Variable Leerzeichenfolgen

Ein großes Problem stellen Textdateien dar, die mehrere Leerzeichen als Spaltentrenner verwenden. Das Problem entsteht dadurch, dass Sie in Ihrer Textverarbeitung die einzelnen Spalten durch Tabulatoren trennen, manche Textverarbeitungen/Editoren diese Tabulatoren jedoch beim Speichern durch variabel lange Leerzeichenfolgen ersetzen.

Beispielsweise sind die Daten aus Abbildung 19.3 spaltengerecht untereinander angeordnet. Enthält die Datei zu diesem Zweck Tabulatoren, können Sie sie problemlos importieren, indem Sie als Trennzeichen *Tab* aktivieren.

Ersetzt Ihre Textverarbeitung oder Ihr Editor jedoch die Tabulatoren beim Speichern durch Leerzeichen, so befinden sich zwischen den einzelnen Werten anschließend statt eines Tabulators mehrere Leerzeichen – wobei deren Anzahl unterschiedlich ist und von der jeweiligen Textlänge und der Spaltenbreite abhängt.

Wird die in Abbildung 19.3 abgebildete Datei (die im Format *Formatierter Text* gespeichert wurde) mit der Option *Getrennt* (1. Schritt) und mit *Leerzeichen* als Trennzeichen und deaktivierter(!) Option *Aufeinander folgende Trennzeichen als ein Zeichen behandeln* (2. Schritt) importiert, erfolgt der Import fehlerhaft, wie bereits die Vorschau zeigt (Abbildung 19.14).

Zwischen den Texten »Name« und »Vorname« befanden sich mal zwei, mal drei Leerzeichen. Da angeblich jedes Leerzeichen den Beginn einer neuen Spalte einleitet, beginnt »Vorname« entsprechend viele Spalten neben »Name«.

Das Problem ist meist gelöst, wenn Sie in Schritt 2 zusätzlich zu *Leerzeichen* die Option *Aufeinander folgende Trennzeichen als ein Zeichen behandeln* aktivieren, da Excel dann eine beliebig lange Leerzeichenfolge als ein einziges Trennzeichen interpretiert.

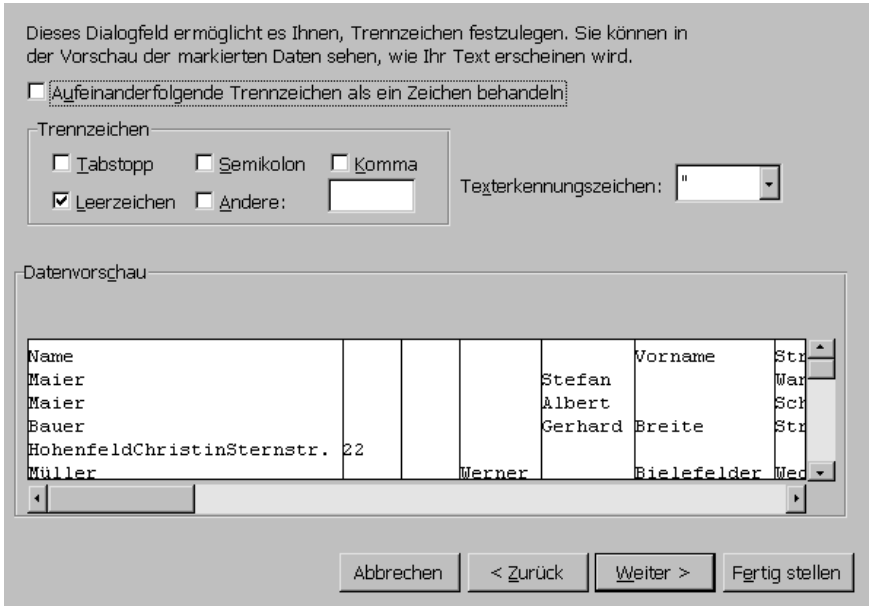


Abbildung 19.14:
Fehlerhafter
Datei-Import

Die in Abbildung 19.3 abgebildete Datei würde trotzdem noch nicht völlig korrekt importiert werden, da sich ab und zu zwischen mehreren Spalten *überhaupt kein Trennzeichen* befindet, dafür jedoch teilweise genau umgekehrt sich Leerzeichen in eigentlich *zusammenhängenden Texten* wie »Bielefelder Weg 2« befinden: (Abbildung 19.15).

	A	B	C	D	E	F	G
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Dat
2	Maier	Stefan	Wannstr.6	80003München	089/826254	14.01.09	
3	Maier	Albert	Schwalmweg	20	67005Ludwigshafen	21.01.09	
4	Bauer	Gerhard	Breite	Str.10	68002Mannheim	0621/749822	15.
5	Hohenfeld	22	67009Ludwigshafen	23.08.09	6254		
6	Müller	Werner	Bielefelder	Weg	2	10023Berlin	0372/73
7	Braun	Otto	Hansestr.	75	40007Düsseldorf	0125/262529	18.
8	Schmidt	Anton	Weberweg	54	60002Heidelberg	0662/326738	21.
9	Maier	Stefanie	Schweriner	Weg	18	20034Hamburg	0326/83626
10	Bühn	Walter	Hans-Pinsel-Str.2	80042München	089/836395	18.02.09	
11	Irlbeck	Gustav	Maibachweg	21	80017München	089/726384	01.
12	Müller	Ottokar	Schmalweg	9	10065Berlin	0372/93651	10.
13	Meier	Franz	Stendaler	Weg	23	68003Mannheim	0621/83

Abbildung 19.15:
Nach wie vor fehlerhafter
Datei-Import

Wie Zeile 6 der Tabelle zeigt, wird dieser Datensatz daher nur bis zum Straßennamen »Bielefelder Weg 2« korrekt importiert, der zwar eine Einheit bildet, aber dennoch Leerzeichen enthält, also Spaltentrennzeichen.

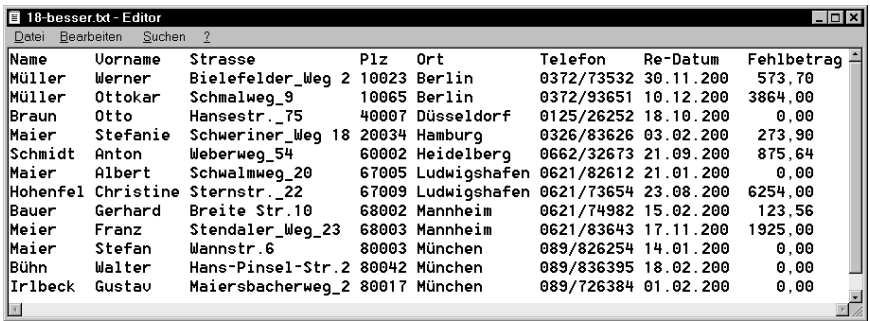
Da sich genau umgekehrt zwischen der Postleitzahl und dem darauf folgenden Ort, also zwischen zwei verschiedenen Elementen, überhaupt kein Leerzeichen befindet, werden beide ebenfalls fehlerhaft als ein einziger Wert behandelt (»67005Ludwigshafen« etc.).

Der Import derartiger Dateien klappt nur, wenn Sie folgende Regeln einhalten:

- 1. Ein zusammenhängender Text darf kein Leerzeichen enthalten.
- 2. Zwischen zwei Texten muss sich mindestens ein Leerzeichen befinden.

In der folgenden Datei wurden Leerzeichen durch Unterstriche »_« ersetzt (Abbildung 19.16).

Abbildung 19.16:
Geeignete Datei



Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
Müller	Werner	Bielefelder_Weg 2	10023	Berlin	0372/73532	30.11.200	573,70
Müller	Ottokar	Schmalweg_9	10065	Berlin	0372/93651	10.12.200	3864,00
Braun	Otto	Hansestr._75	40007	Düsseldorf	0125/26252	18.10.200	0,00
Maier	Stefanie	Schweriner_Weg 18	20034	Hamburg	0326/83626	03.02.200	273,90
Schmidt	Anton	Weberweg_54	60002	Heidelberg	0662/32673	21.09.200	875,64
Maier	Albert	Schwalweg_20	67005	Ludwigshafen	0621/82612	21.01.200	0,00
Hohenfel	Christine	Sternstr._22	67009	Ludwigshafen	0621/73654	23.08.200	6254,00
Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/74982	15.02.200	123,56
Meier	Franz	Stendaler_Weg_23	68003	Mannheim	0621/83643	17.11.200	1925,00
Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.200	0,00
Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.200	0,00
Irlbeck	Gustav	Maiersbacherweg_2	80017	München	089/726384	01.02.200	0,00

Nun wird sie mit *Leerzeichen* und der Option *Aufeinander folgende Trennzeichen als ein Zeichen behandeln* korrekt importiert.

Bei Dateien, die diese Bedingungen nicht erfüllen, müssen Sie leider trotz vorhandener Trennzeichen in Schritt 1 *Feste Breite* wählen. Schritt 2 sieht dann folgendermaßen aus (Abbildung 19.17):

Abbildung 19.17:
Schritt 2 bei festen
Spaltenbreiten



Textkonvertierungs-Assistent - Schritt 2 von 3

Dieses Dialogfeld ermöglicht es Ihnen, Feldbreiten (Spaltenumbrüche) festzulegen.

Pfeillinien zeigen einen Spaltenumbruch an.

Um einen Spaltenwechsel einzufügen, klicken Sie an die gewünschte Position.

Um einen Spaltenwechsel zu löschen, doppelklicken Sie auf den gewünschten Pfeil.

Um einen Spaltenwechsel zu verschieben, ziehen Sie den Pfeil mit der Maus.

Vorschau der markierten Daten:

Name	Vorname	Strasse	Plz	Ort	Telefon
Müller	Werner	Bielefelder_Weg 2	10023	Berlin	0372/73532
Müller	Ottokar	Schmalweg_9	10065	Berlin	0372/93651
Braun	Otto	Hansestr._75	40007	Düsseldorf	0125/26252
Maier	Stefanie	Schweriner_Weg 18	20034	Hamburg	0326/83626
Schmidt	Anton	Weberweg_54	60002	Heidelberg	0662/32673

Abbrechen < Zurück Weiter > Fertig stellen

Diese Interpretation korrigieren Sie nun. Sie geben Excel bekannt, wo neue Spalten beginnen sollen, indem Sie in der Linealzeile an die betreffende Position klicken, um dort eine vertikale Trennlinie einzufügen. Ein Doppelklick entfernt eine Trennlinie und per Anklicken und Ziehen können Sie sie verschieben. Die optimale Trennung in diesem Fall würde so aussehen: (Abbildung 19.18)

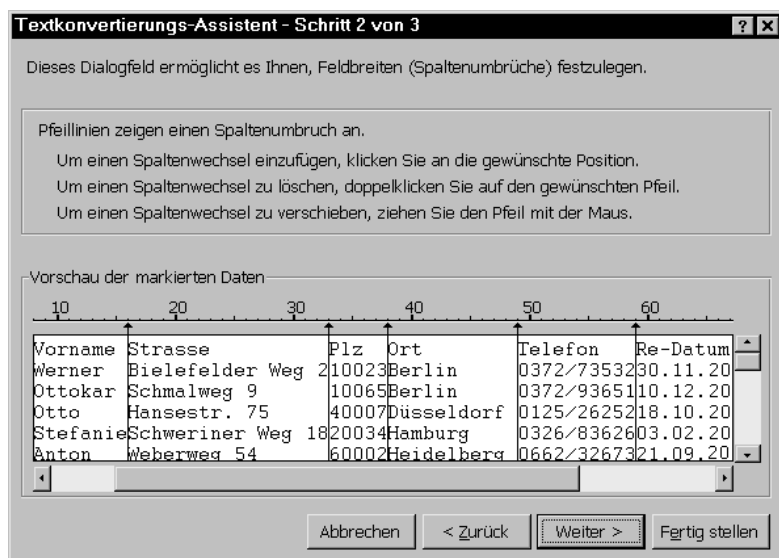


Abbildung 19.18:
Eigendefinierte
Spaltenbreiten

Am Abschneiden des Datums ist leider nichts zu ändern. Diese Spalte müssen Sie nach dem Datei-Import nachbearbeiten.

Im nun folgenden Kapitel erläutere ich ein kleines Hilfsprogramm von mir, das sich auf der Begleit-CD befindet und das Problem der variablen Leerzeichenfolgen elegant löst.



19.4.4 Ein nützliches Hilfsprogramm

Excel verwendet zur Trennung zwischen Vor- und Nachkommastellen das Dezimalkomma – im Gegensatz zu allen möglichen englischen Versionen von Kalkulationsprogrammen, Textverarbeitungen, Statistikprogrammen etc., die den Dezimalpunkt benutzen.

Speichert ein solches Programm die Zahl 12,3 als »12.3«, so erkennt Excel nicht, dass es sich um eine Zahl handelt und betrachtet diese Zeichenfolge einfach als Text, sodass später damit erfolgende Berechnungen zwangsläufig zur Fehlermeldung #WERT! führen. Vor dem Importieren müssen Sie somit unbedingt die Dezimalpunkte durch Kommata ersetzen!

Verwenden Sie dazu entweder einen Editor oder eine Textverarbeitung – oder aber mein Programm TAB_REPL.EXE, das sich auf der Begleit-CD befindet.

Es löst das im vorhergehenden Kapitel erläuterte Problem variabler Leerzeichenfolgen, indem es *beliebig lange* Leerzeichen-Sequenzen durch *je ein* Tabulatorzeichen ersetzt.



Zusätzlich ist es in der Lage, ein beliebiges Zeichen in der gesamten Datei durch ein anderes zu ersetzen – zum Beispiel den unerwünschten Dezimalpunkt durch ein Dezimalkomma.

Die Syntax:

```
TAB_REPL FILEIN FILEOUT CHARALT CHARNEU
```

FILEIN ist die zu behandelnde Eingabedatei und *FILEOUT* die zu erzeugende Ausgabedatei. Die Eingabedatei bleibt unverändert erhalten!

Angenommen, Sie rufen TAB_REPL mit

```
TAB_REPL TEST.DOC KOPIE.DOC . ,
```

auf: Das Programm liest die Datei TEST.DOC ein und erzeugt die neue Datei KOPIE.DOC. Sie ist eine Kopie von TEST.DOC, allerdings sind darin Leerzeichenfolgen durch je ein Tabulatorzeichen ersetzt.

Zusätzlich sind in KOPIE.DOC alle in der Originaldatei enthaltenen Punkte durch Kommata ersetzt.

Um das DOS-Programm TAB_REPL von Windows aus aufzurufen, gehen Sie am besten so vor:

- ➔ Öffnen Sie mit START|PROGRAMME|MS-DOS-EINGABEAUFFORDERUNG ein DOS-Fenster.
- ➔ Begeben Sie sich mit *CD \Ordner* in jenen Ordner, in dem sich die zu behandelnde Datei befindet, beispielsweise mit *CD \DEMOS\DATEIEN* in den Ordner DATEIEN, der sich im Ordner DEMOS befindet.
- ➔ Rufen Sie danach TAB_REPL mit LAUFWERK:\PFAD\TAB_REPL PARAMETER auf, also unter Angabe des Laufwerks und des Ordners, in dem sich dieses Programm befindet; zum Beispiel so, falls sich TAB_REPL im Ordner UTILS von C: befindet:

```
C:\UTILS\TAB_REPL TEST.DOC KOPIE.DOC . ,
```


Setzen wir voraus, dass TEST.DOC folgenden Inhalt besitzt (Leerzeichen stelle ich als »-« dar, um das Abzählen zu erleichtern):

```
19745---7.73----1550.40
```

Die zugehörigen ASCII-Codes:

```
49 57 55 52 53 32 32 32 55 46 55 51 32 32 32 32 49 53 53 48 46 52 48
```

KOPIE.DOC enthält nun folgende Zeichenkette (das Tabulatorzeichen stelle ich als »^« dar):

```
19745^7,73^1550,40
```

Auch hier wieder die zugehörigen ASCII-Codes:

```
49 57 55 52 53 9 55 44 55 51 9 49 53 53 48 44 52 48
```

Beachten Sie bitte, dass nicht nur die beiden Leerzeichen-Sequenzen »32 32« und »32 32 32 32« durch je ein Tabulatorzeichen mit dem Code 9 ersetzt wurden, sondern zusätzlich der ASCII-Code 46 (»,.«) durch den Code 44 (»,«) ersetzt wurde, sodass die Zahlen nun Dezimalkommata statt wie zuvor Dezimalpunkte enthalten und die Trennung der Nachkommastellen nun so erfolgt, wie es Excel erwartet!

Wollen Sie TAB_REPL nur zum Ersetzen von Leerzeichenfolgen durch Tabulatoren verwenden, aber kein Zeichen durch ein anderes ersetzen, geben Sie als Argumente CHARALT und CHARNEU einfach das gleiche Zeichen an:

```
TAB_REPL TEST.DOC KOPIE.DOC X X
```

Dieser Aufruf erzeugt aus der Datei TEST.DOC die Datei KOPIE.DOC, in der Leerzeichen durch Tabulatoren ersetzt wurden, und ersetzt zusätzlich das Zeichen »X« durch »X« – was natürlich nichts am Datei-Inhalt ändert.

TAB_REPL kann beliebig viele Dateien auf einmal behandeln, wenn alle identische Endungen besitzen!

Stellen Sie sich vor, Sie haben ein Verzeichnis mit 500 Aktienkursen, die im Textformat in 500 separaten Dateien mit der Endung ASC gespeichert sind, BASF.ASC, DAIMLER.ASC etc. – und alle enthalten Dezimalpunkte statt Dezimalkommata.

Mit einem einzigen Aufruf von TAB_REPL können Sie alle 500 Dateien behandeln. Dazu müssen Sie wissen, dass TAB_REPL alle Dateien im aktuellen Verzeichnis behandelt, auf die die als Argument FILEIN angegebene Dateimaske zutrifft.



Die neu erzeugten Dateien besitzen den gleichen Namen wie die Originale, aber auch die von Ihnen mit der Maske *FILEOUT* festgelegte Erweiterung (bzw. überhaupt keine Erweiterung, wenn Sie keine angaben. Beispiel: *TAB_REPL *.ASC *. ,*):

- ➔ Als Argument *FILEIN* übergeben Sie somit eine Dateimaske, die alle zu behandelnden Dateien erfasst, zum Beispiel »*.ASC«, wenn alle Kursdateien die Endung ASC besitzen.
- ➔ Außerdem übergeben Sie als Argument *FILEOUT* eine Maske, in der der Dateiname ebenfalls ausmaskiert und nur die Namensweiterung der erzeugten Dateien definiert ist, zum Beispiel »*.EXC«.

Wollen Sie alle Dateien im aktuellen Verzeichnis behandeln, die die Endung .ASC besitzen, verwenden Sie daher folgenden Aufruf:

```
TAB_REPL *.ASC *.EXC . ,
```

TAB_REPL sucht die erste Datei, die mit .ASC endet und findet zum Beispiel *BASF.ASC*. Die Datei wird behandelt und das Ergebnis in einer Datei mit dem gleichen Namen und der von Ihnen angegebenen Erweiterung gespeichert, also in *BASF.EXC*.

Danach sucht *TAB_REPL* die zweite Datei, auf die die Maske »*.ASC« zutrifft, und findet zum Beispiel *DAIMLER.ASC*. Also wird auch diese Datei behandelt und das Ergebnis in *DAIMLER.EXC* gespeichert und so weiter, bis alle Dateien behandelt sind, auf die die Dateimaske zutrifft.

19.5 Spezifische Dateiformate

Excel unterstützt außer den allgemeinen Textformaten mehrere programm-spezifische Formate, die ich nun anhand einiger ausgewählter Beispiele behandle.

19.5.1 Multiplan

SYLK-Format

Für den Datenaustausch mit Multiplan wird das Microsoft-eigene SYLK-Format benutzt. Sie müssen Ihre Excel-Dateien im SYLK-Format speichern, wenn Sie diese Dateien mit Multiplan weiterverarbeiten wollen.

Ausnahme: Benutzen Sie Multiplan ab Version 4.0, so können Sie Ihre Excel-Tabellen im Excel-2.1-Format speichern, die diese Multiplan-Version direkt lesen kann.

Beim Speichern von Multiplan aus müssen Sie die Speicher-Option »SYLK-Format« wählen, wenn Sie diese Dateien in Excel einlesen wollen.

Nach dem Speichern einer Datei unter Multiplan im SYLK-Format und dem Einlesen in Excel werden Zahlen, Operatoren und numerische Formate korrekt gelesen und in entsprechende Excel-eigene Formate übersetzt.

Probleme kann es allerdings geben, wenn Sie eine englische Multiplan-Version benutzen. Darauf war es zurückzuführen, dass ich beim Einlesen einer im SYLK-Format gespeicherten Multiplan-Tabelle in Excel Schwierigkeiten bekam. Excel wollte zum Beispiel den Aufruf der Multiplan-Funktion *LOOKUP* nicht akzeptieren – kein Wunder, schließlich heißt die entsprechende eingedeutschte Excel-Funktion *VERWEIS*.

Umgekehrt liest Multiplan von Excel im SYLK-Format gespeicherte Tabellen zwar ein, versteht jedoch bei weitem nicht alles, da Excel ungleich mehr Features besitzt als Multiplan.

Verschiedene Formate (Schriftarten, Ausrichtung etc.) werden von Multiplan einfach ignoriert, was nicht weiter tragisch ist.

Unangenehmer ist, dass Multiplan einige Excel-Funktionen nicht kennt, was zu einer entsprechenden Fehleranzeige führt und bei verknüpften Dateien nur einfache externe Bezüge erlaubt. Die in Excel möglichen komplexen Bezüge werden nicht unterstützt.

In der Praxis sind daher nach dem Einlesen einer Excel-Tabelle in Multiplan oft noch größere Überarbeitungen der importierten Tabelle erforderlich.

19.5.2 Lotus und Symphony

Ähnliches wie für Multiplan gilt auch für Lotus- und Symphony-Tabellen. Nach dem Einlesen werden darin enthaltene Werte, Formatierungen, Namen und Formeln in Excel korrekt wiedergegeben. Bei nicht standardmäßig von Excel unterstützten Zahlenformaten erzeugt das Programm ein entsprechendes neu definiertes Format.

Excel unterstützt Lotus- und Symphony-Funktionen, auch wenn sich die Funktionsbezeichnungen und die Arbeitsweise teilweise unterscheiden. So heißt zum Beispiel die Lotus-Funktion *F* bei Excel *T* und *@EIGENNAME* bei Excel *GROSS2*. Die Übersetzung in Excel-Funktionsnamen und die Anpassung der Funktionsparameter nimmt Excel jedoch automatisch vor.

Ein wichtiger Unterschied, der nicht angepasst wird, betrifft die *GANZZAHL*-Funktion, wenn sie auf negative Zahlen angewandt wird. Bei Lotus ist das Resultat dieser Funktion immer der ganzzahlige Anteil einer Zahl, wobei die Nachkommastellen einfach entfernt werden. Daher liefert die Formel *@GANZZAHL(-10,5)* in Lotus das Resultat -10. Da die *GANZ-*

ZAHL-Funktion bei Excel jedoch immer die nächstkleinere ganze Zahl ermittelt, liefert in Excel *GANZZAHL*(-10,5) als Resultat 11 – eine Abweichung, die Sie selbst korrigieren müssen!

Excel ist ausgezeichnet auf das Einlesen und Konvertieren von Lotus-Tabellen eingerichtet, um Umsteigern von Lotus auf Excel die Übernahme ihrer Tabellen zu ermöglichen. Nach dem Öffnen einer Lotus-Datei können Sie darin enthaltene Makros ausführen, indem Sie die **[Strg]**-Taste gedrückt halten und den Anfangsbuchstaben des Makronamens eingeben. Das Resultat ist nicht unbedingt perfekt, da Lotus-Add-Ins nicht unterstützt werden.

Der umgekehrte Weg ist problematischer. Lotus kann weder mit von Excel erstellten Grafiken noch mit Excel-Makros etwas anfangen. Außerdem besitzt Excel viele Funktionen, zu denen es in Lotus kein Äquivalent gibt, ebenso wenig wie zu verschiedenen Formatierungen. Auch hier ist somit wie fast immer beim Exportieren von Excel-Tabellen in andere Kalkulationsprogramme ein Menge Nacharbeit erforderlich.

19.5.3 dBase

Der Datenaustausch mit dBase ist sehr unproblematisch, da wie bei Lotus sogar unterschiedliche Versionen unterstützt werden. Allerdings werden dabei weder Formate noch Formeln exportiert oder importiert, sondern ausschließlich in Listen enthaltene Texte und Zahlen. Nehmen wir an, Sie speichern die folgende Tabelle im dBase-IV-Format (Abbildung 19.19).

Abbildung 19.19:
Datenbasis

	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	30.11.09	573,70
3	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.09	3864,00
4	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.10.09	0,00
5	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.02.09	273,90
6	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.09.09	875,64
7	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826125	21.01.09	0,00
8	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.08.09	6254,00
9	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.02.09	123,56
10	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.11.09	1925,00
11	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.09	0,00
12	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.02.09	0,00
13	Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/726384	01.02.09	0,00
14								
15	Name		Datum					
16	FALSCH		11.12.09					
17								

Excel exportiert beim Speichern nur die momentan aktive Datenbank, also jene, in der sich die aktive Zelle befindet, und ignoriert alle sonstigen Tabellenelemente, im Beispiel die Zeilen 15 und 16.

Die resultierende DBF-Datei im dBase-Format kann problemlos mit dBase selbst, mit Clipper oder mit beliebigen Utilities weiterbearbeitet werden, welche die dBase-Dateien lesen können (Abbildung 19.20).



Abbildung 19.20:
Exportierte Datei im
dBase-Format

Sie können anschließend Indexdateien erzeugen, Such- oder Sortierläufe starten und so weiter.

Umgekehrt kann Excel eine dBase-Datei direkt importieren und daraus eine Excel-Datenbank erstellen.

Wollen Sie eine dBase-Datei ohne Abstriche bearbeiten, also auch bereits definierte Indizes verwenden oder gar neue definieren, müssen Sie MSQuery statt Excel verwenden, ein vollwertiges, von Excel unabhängiges Datenbankprogramm, mit dem Sie auf bereits vorhandene dBase-, SQL-, Excel- oder Datenbanken im Textformat zugreifen oder alternativ neue Datenbanken definieren können (siehe Kapitel 14, »Microsoft Query – Zugriff auf externe Datenbanken«).



19.6 Windows-Programme

Zum Datenaustausch zwischen Excel und anderen Windows-Applikationen können Sie entweder die Zwischenablage (Clipboard) benutzen oder aber Texte oder Objekte per OLE einfügen bzw. verknüpfen.

TIPP

Unterstützen beide Programme, zwischen denen Sie Daten austauschen wollen (wie Excel ab der Version 5.0 und WinWord ab 6.0), OLE in der Version 2.0 (1.0 genügt nicht!), und sind Sie nicht an dynamischen Verknüpfungen interessiert, bei denen das eingebettete Objekt bei Änderungen der Quelldaten automatisch aktualisiert wird, können Sie die drei Kapitel übergehen und gleich Kapitel 19.6.4, »OLE 2.0«, lesen.

19.6.1 Die Office-Zwischenablage

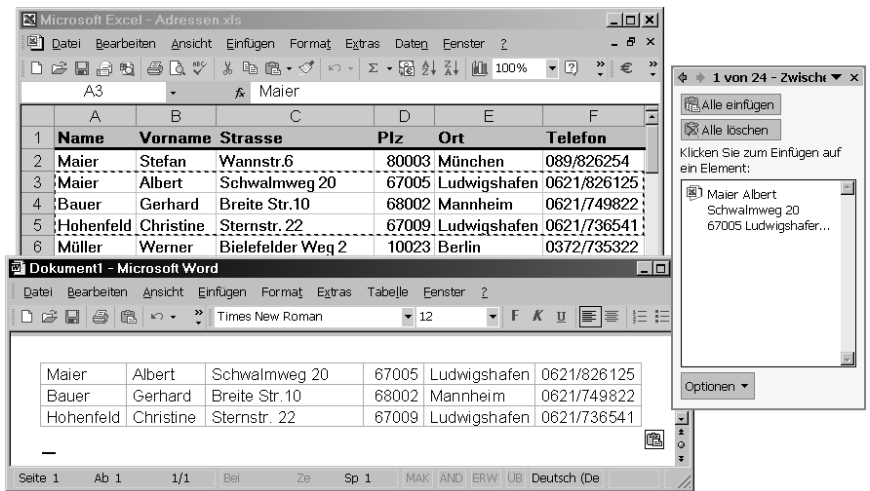
Die Zwischenablage besitzt den großen Vorteil, mit jedem Windows-Programm zusammenzuarbeiten.

Nehmen wir an, Sie wollen einen Ausschnitt einer Excel-Tabelle, der einige Adressen enthält, in einen Word-Text integrieren: Sie markieren den Ausschnitt und wählen BEARBEITEN|KOPIEREN, um ihn in die Zwischenablage zu befördern, deren Inhalt jeder anderen Windows-Applikation zugänglich ist.

Nun rufen Sie Word auf. Sollte das am verfügbaren Speicherplatz scheitern, schließen Sie zuvor das Excel-Fenster; der Inhalt der Zwischenablage bleibt dabei erhalten.

In Word bewegen Sie den Cursor zur Textstelle, an der Sie die Tabelle einfügen wollen und wählen BEARBEITEN|EINFÜGEN (Abbildung 19.21).

Abbildung 19.21:
Einfügen einer
Tabelle über die
Zwischenablage



Nach dem Einfügen erscheint in Word übrigens ein Einfügen-Smart Tag, mit dem Sie das Format festlegen können, in dem die Daten eingefügt werden, und mit dem Sie festlegen können, ob nur reiner Text eingefügt wird oder das Originalformat der Daten erhalten bleibt oder die Formatierung einer Word-Tabelle angepasst wird, in die Sie die Daten eventuell einfügten, oder ob Sie das Format nun individuell festlegen wollen.

Zusätzlich gibt es eine Option, um die eingefügten Daten mit der Excel-Tabelle zu verknüpfen, damit sie bei Änderungen dieser Tabelle automatisch entsprechend aktualisiert werden (siehe Kapitel 19.6.2, »Verknüpfen«).



Vielleicht wollen Sie in Ihren Text zusätzlich eine Grafik integrieren, um das Zahlenmaterial etwas anschaulicher zu präsentieren: Selektieren Sie dazu beispielsweise ein Excel-Diagramm, wählen Sie wieder BEARBEITEN|KOPIEREN, um die Grafik in die Zwischenablage zu befördern, bewegen Sie in WinWord den Cursor zur gewünschten Position und fügen Sie die Grafik dort mit BEARBEITEN|EINFÜGEN ein (Abbildung 19.22).

Grafik exportieren

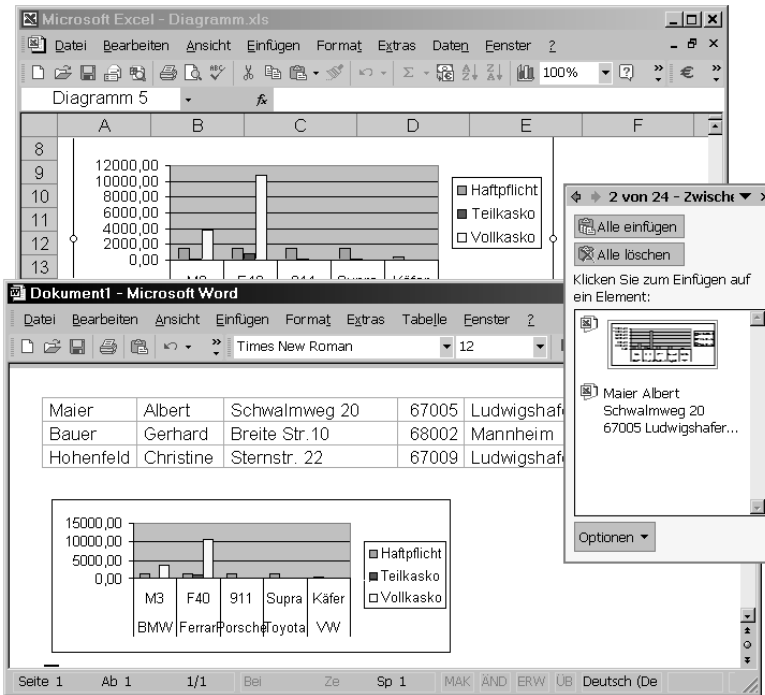


Abbildung 19.22:
Einbindung einer Grafik über die Zwischenablage

Der Datenaustausch über die Zwischenablage funktioniert in beiden Richtungen. Sie können beliebige Zahlen oder Zeichenketten eines Textes oder einer Grafik in einem anderen Windows-Programm selektieren, in die Zwischenablage befördern und in einer Excel-Tabelle dort einfügen, wo sich gerade der Zellcursor befindet. Die betreffende Zelle wird als linke obere Ecke des Einfügebereichs benutzt.

Die Office-Zwischenablage unterscheidet sich von der normalen Windows-Zwischenablage und kann bis zu 24 verschiedene Dokumente aller Art aufnehmen. Sie können mehrfach nacheinander Daten in die Zwischenablage kopieren und sich danach entscheiden, welche dieser Daten Sie in ein Dokument einfügen wollen.

Die Office-Zwischenablage

Das Prinzip: Kopieren Sie irgendwelche Daten und danach erneut etwas, enthält die Office-Zwischenablage anschließend entsprechend zwei verschiedene Dokumente und wird automatisch eingeblendet.

All diese Dokumente werden im Register *Zwischenablage* des Aufgabenbereichs angezeigt, das Sie durch zweifaches Drücken von **[Strg]+[C]** oder mit dem Befehl **BEARBEITEN|OFFICE-ZWISCHENABLAG**e einblenden (Abbildung 19.23).

Abbildung 19.23:
Office-Zwischenablage



Im Beispiel enthält die Office-Zwischenablage Word-, Access- und Excel-Dokumente. Schwebt der Cursor über einem davon, erscheint ein Listepfeil, der nach dem Anklicken eine Liste öffnet, deren Befehle Ihnen ermöglichen, das betreffende Dokument an der aktuellen Cursorposition einzufügen oder aus der Zwischenablage zu löschen.

Mit den Schaltflächen ganz oben können Sie alle Dokumente gleichzeitig einfügen bzw. löschen.

Zusätzlich ermöglicht das Listenfeld *Optionen* das Verhalten der Office-Zwischenablage zu beeinflussen, beispielsweise festzulegen, ob sie nach dem Kopieren von Dokumenten automatisch eingeblendet werden soll oder ob in der Windows-Statusleiste ein kleines Symbol einblendete wird, das Ihnen

durch Anklicken mit der rechten Maustaste jederzeit den Zugriff auf alle verfügbaren Steuerungsoptionen ermöglicht, unter anderem auch auf die recht nützliche Option, mit dem »Sammeln« kopierter Objekte in der Office-Zwischenablage aufzuhören.

19.6.2 Verknüpfen

Beim Verknüpfen werden zwischen zwei Windows-Programmen Verbindungen ähnlich den externen Verknüpfungen zwischen zwei Excel-Tabellen aufgebaut, sodass nach jeder Änderung der Daten im Ursprungsprogramm (z.B. Word) die entsprechenden Daten im Zielprogramm (z.B. Excel) automatisch aktualisiert werden.

Ich führe Ihnen das nun stellvertretend für andere Windows-Applikationen an Word vor. Nehmen wir an, ein Word-Dokument enthält eine Tabelle, die in eine Excel-Tabelle eingefügt werden soll – und dort jedesmal automatisch aktualisiert werden soll, wenn sich die Originaldaten im Word-Dokument ändern.

Zunächst selektieren Sie im Word-Dokument den betreffenden Datenbereich und wählen KOPIEREN (Abbildung 19.24).

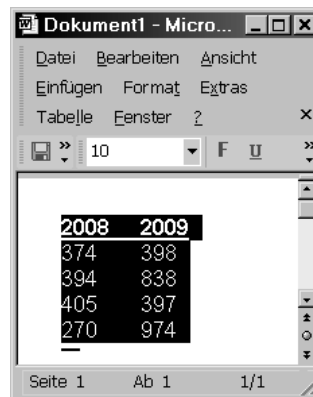


Abbildung 19.24:
Dynamischer
Datenaustausch von
Word zu Excel

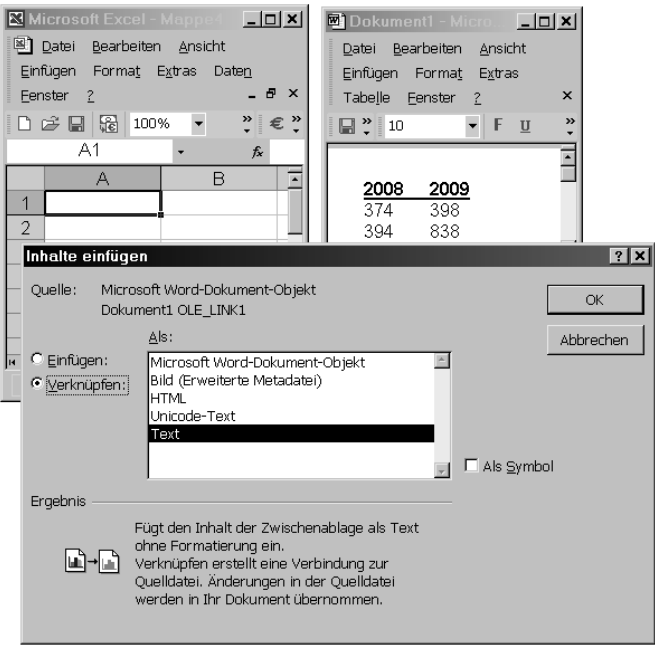
Danach aktivieren Sie Excel, wählen jedoch statt EINFÜGEN den Befehl BEARBEITEN|INHALTE EINFÜGEN... .

*Verknüpfung
erzeugen*

Befindet sich dabei wie hier ein Objekt aus einer anderen Anwendung in der Zwischenablage, öffnet sich statt des gewohnten nun folgendes Dialogfeld (Abbildung 19.25).

Der vorselektierte Eintrag *Einfügen* würde die Tabelle statisch einfügen. Sie müssen unbedingt *Verknüpfen* aktivieren, um die gewünschte Verbindung erstellen zu lassen.

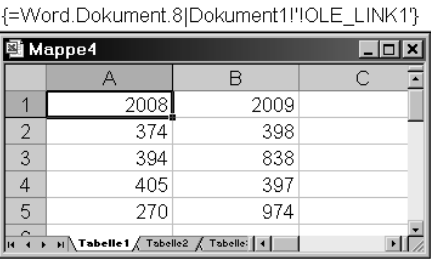
Abbildung 19.25:
Verknüpfung
einfügen



Zusätzlich haben Sie im Listenfeld Als verschiedene Optionen: »Microsoft Word-Dokument-Objekt« fügt die Tabelle als Objekt ein, das genauso aussieht wie die Word-Tabelle selbst inklusive aller Formatierungen, die diese besitzt. »Abbildung (Erweiterte Metadatei)« fügt sie als Grafikobjekt ein (siehe Kapitel 20, »Grafikobjekte«).

Mit »Text« verlieren Sie jegliche Formatierungen. Dennoch ziehe ich persönlich diese Alternative vor, da die Daten auf diese Weise am harmonischsten in die Excel-Tabelle integriert werden (Abbildung 19.26).

Abbildung 19.26:
OLE-Verknüpfung



In der Bearbeitungsleiste sehen Sie die für die Verknüpfung mit Word verantwortliche Formel, bei der es sich um eine Matrixformel handelt, die aus folgenden Elementen besteht:

=Anwendung | Thema!Element

Dank dieser Formel wirkt sich jede Änderung der Tabelle in Word augenblicklich auf die Excel-Tabelle aus: Ändern Sie in der Word-Tabelle einen Wert, wird sofort die entsprechende Zelle der Excel-Tabelle aktualisiert.

Verknüpfungen sind einseitig orientiert: Eine Anwendung ist der Server, der die Daten liefert (im Beispiel Word), die andere der Client, der sie empfängt (im Beispiel Excel).

*Verknüpfungs-
richtung*

Es ist nicht möglich, diese Richtung umzukehren, beispielsweise nun in der Excel-Tabelle eine Zahl zu ändern und zu erwarten, dass daraufhin der zugehörige Teil des Word-Dokuments entsprechend aktualisiert wird.

In umgekehrter Richtung verläuft der Vorgang genauso: Sie selektieren in Excel den benötigten Tabellenbereich und wählen KOPIEREN. Im Fremdprogramm, beispielsweise in Word, wählen Sie erneut INHALTE EINFÜGEN..., worauf sich je nach Programm das erläuterte Dialogfeld öffnet oder aber eines mit leicht anderem, funktional jedoch mit Sicherheit identischem Inhalt.

Öffnen Sie ein Dokument, das eine Verknüpfung enthält, beispielsweise eine Excel-Tabelle, die eine Verknüpfung zu einem Word-Dokument enthält, so werden Sie gefragt, ob die Verknüpfung aktualisiert werden soll (Abbildung 19.27).

*Verknüpfung
aktualisieren und
bearbeiten*

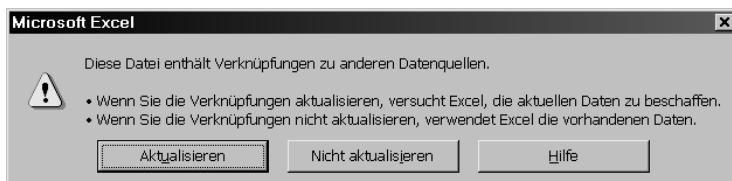


Abbildung 19.27:
Aktualisierung

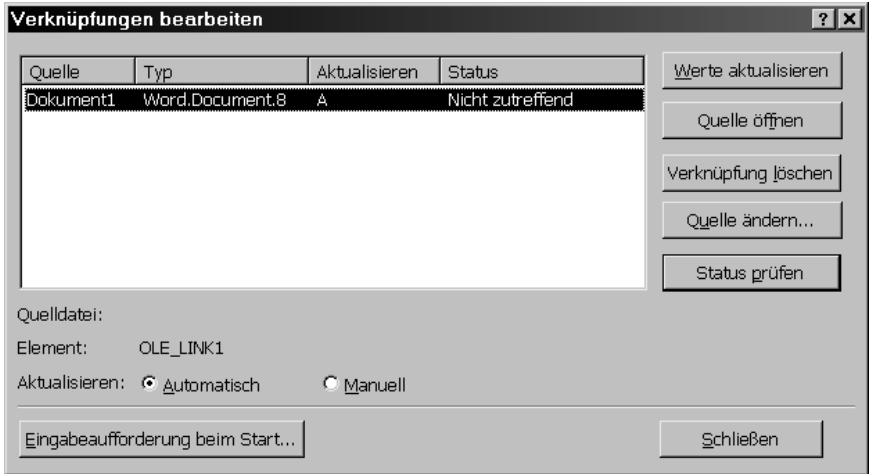
Bejahen Sie die Frage, benutzt Excel Word, um die in der Tabelle enthaltenen Daten zu aktualisieren.

Mit BEARBEITEN|VERKNÜPFUNGEN... können Sie Verknüpfungen bearbeiten (Abbildung 19.28).

Die Liste enthält alle Verknüpfungen des aktiven Dokuments und gibt über den aktuellen Status der Verknüpfung (der mit *Status prüfen* aktualisiert werden kann) Auskunft und auf welches Dokument welcher Anwendung sich die Verknüpfung bezieht.

Selektieren Sie eine Verknüpfung und aktivieren Sie unter *Aktualisieren* »Manuell« statt der vorgegebenen Option »Automatisch«, so wird die Verknüpfung nicht mehr bei jeder Änderung des Objekts in der Server-Anwendung automatisch aktualisiert, sondern nur noch, wenn Sie dieses Dialogfeld öffnen, die Verknüpfung selektieren und mit der Schaltfläche *Werte aktualisieren* den Befehl dazu geben.

Abbildung 19.28:
Verknüpfungen
bearbeiten



Diese Option kann sehr sinnvoll sein, wenn Sie Dutzende oder gar Hunderte von Verknüpfungen in einer Tabelle benutzen. Enthält Ihre Excel-Tabelle beispielsweise Dutzende von Verknüpfungen zu einem Windows-DFÜ-Programm, mit dem Sie Aktienkurse per Modem einspielen, genügt es in der Praxis sicherlich, wenn diese Kurse nur auf Ihren Befehl hin aktualisiert werden, da viele aktive Verknüpfungen aufgrund der sehr langen Aktualisierungszeiten deutlich verlangsamen wirken.

Quelle öffnen öffnet die Server-Anwendung und darin wiederum das Dokument, zu dem die Verknüpfung besteht. *Quelle ändern...* ermöglicht Ihnen, die verknüpfende Formel in einem Eingabefeld zu editieren.

Verknüpfung löschen entfernt die Verbindung zum Objektlieferanten. Haben Sie beispielsweise Formeln eingefügt, bleiben nur die momentan resultierenden Werte erhalten, die nicht mehr aktualisiert werden können, wenn sie sich in der Quellanwendung ändern.

Öffnen Sie eine Mappe, die Verknüpfungen enthält, fragt Excel normalerweise nach, ob die Verknüpfungen aktualisiert werden sollen. *Eingabeaufforderung beim Start...* ermöglicht Ihnen, diese Meldung zu unterdrücken und festzulegen, ob die Verknüpfungen beim Öffnen der Mappe automatisch aktualisiert werden sollen oder nicht.

Haben Sie den Ausschnitt nicht als Text, sondern als Objekt eingefügt, ermöglichen die Unterbefehle von BEARBEITEN|...OBJEKT, das – zuvor selektierte – Objekt im zugehörigen Anwendungsprogramm zu öffnen oder zu bearbeiten oder es in ein anderes Format zu konvertieren.

19.6.3 Einfügen und Einbetten

Statt Verknüpfen können Sie im Inhalte einfügen-Dialogfeld die Option Einfügen wählen. Selektieren Sie unter Als den Eintrag »Text«, wird der Inhalt der Zwischenablage dabei auf die gleiche Weise wie mit BEARBEITEN|EINFÜGEN eingefügt. »Grafik« fügt ihn als Grafikobjekt ein (siehe Kapitel 20, »Grafikobjekte«).



Weitaus interessanter ist jedoch das Einfügen als Objekt, angewandt auf die im vorhergehenden Kapitel verwendete Word-Tabelle, die in ein Excel-Arbeitsblatt als Objekt eingefügt wird (Abbildung 19.29).

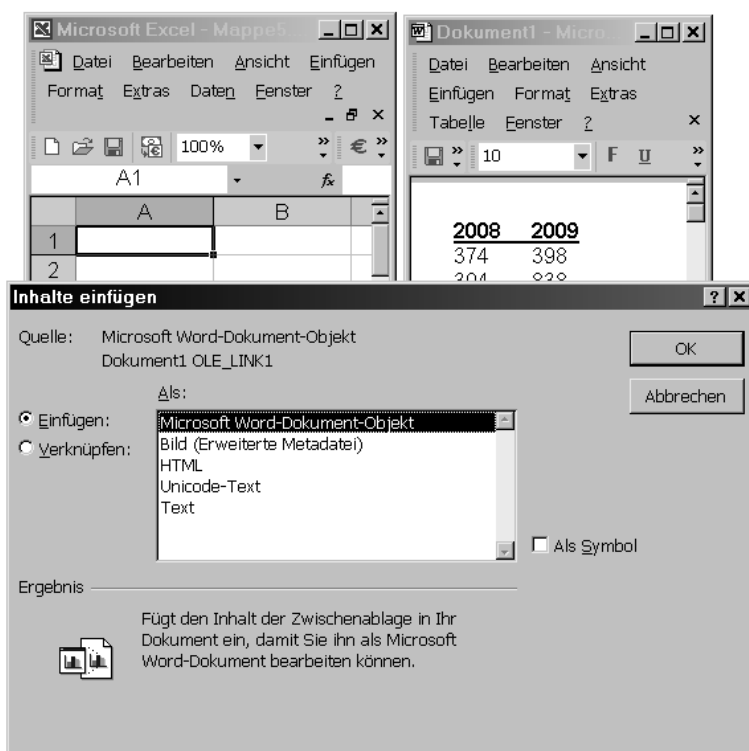


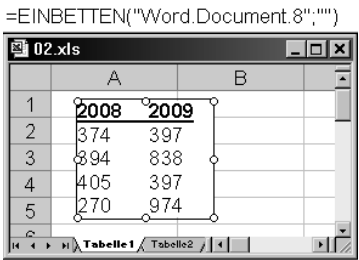
Abbildung 19.29:
Objekt einbetten

Statt von »Einfügen« wird in diesem Fall übrigens von »Einbetten« gesprochen. In der Bearbeitungsleiste sehen Sie die für diesen Vorgang zuständige Formel (Abbildung 19.30).

Der für den Anwender offensichtlichste Vorteil des Einbettens besteht darin, dass der Befehl BEARBEITEN|...OBJEKT|BEARBEITEN (einfacher: Doppelklick auf das eingebettete Objekt) genügt, um den Objektlieferanten, den Server, aufzurufen und das Objekt darin zu bearbeiten.

Objekt bearbeiten

Abbildung 19.30:
Eingebettetes
Objekt



Sind Sie mit der Bearbeitung fertig, wählen Sie DATEI|SCHLIESSEN UND ZURÜCKKEHREN ZU... oder schließen das Programm einfach. Windows fragt Sie daraufhin, ob das geänderte Objekt in der Client-Anwendung Excel aktualisiert werden soll. Antworten Sie mit *Ja*, werden Ihre Änderungen sofort in Excel sichtbar.

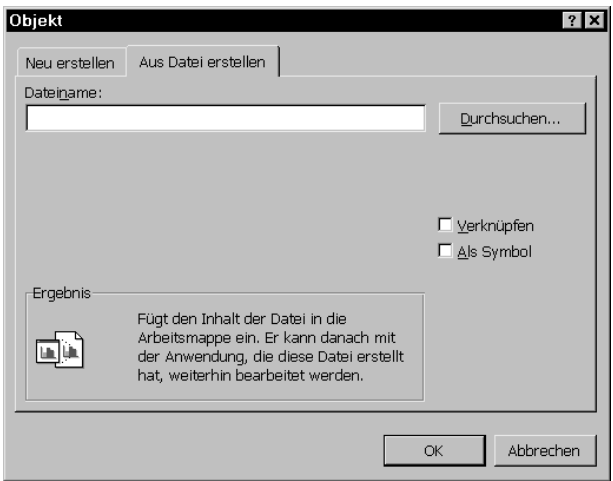


Dokument
einbetten und
erstellen

Unterstützt der Objektlieferant OLE 2.0, erfolgt die Bearbeitung sogar innerhalb von Excel selbst (siehe Kapitel 19.6.4, »OLE 2.0«). Zum Beenden der Bearbeitung genügt in diesem Fall ein Klick auf irgendeine Stelle außerhalb des Objekts.

Statt Teile eines Dokuments einzubetten, können Sie mit dem Befehl EINFÜGEN|OBJEKT... und dem Register *Aus Datei erstellen* des zugehörigen Dialogfelds auch gleich komplette Dokumente einbetten, indem Sie den Namen der gewünschten Datei eingeben oder aber auf die Schaltfläche *Durchsuchen...* klicken, um das *Dateiselektions*-Dialogfeld zu öffnen und die Datei darin auszuwählen (Abbildung 19.31).

Abbildung 19.31:
Dokumente
einbetten



Um ein noch nicht existierendes Objekt zu erzeugen und danach einzubetten, wählen Sie ebenfalls EINFÜGEN|OBJEKT..., aktivieren jedoch das Register *Neu erstellen* (Abbildung 19.32).

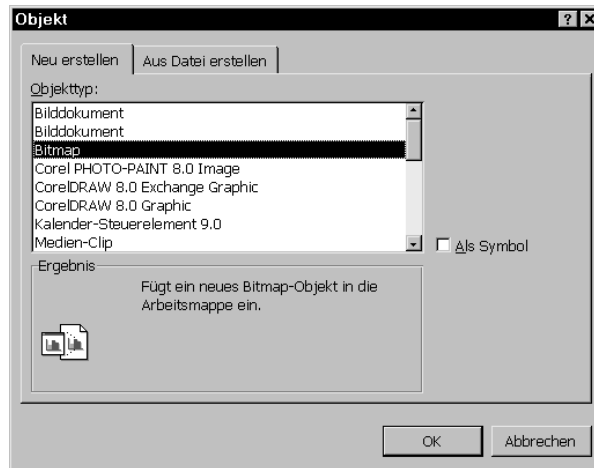


Abbildung 19.32:
Neues Objekt
erstellen

Das Dialogfeld listet alle momentan installierten Windows-Anwendungen auf, die OLE unterstützen.

Wenn Sie ein Objekt dieser Liste selektieren, bettet Excel das noch nicht vorhandene Objekt zunächst als weißen Fleck in die Tabelle ein und startet anschließend die betreffende Anwendung, zum Beispiel Paint oder Word, die sich in Excel integriert.

Sie erzeugen damit das gewünschte Objekt, klicken auf irgendeine Zelle außerhalb des Objekts und beantworten die Aktualisierungsfrage mit *Ja*.

19.6.4 OLE 2.0

Wird OLE ab der Version 2.0 außer von Excel auch vom zweiten Anwendungsprogramm erkannt, so wird das Einfügen, Einbetten und Editieren von Objekten erheblich komfortabler. Beides demonstriere ich mit Excel und Word.

Beide Anwendungen müssen geöffnet sein, Excel und Word. Selektieren Sie in Word den Ausschnitt, den Sie in eine Excel-Tabelle einfügen wollen und lassen Sie die Maustaste wieder los.

*Bequemes
Einfügen*

Klicken Sie nun auf den selektierten Textteil, halten Sie die Maustaste gedrückt, ziehen Sie den Text zur Excel-Tabelle und lassen Sie ihn dort fallen. Er wird aus dem Word-Dokument ausgeschnitten und dort in die Excel-Tabelle eingefügt, wo Sie ihn fallen lassen.

Halten Sie beim Ziehen die **[Strg]**-Taste gedrückt, so wird der Text übrigens nicht von Word in Excel verschoben, sondern kopiert.



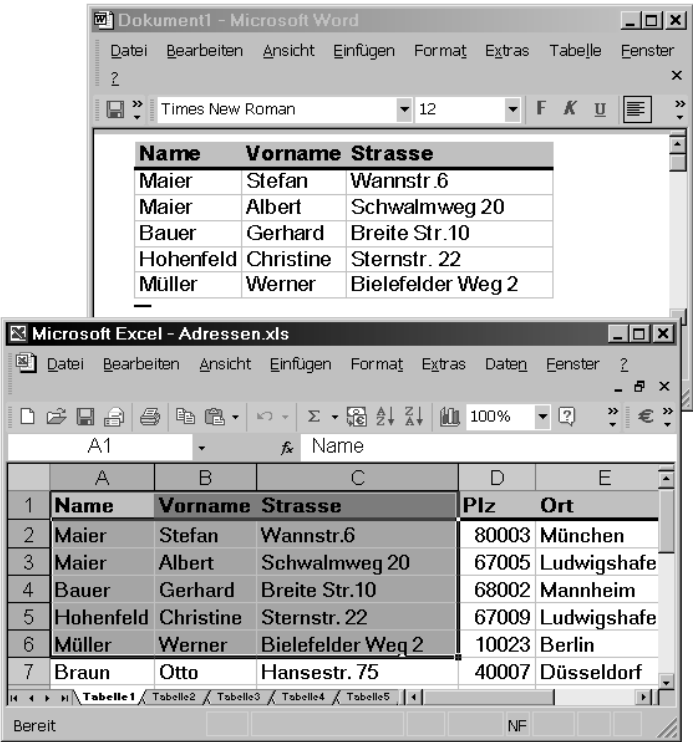
Diese Technik, die OLE 2.0-fähige Programme voraussetzt, entspricht dem in Kapitel 19.6.1, »Die Office-Zwischenablage«, beschriebenen Datenaustausch über die Zwischenablage beziehungsweise dem in Kapitel 19.6.3, »Einfügen und Einbetten«, erläuterten Einfügen als Text.

Voraussetzung dafür ist, dass die Anwendung, die das Objekt empfängt, damit wie mit einem eigenen Objekt umgehen kann. Im Beispiel ist das kein Problem, da Excel selbstverständlich mit ordinären Texten umgehen kann: Er wird genauso behandelt, als hätten Sie ihn manuell in der Zelle eingetippt.

Einbetten mit
OLE 2.0

Völlig anders verhält sich OLE 2.0, wenn es um ein Objekt geht, mit dem der Empfänger nichts anfangen kann; wenn Sie beispielsweise einen Bereich einer Excel-Tabelle in ein Word-Dokument verschieben und dort fallen lassen. Halten Sie dabei `Strg` gedrückt, wird das Objekt wieder kopiert und ist anschließend weiterhin in der Excel-Tabelle vorhanden (Abbildung 19.33).

Abbildung 19.33:
Unbekanntes Objekt
übergeben



Word ist nun einmal keine Tabellenkalkulation und kann mit einem Excel-Tabellenbereich eigentlich gar nichts anfangen. Dennoch klappt der Datenaustausch auch diesmal, wie die Abbildung zeigt. Dabei waren nur wenige

Schritte notwendig: Ich selektierte in Excel den Bereich A1:C6, ließ die Maustaste los, klickte einen Bereichsrand an und zog den Bereich bei gedrückter **[Strg]**-Taste in das Word-Dokument, wo ich ihn fallen ließ.

Nach dem Einfügen dieses Objekts per Drag&Drop genügt nun ein Doppelklick darauf, um den Objektlieferanten, also Excel, vor Ort zu aktivieren. Vorortaktivierung heißt, dass die Editierung in der gewohnten Umgebung erfolgt (Abbildung 19.34).

Vorortaktivierung

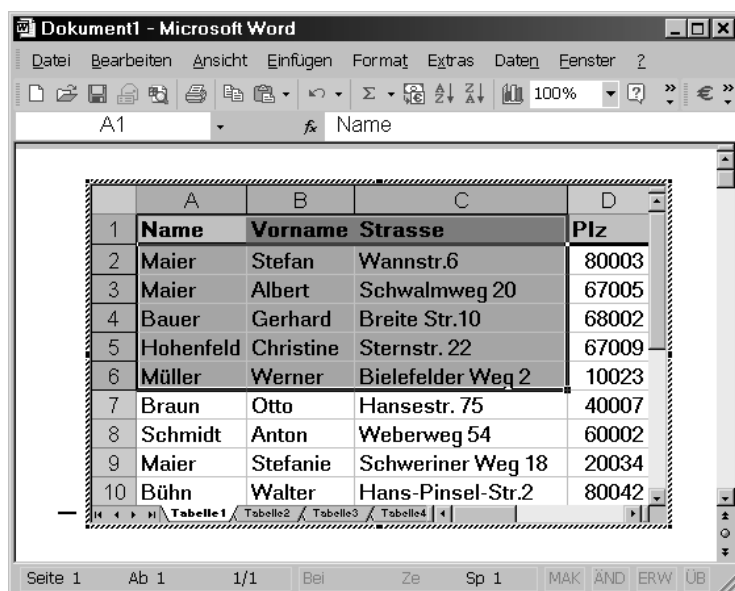


Abbildung 19.34:
Excel vor Ort
aktivieren

Excel fügt sich nahtlos in den umgebenden Rahmen ein, das heißt in die Anwendung, in der sich das Objekt befindet, auf das doppelgeklickt wurde, im Beispiel in Word.

Excel verzichtet sogar auf eine eigene Menüleiste. Stattdessen ist die Word-Menüleiste um die Excel-Menüs erweitert worden, beispielsweise um das Menü DATEN, sodass Sie den Tabellenausschnitt direkt in Word mit den gewohnten Excel-Techniken bearbeiten können.

Sie beenden die Editierung, indem Sie irgendeinen Punkt des Word-Dokuments anklicken, der sich außerhalb des eingebetteten Objekts befindet.

Teil 4 Grafikobjekte und Diagramme

Kapitel 20: Grafikobjekte	577
Kapitel 21: Bildobjekte aller Art	589
Kapitel 22: Diagramme erzeugen	597
Kapitel 23: Diagrammarten	629
Kapitel 24: Diagramme formatieren	655

Excel kann mit Grafikobjekten unterschiedlichster Art umgehen, nicht nur wie zu erwarten mit Diagrammen, die irgendwelche in einer Tabelle enthaltenen Daten in grafischer Form präsentieren, sondern auch mit allen möglichen sonstigen Grafikobjekten: mit Grafikdateien oder eingescannten Bildern, die Sie als Tabellenhintergrund verwenden können, mit Cliparts oder mit dem sehr spezifischen Landkarten-Feature.

Ich erläutere zunächst die allgemeinen Regeln beim Umgang mit Grafikobjekten, da sie auf all diese Objekte gleichermaßen zutreffen. Beispielsweise wird zum Formatieren immer ein objektspezifisches *Formatier*-Dialogfeld benutzt und auch das Vergrößern, Verkleinern oder Positionieren eines Grafikobjekts geschieht immer auf die gleiche Weise.

Danach gehe ich relativ kurz auf Objekte unterschiedlicher Art wie Auto-Formen, Clipart-Objekte und vor allem Landkarten ein, bevor ich mich dem eigentlichen Kern zuwende: den Diagrammen, die Daten einer Excel-Tabelle übersichtlich und präsentationsfähig aufbereiten.

Ich zeige Ihnen, wie Diagramme erstellt und nachträglich verändert werden, welche Diagrammtypen Excel zur Verfügung stellt und wie jedes einzelne Diagrammelement höchst individuell formatiert werden kann, egal ob es sich dabei um die Legende, die Achsenskalierung oder eine einzelne Säule eines Säulendiagramms handelt.

20 Grafikobjekte

Excel ist in der Lage, verschiedenste Grafikobjekte in eine Tabelle einzufügen, angefangen bei einfachen Linien und Kreisen bis hin zu mitgelieferten Cliparts, Zeichnungen oder eingescannten Vorlagen.

Ich erläutere nun, wie Sie diese Objekte einfügen und nach allen Regeln der Kunst manipulieren können. Zum Beispiel, um festzulegen, dass die Kreisumrandung rot und gestrichelt dargestellt sein soll, der Kreis mit gelber Farbe gefüllt wird und das gesamte Objekt an die darunter liegenden Zellen gebunden und mit diesen verschoben wird.

Anschließend erläutere ich den Umgang mit speziellen Bildobjekten und zeige Ihnen, wie Sie beispielsweise den Inhalt einer beliebigen Grafikdatei in eine Tabelle einfügen oder gar als Muster verwenden, mit dem die Balken eines Balkendiagramms gefüllt werden, oder aber, wie Sie mithilfe der WordArt-Symbolleiste teilweise recht kunstvolle Schriftvarianten in Ihre Tabelle einfügen und manipulieren.

Dieses Kapitel erläutert übrigens nicht nur den Umgang mit den erwähnten Objekten, die man noch als Spielerei abtun könnte, sondern stellt gleichzeitig eine Vorbereitung für den Umgang mit Landkarten und vor allem mit Diagrammen dar.

Praktisch alle nun erläuterten Techniken zur Selektion und Manipulation von Grafikobjekten werden nämlich auch zur Selektion und Manipulation von kompletten Diagrammen oder einzelnen Diagrammelementen wie Säulen, Balken oder Kreissegmenten angewandt.

20.1 Erzeugen

Nahezu alle Grafikobjekte sind über die entsprechenden Symbole der Zeichnen-Leiste erreichbar (die Kategorien *Zeichnen* und *Autoformen* des *Anpassen*-Dialogfelds enthält darüber hinaus weitere Symbole zum Umgang mit Grafikobjekten) (Abbildung 20.1).

Um ein neues Grafikobjekt zu erzeugen, blenden Sie diese Leiste ein und klicken auf das betreffende Symbol, zum Beispiel auf das Textfeld.



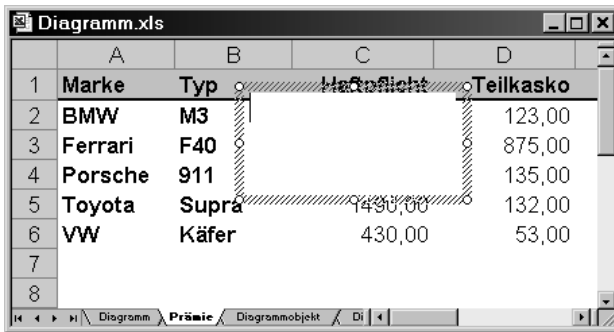
Abbildung 20.1:
Die Zeichnen-Leiste



*Position und
Größe*

Daraufhin wird der Mauscursor zu einem Fadenkreuz. Damit klicken Sie auf jene Stelle in der Tabelle, an der sich eine Ecke des Objekts befinden soll, zum Beispiel die linke obere Ecke. Sie halten die linke Maustaste weiter gedrückt und ziehen das Objekt in die Länge bzw. in die Breite. Ist die gewünschte Größe erreicht, lassen Sie die Maustaste los und das Objekt erscheint (Abbildung 20.2).

Abbildung 20.2:
Grafikobjekt
einfügen



Der blinkende Cursor im Textfeld fordert Sie auf, Text einzugeben. Geben Sie Text als Fließtext ein. Excel kümmert sich automatisch um den Zeilenumbruch.

Durch das Einfügen von Bindestrichen können Sie jederzeit nachträglich Worttrennungen festlegen. Excel ändert den Zeilenumbruch entsprechend.

 ermöglicht Ihnen das Einfügen von Absätzen.

Übrigens: Momentan ist das Grafikobjekt aktiviert. Wollen Sie wieder die Tabelle aktivieren, klicken Sie einfach irgendeine Zelle an.

Das Erzeugen anderer Grafikobjekte läuft analog ab (auf zugehöriges Symbol klicken, Fadenkreuz positionieren und bei weiter gedrückter Maustaste die Objektgröße/-form festlegen).

All diese Objekte werden wie beschrieben in die Tabelle eingefügt, mit wenigen Ausnahmen wie zum Beispiel den folgenden:



Beim Erzeugen von Pfeilen sollten Sie beachten, dass sich das Pfeilende anschließend dort befindet, wo Sie bei der Größendefinition die Maustaste loslassen.

Nach dem Anklicken des *Freihandformen*-Symbols lassen Sie die Maustaste los, bewegen den Mauscursor zu jener Position, an der sich die erste Ecke befinden soll und legen diese Ecke mit einem Mausklick fest.



Danach lassen Sie die Maustaste wieder los, bewegen die Maus zur zweiten gewünschten Ecke (wobei Sie eine Linie vom zuletzt festgelegten Eckpunkt hinter sich herziehen), verankern auch diese mit einem Mausklick und so weiter.

Jeder neu festgelegte Eckpunkt wird mit dem zuvor festgelegten durch eine Linie verbunden. Zusätzlich können Sie jedoch bei gedrückter Maustaste Freihandlinien zeichnen.

Nach der Verankerung des letzten Eckpunkts beenden Sie die Erstellung des Objekts per Doppelklick.

Bei vielen Symbolen, die leicht verwackelt gezeichnet werden können, ist es sehr nützlich, bei der Größen- und Formfestlegung die -Taste gedrückt zu halten. Damit wird das Verwackeln vermieden.

*Verwackeln
vermeiden*

Beispielsweise können Sie mit dem Liniensymbol nur noch waagerechte, senkrechte oder Linien mit einem 45-Grad-Winkel zeichnen und mit dem *Kreisbogen*-Symbol nur echte Kreisausschnitte und keine krummen Ausschnitte von Ellipsen. Das *Kreis*-Symbol ermöglicht dann analog dazu ebenfalls nicht mehr das Zeichnen von Ellipsen, sondern nur noch das Zeichnen wirklich runder Kreise.

20.2 Selektieren und manipulieren

Um ein Grafikobjekt zu verschieben, bewegen Sie den Mauscursor über das Objekt, bis er vier Richtungspfeile mit sich herumschleppt. Drücken Sie nun die linke Maustaste und halten Sie sie gedrückt, so können Sie das Objekt durch Ziehen oder Schieben der Maus in die betreffende Richtung verschieben.

Für alle anderen Manipulationen müssen Sie das Objekt zunächst durch Anklicken selektieren. Um das Objekt herum erscheinen kleine Rechtecke, die Anfasser (Abbildung 20.3).

Um die Objektgröße zu verändern, lassen Sie die Maustaste nach dem Selektieren (=Anklicken) los und bewegen den Cursor zu einem der Anfasser am Objektrand. Dort wird er zu einem Doppelpfeil, der die Richtungen anzeigt, in denen die Objektausdehnung nun manipuliert werden kann.

*Verschieben/
Größe ändern*

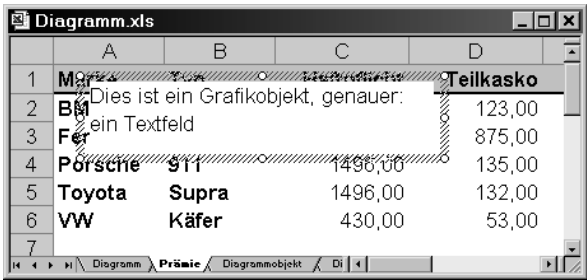
Abbildung 20.3:
Grafikobjekt
selektieren



	A	B	C	D
1	Marke	Typ	Preis	Teilkasko
2	BMW	M3	123,00	123,00
3	Ferrari	F40	875,00	875,00
4	Porsche	911	135,00	135,00
5	Toyota	Supra	132,00	132,00
6	VW	Käfer	430,00	53,00
7				

Drücken Sie nun die Maustaste und halten Sie sie gedrückt, so können Sie das Objekt (in den durch den Doppelpfeil angezeigten Richtungen) beliebig in die Länge oder Breite ziehen bzw. schieben und auf diese Weise seine Form und Größe verändern (Abbildung 20.4).

Abbildung 20.4:
Form und Größe
verändern



	A	B	C	D
1	Marke	Typ	Preis	Teilkasko
2	BMW	M3	123,00	123,00
3	Ferrari	F40	875,00	875,00
4	Porsche	911	135,00	135,00
5	Toyota	Supra	132,00	132,00
6	VW	Käfer	430,00	53,00
7				

Ziehen Sie an einem der vier Eckanfasser und halten Sie bei diesen Manipulationen die -Taste gedrückt, so bleibt das Breite-Höhe-Verhältnis und damit die Objektform erhalten: Jede Vergrößerung/Verkleinerung wirkt sich auf die Breite *und* die Höhe des Objekts im gleichen Maß aus, sodass die Objektform erhalten bleibt.

Halten Sie statt der -Taste die -Taste gedrückt, erfolgen alle Veränderungen zellweise: Das Objekt wird nicht mehr in winzigen Schritten verschoben, sondern springt von Zellrand zu Zellrand, was die Erstellung von Grafiken ermöglicht, in denen Objekte exakt neben- oder untereinander ausgerichtet sein müssen.

**Mehrfach-
selektion**

Sie können mehrere Objekte gleichzeitig manipulieren, indem Sie sie nacheinander anklicken und dabei jeweils die -Taste gedrückt halten. Manipulieren Sie anschließend *irgendeines* dieser Objekte, so wirken sich Ihre Manipulationen auf *alle* *selektierten* Objekte *gleichzeitig* aus.



Die nachfolgend beschriebenen Aktionen leiten Sie entweder mit dem jeweils abgebildeten Symbol ein oder aber mit den entsprechenden Unterbefehlen des Befehls ZEICHNEN der gleichnamigen Symbolleiste.

Sind viele Objekte zu selektieren, können Sie das vereinfachen, indem Sie das abgebildete Symbol anklicken und danach einen Rahmen um alle interessierenden Objekte legen, die nach dem Loslassen alle selektiert sind.



Sind mehrere Objekte selektiert, wird diese Selektion wieder aufgehoben, wenn Sie auf irgendeinen Punkt außerhalb der Objekte klicken, beispielsweise auf eine Zelle.

Sollen jedoch mehrere Objekte dauerhaft zu einem übergeordneten Objekt zusammengefasst werden, zu einer Gruppe, so legen Sie wie erläutert einen Rahmen darum herum und klicken anschließend auf das abgebildete Gruppierungs-Symbol (Kategorie *Zeichnen* des *Anpassen*-Dialogfeldes) oder wählen im Kontextmenü GRUPPIERUNG|GRUPPIERUNG.



Wie zuvor können Sie die gruppierten Objekte nun gemeinsam manipulieren und wie ein einziges Objekt behandeln. Die Gruppierung wird erst aufgehoben, wenn Sie auf das *Gruppierung aufheben*-Symbol klicken oder im Kontextmenü GRUPPIERUNG|GRUPPIERUNG AUFHEBEN wählen.



Grafikobjekte können sich überlappen (Abbildung 20.5).

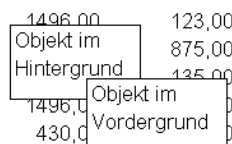


Abbildung 20.5:
Objekt-Überlappung

Soll das teilweise verdeckte Objekt in den Vordergrund kommen, selektieren Sie es und klicken auf das abgebildete Symbol (Kategorie *Zeichnen* des *Anpassen*-Dialogfeldes) oder wählen im Kontextmenü REIHENFOLGE|IN DEN VORDERGRUND.



Umgekehrt können Sie mit diesem Symbol bzw. dem Befehl REIHENFOLGE|IN DEN HINTERGRUND des Kontextmenüs ein Objekt, das sich im Vordergrund befindet, nach der Selektion in den Hintergrund verbannen.



Selektierte Grafikobjekte können Sie wie gewohnt mit dem BEARBEITEN-Menü, den Symbolleisten oder den Kurztasten editieren: LÖSCHEN|ALLES löscht es, KOPIEREN befördert es in die Zwischenablage, AUSSCHNEIDEN entfernt es zusätzlich an der aktuellen Position und EINFÜGEN fügt ein in der Zwischenablage enthaltenes Objekt an der aktuellen Cursorposition ein.

Editieren

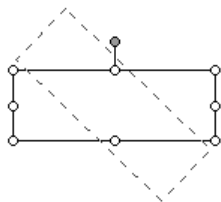
Nicht nur die speziellen Textfelder, sondern nahezu alle Objekte können Texte enthalten (mit Ausnahme einiger weniger wie Pfeile oder Linien). Um Text einzugeben, klicken Sie das betreffende Objekt einfach an und tippen drauf los oder – wenn es bereits Text enthält – bewegen den Cursor zuvor zu jener Textstelle, die Sie editieren wollen.

Texte

Sie können Textteile wie gewohnt markieren (Maus darüber ziehen) und danach mit den bekannten Symbolen »F«, »K« etc. wie gewohnt fett oder kursiv formatieren oder die Textgröße und/oder die Textausrichtung verändern.

Drehen Durch Ziehen am grünen »Henkel« oberhalb eines aktivierten Objekts können Sie es drehen (Abbildung 20.6).

Abbildung 20.6:
Objekt drehen

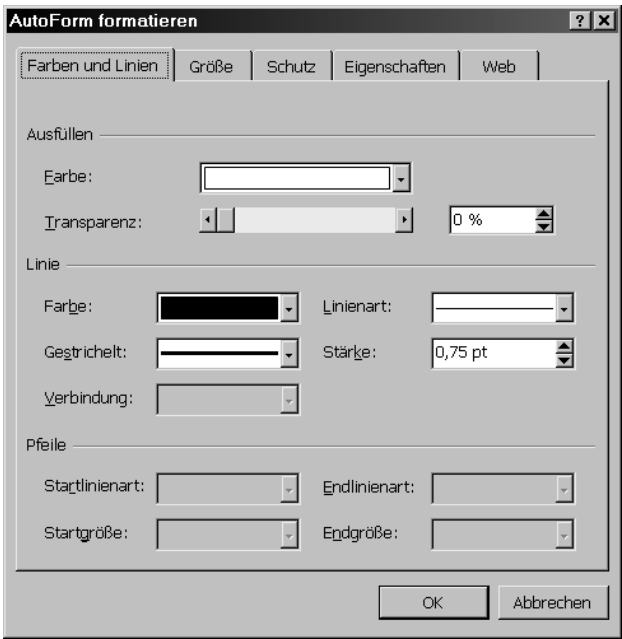


20.3 Formatieren

Erheblich feinere Möglichkeiten zur Objektmanipulation bieten die objektbezogenen Dialogfelder.

Sie öffnen sie, indem Sie sich einem der Objektränder nähern, bis der Mauscursor vier Richtungspfeile mitschleppt und indem Sie das Objekt nicht wie zuvor durch einen einfachen Mausklick, sondern durch einen Doppelklick selektieren (Abbildung 20.7).

Abbildung 20.7:
Objektbezogenes
Dialogfeld, Register
»Farben und Linien«



Der Doppelklick besitzt die gleiche Wirkung wie die einfache Selektion und nachfolgende Wahl von `FORMAT[[OBJEKTTYPE]...` (lautet z.B. `FORMAT[AUTOFORM]...`, wenn ein `AutoForm`-Objekt selektiert ist) bzw. des Befehls `[OBJEKTTYPE] FORMATIEREN...` (z.B. `AUTOFORM FORMATIEREN...`) im `OBJEKT`-Kontextmenü.

Alle in den Dialogfeldern enthaltenen Manipulationsmöglichkeiten können Sie auf mehrere Objekte gleichzeitig anwenden, wenn Sie sie wie erläutert gemeinsam selektieren oder zu einem übergeordneten Objekt gruppieren.

Welche Optionen in diesen Dialogfeldern verfügbar sind, hängt vom selektierten Objekt ab: Pfeile können Sie auf andere Art und Weise manipulieren als Kreise. Textfelder bieten wiederum andere Möglichkeiten. Ich beschränke mich daher im Folgenden auf die Erläuterung der weitest verbreiteten Optionen und Manipulationsmöglichkeiten, die auf nahezu alle Grafikobjekte angewendet werden können.



Im Register *Farben und Linien* beeinflussen Sie die Darstellung des Objekttinneren und des Objektrahmens. Der unterste Abschnitt, *Pfeile*, ist ausschließlich für das Formatieren von Pfeilen gedacht, um damit die Pfeileinzelteile individuell zu formatieren.

**Register Farben
und Linien**

Der Abschnitt *Ausfüllen* ist immer für das Objekttinnere zuständig: Ziehen Sie den Schieberegler *Transparenz* nach rechts, so wird das Grafikobjekt durchscheinender und der Tabellenausschnitt darunter ist zu erkennen.

Im *Farb*-Listefeld wählen Sie zunächst die Farbe des Objekttinneren. (»Keine Füllung« bedeutet, dass das Objekttinnere farblos und der Tabellenhintergrund vollständig sichtbar ist; »Automatisch« ist die Standardfarbe, die durch die Windows-Systemsteuerung festgelegt wird.)

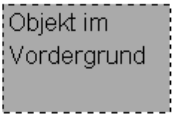
Zusätzlich enthält das Listefeld die Option »Weitere Farben...«, die eine noch detailliertere Farbauswahl ermöglicht, und die Option »Fülleffekte...«, die ein weiteres Dialogfeld öffnet, das Ihnen ermöglicht, die ausgewählte Farbe als Farbverlauf zu verwenden oder Muster darzustellen.

Im Abschnitt *Linienart* legen Sie die Eigenschaften der Linie fest, die den Objektrahmen bildet: Im Listefeld *Farbe* wählen Sie die Rahmenfarbe aus. (»Keine Linie« entfernt den Rahmen und »Automatisch« steht für die Standardfarbe, die durch die Windows-Systemsteuerung festgelegt wird.)

Im Listefeld *Art* wählen Sie den Linientyp (dünne oder dicke Linie, einfache oder doppelte Rahmenlinie etc.). Darunter, im Drehfeld *Stärke*, können Sie die Linienstärke der ausgewählten Linie noch weitergehend beeinflussen.

Im Listefeld *Gestrichelt* bestimmen Sie, ob die Rahmenlinie durchgehend oder – und wenn ja, wie – gestrichelt dargestellt werden soll (Abbildung 20.8).

Abbildung 20.8:
Formatierung des
Rahmens und des
Objektinneren



Register *Schrift* Das Register *Schrift* bestimmt die Darstellung von Text im Objektinneren. In genau der gleichen Weise wie bei Tabellen wählen Sie darin die Schriftart und -größe und verschiedene Schriftattribute aus.

Register *Ausrichtung* Das Register *Ausrichtung* bietet ebenfalls nichts Neues. Es ermöglicht wie bei Tabellentexten die linksbündige, zentrierte, rechtsbündige oder bündige Textausrichtung; und zwar in beiden Richtungen, horizontal oder vertikal.

Aktivieren Sie *Automatische Größe*, passt Excel die Objektgröße nach dem Schließen des Dialogfelds an den darin enthaltenen Text an. Sie ersparen es sich dadurch, die Objektgröße nach Textformatierungen selbst anpassen zu müssen, beispielsweise, weil Sie den Schriftgrad erhöhten und der Text anschließend nicht mehr vollständig in das Textfeld hineinpasst.

Register *Abstände* Das *Abstände*-Register ist trivial (Abbildung 20.9).

Abbildung 20.9:
Register
»Abstände«



In den vier Drehfeldern legen Sie den Abstand des im Grafikobjekt enthaltenen Texts zu den vier Objekträndern fest. Aktivieren Sie *Automatisch*, so werden die Standardrandabstände wiederhergestellt.

Im Register *Größe* bestimmen Sie die Größe des Grafikobjekts (Abbildung 20.10).

Register *Größe*

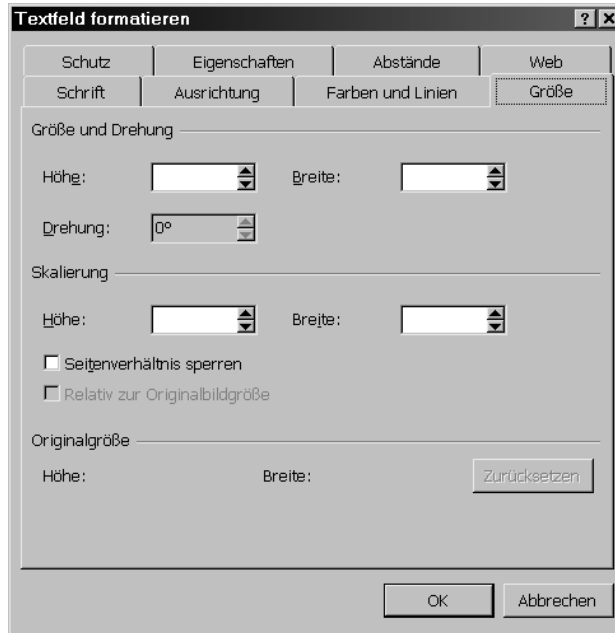


Abbildung 20.10:
Register »Größe«

Entweder legen Sie die Größe in Zentimetern fest oder aber in Prozentwerten, wobei Letzteres sehr einfach ermöglicht, das Objekt beispielsweise doppelt so hoch wie breit zu machen.

Aktivieren Sie *Seitenverhältnis sperren*, wird bei jeder Änderung der Breite die Höhe und bei jeder Änderung der Höhe die Breite entsprechend angepasst und somit die Objektform beibehalten.

Drehung rotiert das Objekt im Uhrzeigersinn. »90 Grad« bedeutet entsprechend eine Vierteldrehung nach rechts.

Das *Schutz*-Register ist trivial und enthält nur zwei Kontrollkästchen (Abbildung 20.11).

Register *Schutz*

- ➔ *Gesperrt* sperrt das Objekt.
- ➔ *Text gesperrt* sperrt zusätzlich den darin enthaltenen Text.

Beide Optionen entsprechen den in Kapitel 38.5, »Dokumente schützen«, besprochenen Optionen zum Schützen von Zellen und Formeln und sind nur bei aktiviertem Dokumentschutz wirksam.



Abbildung 20.11:
Register »Schutz«



Register *Eigen-
schaften*

Das Register *Eigenschaften* ist bereits interessanter (Abbildung 20.12).

Abbildung 20.12:
Register
»Eigenschaften«



Grafikobjekte sind nicht ganz so unabhängig von der darunter liegenden Tabelle, wie es auf den ersten Blick erscheint. Tatsächlich sind sie an die Größe und die Position der Zellen, die sie verdecken, gebunden. Diese Bindung können Sie hier beeinflussen.

Nur von Zellposition abhängig bindet das Objekt an die Position der darunter liegenden Zellen. Verschieben Sie diese Zellen, wird das darüber liegende Objekt mitverschoben.

Bei *Von Zellposition und -größe abhängig* ist das Objekt zusätzlich auch von der Größe der darunter liegenden Zellen abhängig: Verbreitern Sie die betreffenden Spalten, wird das Objekt ebenfalls breiter; erhöhen Sie die Zeilenhöhe, wird auch das Objekt entsprechend höher.

Mit *Von Zellposition und -größe unabhängig* ist das Objekt weder von der Größe noch der Position der Zellen darunter abhängig und wird durch Manipulationen dieser Zellen nicht beeinflusst.

Das Kontrollkästchen *Objekt drucken* legt fest, ob das Objekt beim Ausdruck der Tabelle ebenfalls ausgedruckt oder aber einfach ignoriert wird. Außer bei Schaltflächen ist dieses Kästchen standardmäßig immer aktiviert.

Das Register *Web* ist nützlich, wenn Sie eine Excel-Tabelle als Webseite veröffentlichen (Abbildung 20.13).

Register Web



Abbildung 20.13:
Register »Web«

Der hier eingegebene Text erscheint während des Ladens der Webseite, also bevor das Grafikobjekt selbst dargestellt wird, stellvertretend an seiner Stelle.

21 Bildobjekte aller Art

Die verschiedenen Unterbefehle von EINFÜGEN|GRAFIK ermöglichen Ihnen, unterschiedlichste Objekte in Ihre Tabelle einzufügen, wobei VON SCANNER ODER KAMERA... voraussetzt, dass ein Scanner vorhanden und entsprechende TWAIN-kompatible Treiber installiert sind und ermöglicht dann das Scannen und Einbinden von Vorlagen in Excel-Tabellen. Das zugehörige Dialogfeld kann ich nicht erläutern, da sein Inhalt von Ihrem Scanner-treiber abhängt.

21.1 Grafikdateien

EINFÜGEN|GRAFIK|AUS DATEI... öffnet (ebenso wie das entsprechende Symbol der Zeichnen-Leiste) ein *Dateiselektions*-Dialogfeld, in dem Sie eine Grafikdatei auswählen und in die Tabelle einfügen können (Abbildung 21.1).

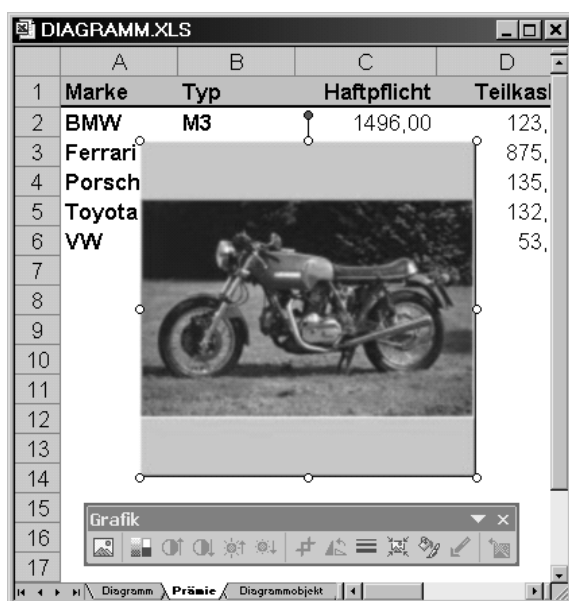


Abbildung 21.1:
Grafik einfügen

Ist das eingefügte Bild selektiert, können Sie es am Henkel drehen, es verkleinern, vergrößern etc. Darüber hinaus ermöglicht Ihnen jedoch die automatisch eingeblendete *Grafik-Symbolleiste* noch weitere Bildmanipulationen.

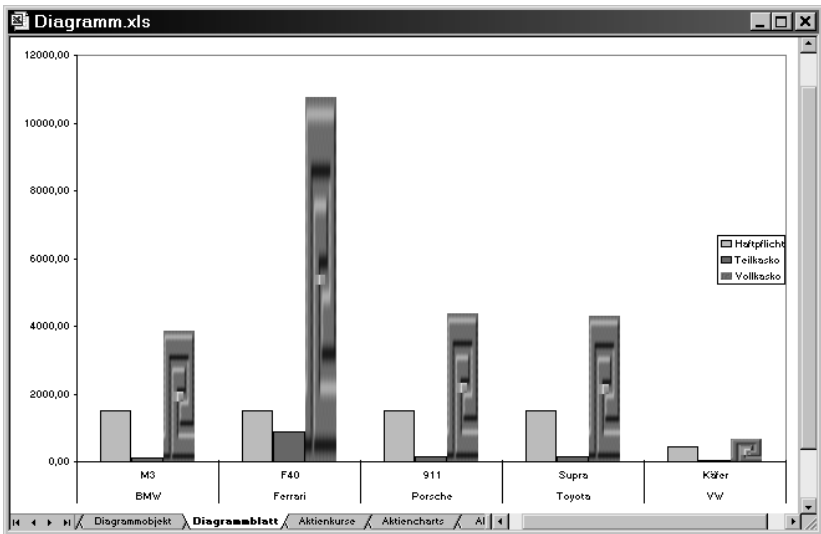


Mit den entsprechenden Symbolen können Sie die Farbanzahl verändern, es beispielsweise auf Graustufen- oder Schwarzweiß-Darstellung reduzieren, den Kontrast und die Helligkeit verändern, es zuschneiden, es komprimieren (siehe Kapitel 30.2, »Bilder komprimieren«) und eine Farbe auswählen, die transparent sein soll (oft nützlich, um grafische Bilder mit einfarbigen Bildrändern an einen farbigen Hintergrund anzupassen).



Sie können eine Grafik wie gezeigt als separates Objekt einfügen. Sie können eine externe Grafikdatei jedoch auch im Zusammenhang mit Diagrammen benutzen und in Datenpunktbilder integrieren (siehe Kapitel 22.2.1, »Diagramm erzeugen«) (Abbildung 21.2).

Abbildung 21.2:
In Datenpunktbilder
eingefügte Grafik



Für diese Abbildung selektierte ich die dritte Datenreihe und fügte danach mit EINFÜGEN|GRAFIK|AUS DATEI... eine der im Windows-Verzeichnis enthaltenen Grafikdateien ein. Statt als separates Objekt wurde die Grafik als Muster für die selektierte Datenreihe verwendet.



Ist gerade ein Grafikprogramm aktiv, können Sie alternativ dazu darin eine Grafik oder einen Ausschnitt einer Grafik selektieren und mit BEARBEITEN|KOPIEREN in die Zwischenablage befördern. Anschließend selektieren Sie in Excel eine Diagramm-Datenreihe und fügen die in der Zwischenablage enthaltene Grafik mit BEARBEITEN|EINFÜGEN als Muster für diese Datenreihe ein.

21.2 WordArt

EINFÜGEN|GRAFIK|WORDART... öffnet ein Dialogfeld mit verschiedenen mehr oder weniger kunstvollen Schriftvarianten (Abbildung 21.3).

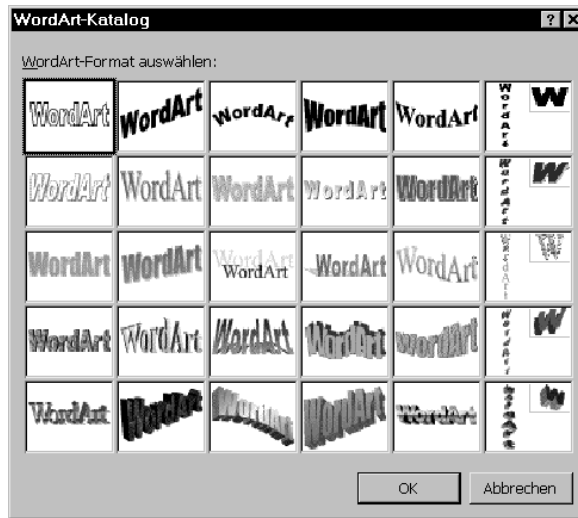


Abbildung 21.3:
WordArt-Katalog

Nach Auswahl der gewünschten Variante erscheint ein weiteres Dialogfeld, in dem Sie die Schriftart/-größe und -attribute selektieren und den gewünschten Text eingeben. Nach dem Schließen auch dieses Dialogfelds wird der Text als Grafikobjekt in die Tabelle eingebettet und die WordArt-Symboleiste eingeblendet.

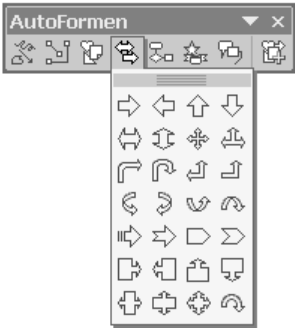
Deren Symbole ermöglichen Ihnen nicht nur, die verwendete Variante oder den Text nachträglich zu ändern, sondern das Grafikobjekt nach allen Regeln der Kunst zu manipulieren: es zu rotieren, den Schriftverlauf zu ändern etc.

21.3 AutoFormen

Das Symbol *AutoFormen* der Symbolleiste öffnet eine Liste mit verschiedenen AutoFormen. Ziehen Sie am oberen Rand dieser Liste, »verwandelt« sie sich in die *AutoFormen*-Symbolleiste, die Sie mit dem Befehl EINFÜGEN|GRAFIK|AUTOFORMEN auch direkt einblenden können (Abbildung 21.4).

Sowohl in der Liste als auch in der eigenständigen Symbolleiste entspricht jedes Symbol einem Menü, das nach dem Anklicken die AutoFormen der betreffenden Kategorie anzeigt. Sie können eines davon anklicken und wie gewohnt in das Blatt einfügen.

Abbildung 21.4:
AutoFormen



Genügt Ihnen die Auswahl nicht, klicken Sie das abgebildete Symbol an, das den Aufgabenbereich einblendet, der Ihnen weitere AutoFormen zur Verfügung stellt (Abbildung 21.5).

Abbildung 21.5:
Weitere
AutoFormen



Mit dem Symbol ganz oben können Sie das AutoFormen-Fenster erweitern bzw. wieder verkleinern.

Per Anklicken fügen Sie eine AutoForm in die Tabelle ein.

Weitere Möglichkeiten und vor allem eine weitaus umfangreichere Auswahl bieten die im folgenden Kapitel besprochenen ClipArts.

21.4 ClipArt

EINFÜGEN|GRAFIK|CLIPART... ermöglicht das Einfügen unterschiedlichster Objekte: ClipArt, optisch meist ansprechendere Grafikobjekte als die doch recht einfachen AutoFormen, aber auch Fotos, Klänge oder Videos.

Wegen der Unmenge verfügbarer Objekte erscheint zunächst der Aufgabenbereich (klicken Sie im zuvor erscheinenden Dialogfeld auf *Später*), um Ihnen eine gezieltere Suche zu ermöglichen (Abbildung 21.6).

Suchen

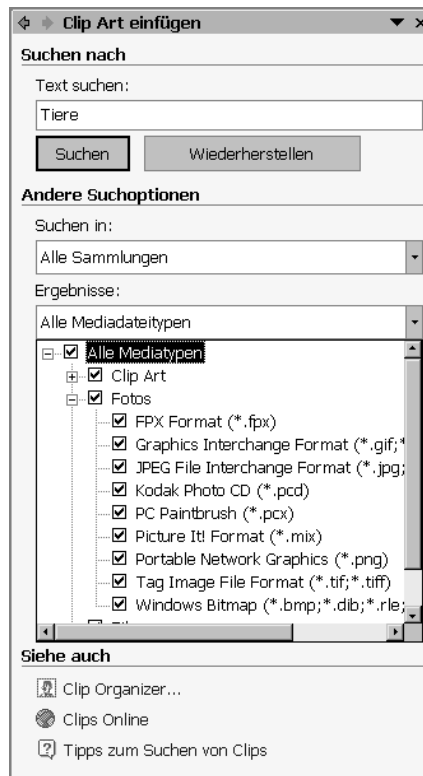


Abbildung 21.6:
ClipArt-Galerie

Sie können einen Suchtext wie »Tiere« eingeben.

Zusätzlich zu einem Suchtext oder alternativ dazu können Sie die Suche mit den beiden Listenfeldern eingrenzen. Im oberen Listenfeld können Sie angeben, wo nach Objekten gesucht werden soll, auf die Ihre Beschreibung zutrifft. Am interessantesten ist *Office-Sammlungen*, worunter die mit Office ausgelieferten Medienobjekte zu verstehen sind, die in Kategorien wie »Tiere« und »Gebäude« unterteilt sind, aus der Sie die gewünschte auswählen.

Zusätzlich können Sie im unteren Listenfeld entsprechend der Abbildung die Suche auf bestimmte Medienarten eingrenzen, um beispielsweise nur Fotos zu finden.

Klicken Sie auf *Suchen*, werden die gefundenen Objekte im Aufgabenbereich angezeigt (Abbildung 21.7).

Abbildung 21.7:
Kategorie »Tiere«



Per Anklicken fügen Sie ein Objekt in die Tabelle ein.

Klicken Sie den Listenpfeil am rechten Rand eines Objekts an, können Sie es mit dem zugehörigen Kontextmenü aus dem *Clip Organizer* löschen, seine beim Suchen verwendeten Schlüsselwörter editieren, es in eine bestimmte Kategorie kopieren etc.

Ändern öffnet wieder das Suchregister und ermöglicht Ihnen damit, die Suche mit geänderten Kriterien erneut durchzuführen.

Klicken Sie im Aufgabenbereich auf *Clips Online*, wird Ihr Webbrowser gestartet und Sie können weitere Objekte von einer Microsoft-Webseite verwenden.

Klicken Sie im Aufgabenbereich auf *Clip Organizer*, so erscheint ein eigenes Fenster, in dem Sie die verfügbaren Objekte organisieren können. Der Ausdruck »Fenster« ist jedoch untertrieben, da es sich praktisch um ein eigenständiges Programm handelt, dessen Möglichkeiten im Rahmen dieses Excel-Buchs leider nicht erläutert werden können.

Organisieren

22 Diagramme erzeugen

Im folgenden Kapitel geht es um die grafische Darstellung Ihrer Daten. Am einfachsten ist die Erzeugung von Diagrammen mit Hilfe des Diagramm-Assistenten. Er ermöglicht Ihnen, Diagramme nahezu auf Knopfdruck zu erstellen und benötigt dazu nur wenige Angaben, vor allem natürlich die Angabe des darzustellenden Datenbereichs.

Ich erläutere zunächst den Umgang mit den grundsätzlichen Optionen des Assistenten; am wichtigsten ist zweifellos die Auswahl des gewünschten Diagrammtyps, beispielsweise »Kreis« oder »Säule«, um entsprechend Kreis- bzw. Säulendiagramme zu erzeugen.

Danach zeige ich Ihnen, wie Sie – ebenfalls mit Hilfe des Assistenten – ein bereits vorhandenes Diagramm nachträglich verändern, ohne es neu erstellen zu müssen. Diese Technik ist äußerst nützlich, um beispielsweise zuvor vergessene Beschriftungen wie eine Legende einzufügen oder gar einen völlig anderen Diagrammtyp zu verwenden.

Anschließend gehe ich auf die interne Arbeitsweise des Assistenten ein, auf seine Interpretation von Datenreihen und Rubriken.

Das ist notwendig, da der Assistent leider nicht perfekt ist und Ihre Angaben zum darzustellenden Datenbereich ohne weiteres falsch verstehen kann. Das Resultat ist ein Diagramm, das recht wenig mit der eigentlich gewünschten Darstellung zu tun hat. Entsprechend erläutere ich, was zu tun ist, wenn sich der Assistent irrt und wie Sie ihn dazu bringen können, doch noch das gewünschte Diagramm zu erzeugen.

Wie ich danach zeige, können Diagramme als Grafikobjekte in eine Tabelle eingefügt oder auf eigenen Diagrammblättern dargestellt werden – deren Inhalt Sie bei Bedarf jederzeit nachträglich als Grafikobjekt in eine Tabelle einfügen können.

Zum Abschluss erläutere ich, welche Elemente Sie über die Standardelemente hinaus in Diagramme einfügen können. Dabei gehe ich nicht nur auf die Möglichkeiten des Diagramm-Assistenten ein, mit dessen Hilfe Sie beispielsweise Achsen, Achsenbeschriftungen, Gitternetzlinien und Legenden

einfügen können. Zusätzlich zeige ich Ihnen, welche Elemente darüber hinaus verfügbar sind, die der Assistent zwar nicht kennt, die Sie aber manuell einfügen können (beliebige Texte, Trennlinien etc.).

22.1 Diagrammelemente

Excel bietet vielfältige Möglichkeiten, um individuell gestaltete Diagramme zu erzeugen. Die folgende Abbildung enthält eine Tabelle mit verschiedenen Versicherungsprämien, aufgesplittet nach Autotypen und Versicherungsarten (Abbildung 22.1).

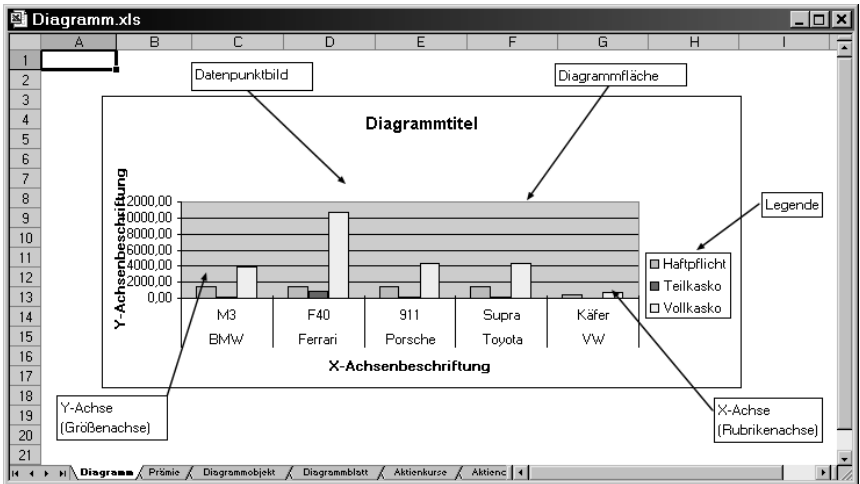
Abbildung 22.1:
Diagramm.xls,
»Prämie«

	A	B	C	D	E
1	Marke	Typ	Haftpflicht	Teilkasko	Vollkasko
2	BMW	M3	1496,00	123,00	3864,00
3	Ferrari	F40	1496,00	875,00	10746,00
4	Porsche	911	1496,00	135,00	4374,00
5	Toyota	Supra	1496,00	132,00	4284,00
6	VW	Käfer	430,00	53,00	653,00

Diagramm-
elemente

Jeder Wert, beispielsweise die Haftpflichtprämie für einen Ferrari F40, ist ein Datenpunkt, dem in einem Diagramm ein Datenpunktbild entspricht, zum Beispiel eine Säule oder ein Kreissegment (Abbildung 22.2).

Abbildung 22.2:
Diagramm.xls,
»Diagramm«



Eine *Wertereihe* (mehrere Haftpflichtprämien) ist eine Datenpunktreihe oder kürzer: eine Datenreihe.

Ein Diagramm besteht aus wenigstens einer Datenreihe. Das Beispiel enthält sogar drei Datenreihen, »Haftpflicht«, »Teilkasko« und »Vollkasko«. Entsprechend enthält das Diagramm drei Säulenreihen, die sich durch unterschiedliche Farben oder Graustufen voneinander unterscheiden.

Die Zeichenfläche wird unten und oben von der X-Achse, der so genannten Rubrikenachse begrenzt, und links und rechts von der Y-Achse, der Größenachse.

Beide Achsen enthalten Achsentitel, die X-Achse die Namen der dargestellten Rubriken, die Y-Achse eine Zahleneinteilung, die Aufschluss über die Höhe der dargestellten Werte gibt.

Zusätzlich kann ein Diagramm eine Legende enthalten, die Auskunft über die darin dargestellten Datenreihen gibt, und einen Diagrammtitel, eine Überschrift.

22.2 Der Diagramm-Assistent

Mit Hilfe des Diagramm-Assistenten erzeugen Sie ein Diagramm oder verändern ein bereits vorhandenes Diagramm nachträglich. Um ein neues Diagramm zu erzeugen, selektieren Sie irgendeine Zelle im darzustellenden Tabellenbereich und wählen EINFÜGEN|DIAGRAMM.... oder klicken auf das abgebildete Symbol (Abbildung 22.3).

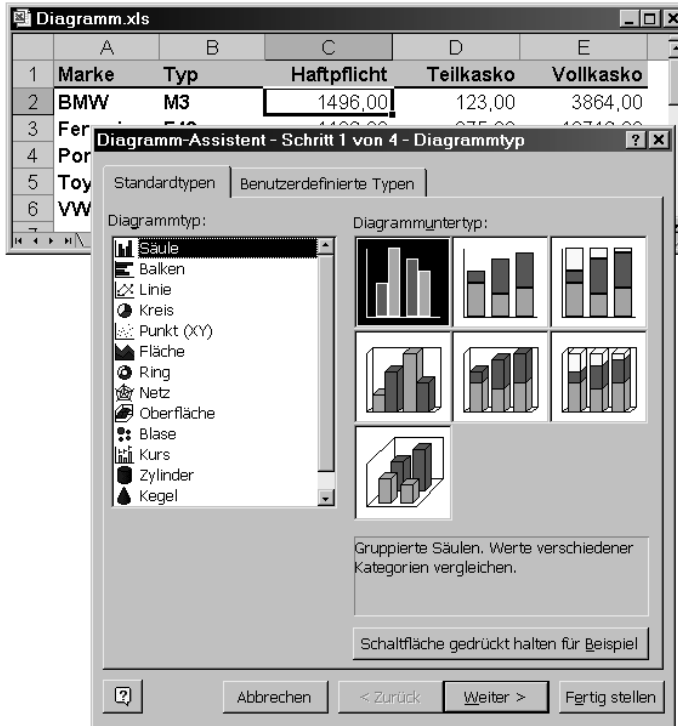


Die Diagrammerzeugung erfordert mehrere Schritte, die Sie mit den Schaltflächen am unteren Rand des Dialogfelds steuern:

- ➔ *Abbrechen* bricht den gesamten Vorgang ab.
- ➔ *< Zurück* bringt Sie zum jeweils vorhergehenden Schritt zurück, beispielsweise von Schritt 3 zu Schritt 2.
- ➔ *Weiter >* schließt den aktuellen Schritt ab und leitet den folgenden Schritt ein.
- ➔ *Fertig stellen* schließt die gesamte Diagrammerzeugung ab und verwendet bei allen eventuell noch nicht ausgeführten Schritten die Standardvorgaben.

Fertig stellen können Sie bereits jetzt wählen, im ersten Schritt. Mit den dadurch verwendeten Standardvorgaben würde ziemlich genau das in Abbildung 22.2 gezeigte Diagramm erzeugt werden. Stattdessen können Sie jedoch auch die einzelnen Schritte der Reihe nach durchlaufen, um bis ins Detail selbst zu beeinflussen, wie das erzeugte Diagramm aussehen soll.

Abbildung 22.3:
Diagramm-Assistent, Schritt 1



Nach Angabe des Datenbereichs können Sie übrigens jederzeit auf die Schaltfläche *Schaltfläche gedrückt halten für Beispiel* klicken, um sich eine Vorschau des zu erzeugenden Diagramms anzeigen zu lassen.

22.2.1 Diagramm erzeugen

*Diagrammtyp/
Diagramm-
variante*

Den größten Einfluss hat der erste Schritt, in dem Sie zunächst den gewünschten Diagrammtyp auswählen, z.B. »Kreis« oder »Linie« (oder die Standardvorgabe »Säule« akzeptieren). Da jeder Typ in mehreren Varianten verfügbar ist, klicken Sie danach auf die von Ihnen gewünschte Diagrammvariante oder akzeptieren erneut die vorselektierte Standardvariante.



Die verschiedenen Typen und Varianten erläutere ich übrigens in Kapitel 23, »Diagrammarten«.

Datenbereich

Im zweiten Schritt müssen Sie in Ihrer Tabelle die Daten selektieren, die als Diagramm dargestellt werden sollen. Excel gibt dabei automatisch den Bereich um die aktive Zelle herum vor, im Beispiel den Bereich A1:E6, also die Zahlen und die zugehörigen Spalten- und Zeilenüberschriften.

Soll wie hier tatsächlich genau dieser Bereich dargestellt werden, so übernehmen Sie die Vorgabe einfach, die Sie jederzeit durch Eintippen eines anderen Bereichsbezugs oder per Daraufzeigen korrigieren können (Abbildung 22.4).

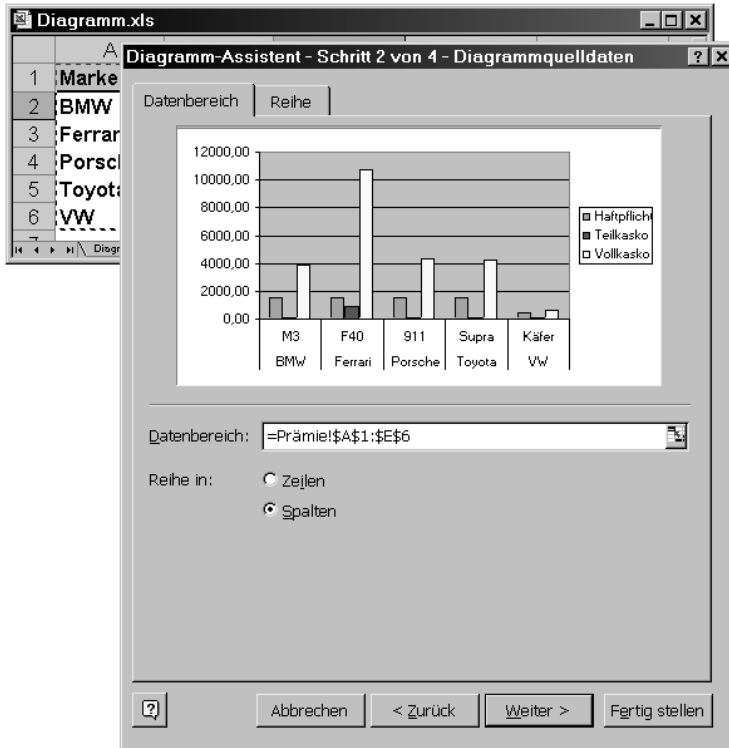


Abbildung 22.4:
Diagramm-
Assistent, Schritt 2

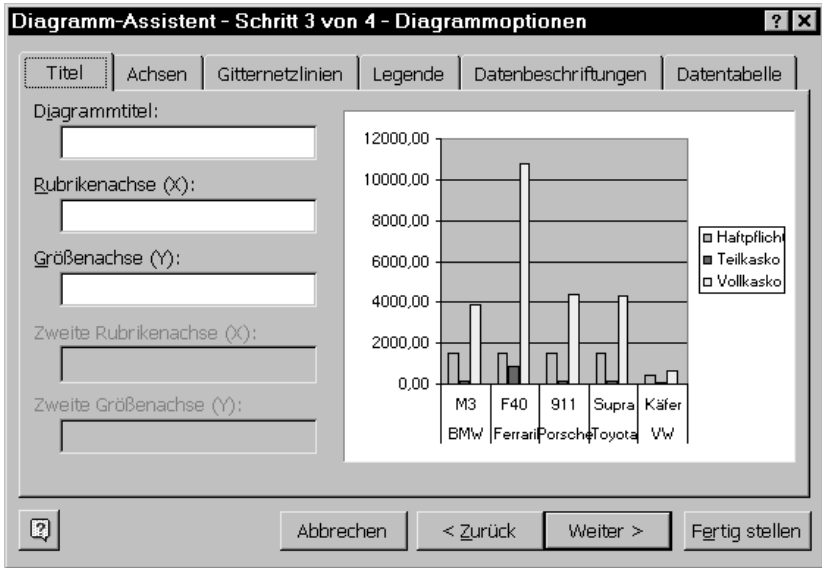
Einige Beispiele für andere Bereichsbezüge: Benötigen Sie im Gegensatz zu Abbildung 22.2 keine Legende, so genügt es, den Bereich A2:E6 ohne die Spaltenüberschriften zu selektieren. Wollen Sie auf die Rubrikenüberschriften verzichten, selektieren Sie nur C1:E6. Verzichten Sie auf beides, selektieren Sie den reinen Zahlenbereich C2:E6.

Ob die Option Zeilen oder aber Spalten aktiviert ist, besitzt enormen Einfluss darauf, wie Excel die im angegebenen Bereich enthaltenen Daten interpretiert. Diese Interpretation und auch das Register Reihe werde ich noch ausführlich erläutern (siehe Kapitel 22.3, »Datenreihen und Rubriken«). Akzeptieren Sie vorläufig die von Excel vorgegebenen Einstellungen, die fast immer korrekt sind.

Im dritten Schritt können Sie verschiedene Diagramm-Beschriftungen definieren, ein Schritt, den ich ebenfalls in einem eigenen Kapitel näher erläutere (siehe Kapitel 22.5, »Zusätzliche Diagrammelemente einfügen« ff.) (Abbildung 22.5).



Abbildung 22.5:
Diagramm-
Assistent, Schritt 3



Der letzte Schritt legt fest, wo das Diagramm erzeugt wird (siehe Kapitel 22.4, »Diagrammblätter«) (Abbildung 22.6).

Abbildung 22.6:
Diagramm-
Assistent, Schritt 4



- ➔ Als neues Blatt mit Angabe eines Namens wie »Diagramm1« erzeugt ein so genanntes Diagrammblatt und verwendet es zur Darstellung des Diagramms.
- ➔ Als Objekt in fügt das Diagramm als Grafikobjekte in das betreffende Blatt ein.

Vorselektiert ist immer *Als Objekt in* unter Angabe des aktuellen Tabellenblatts, in dem sich die darzustellenden Daten befinden. Mit dieser Option erhalten Sie folgendes Resultat (Abbildung 22.7):

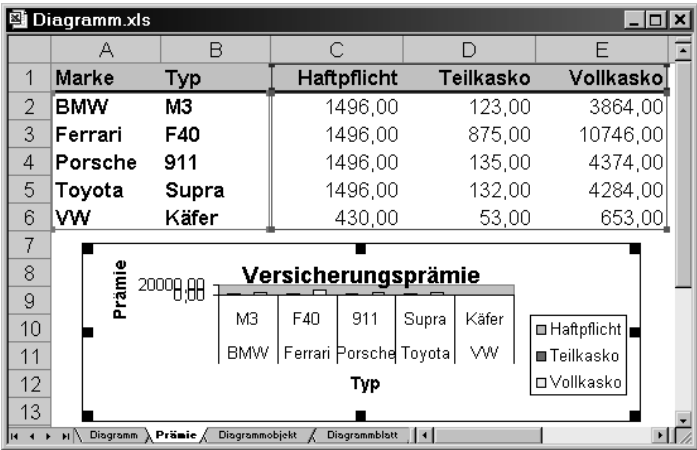


Abbildung 22.7:
Integriertes
Diagramm

22.2.2 Diagramm verändern

Mislungene Schritte der Diagrammerzeugung können Sie auf sehr unterschiedliche Art und Weise nachträglich korrigieren. Am einfachsten ist es meist, auf das Diagramm und danach erneut auf das *Diagramm-Assistenten*-Symbol zu klicken, um diesen Assistenten wieder aufzurufen und die verschiedenen Schritte erneut zu durchlaufen.

Um stattdessen ganz gezielt nur einen bestimmten Schritt zu korrigieren, klicken Sie mit der rechten Maustaste auf die Diagrammfläche, das heißt auf irgendeinen Punkt knapp innerhalb des Diagrammrahmens (Abbildung 22.8).

Kontextmenü

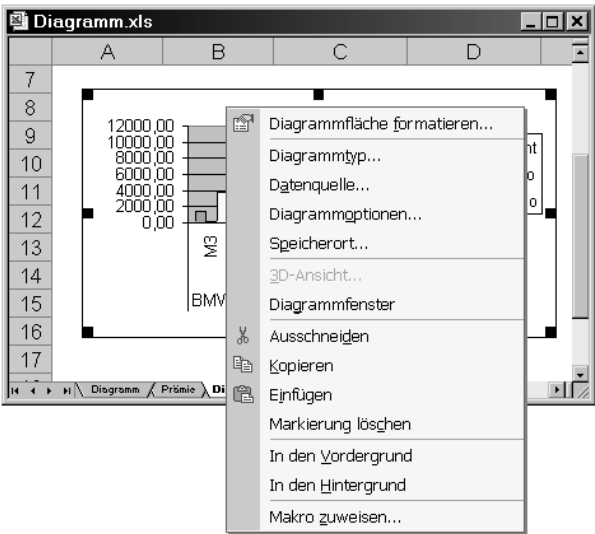


Abbildung 22.8:
Kontextmenü

Die Befehle DIAGRAMMTYP..., DATENQUELLE..., DIAGRAMM-OPTIONEN... und SPEICHERORT... des Kontextmenüs entsprechen den verschiedenen Schritten des Diagramm-Assistenten. DIAGRAMMTYP... aktiviert beispielsweise den ersten Schritt, in dem Sie den Typ des Diagramms und die gewünschte Variante auswählen.

Menü Diagramm

Diese und andere Befehle sind auch über das Menü DIAGRAMM erreichbar, das verfügbar ist, wenn Sie einen beliebigen Teil des Diagramms per Anklicken selektieren.



Zusätzlich können Sie das Diagrammobjekt jederzeit mit den in Kapitel 20.2, »Selektieren und manipulieren«, erläuterten Techniken manipulieren. Die wichtigsten:

- ➔ Anklicken selektiert das Objekt (an den Ecken erscheinen Anfasser).
- ➔ Durch Anklicken und Ziehen eines Diagrammelements können Sie es verschieben: klicken Sie auf die von den beiden Achsen umgebene Zeichnungsfläche selbst, so können Sie diese innerhalb des Diagramms umherschieben, das Gleiche ist mit der Legende möglich; klicken Sie auf einen Punkt, an dem nichts dargestellt ist, so selektieren Sie dadurch das Diagramm selbst und können das gesamte Objekt auf dem Blatt umherschieben.
- ➔ Durch Ziehen an den Anfassern können Sie die Form und/oder Größe des Diagramms verändern.
- ➔ Mit den im Kontextmenü enthaltenen Befehlen können Sie das Diagrammobjekt ausschneiden, kopieren, löschen etc.
- ➔ Mit FORMAT|MARKIERTES [OBJEKT]... (die Bezeichnung [OBJEKT] ist abhängig vom selektierten Objekt) können Sie das momentan selektierte Diagrammelement formatieren.



Das abgebildete Symbol (Kategorie *Diagramm erstellen*) aktiviert eine Art automatisierten Diagramm-Assistenten: Sie selektieren den ausgewählten Tabellenbereich A1:E6 und klicken auf dieses Symbol. Daraufhin wird *ohne jede Nachfrage* ein Diagrammobjekt mit Excels Standardvorgaben so erstellt, als würden Sie im Diagramm-Assistenten jedesmal auf *Weiter >* klicken, ohne die Vorgaben zu verändern.



Passen Ihnen diese Vorgaben nicht, bevorzugen Sie beispielsweise Kreis statt Säulendiagrammen, ändern Sie einfach Excels Standard-Diagrammtyp entsprechend (siehe Kapitel 23.4, »Benutzerdefinierte Diagrammtypen und der Standard-Diagrammtyp«).

Eine ähnliche Funktion besitzt dieses Symbol der Diagramm-Symboleiste, an das eine Liste mit nahezu allen Diagrammarten gebunden ist (siehe Abbildung 23.1). Selektieren Sie einen Datenbereich und klicken Sie auf eines der *Diagrammtyp*-Symbole der Liste, so wird automatisch ein Diagramm im betreffenden Typ erzeugt.



Ein Diagramm ist mit den als Grundlage verwendeten Zahlen dynamisch verbunden. Jede Änderung in der zu Grunde liegenden Tabelle wirkt sich sofort auf das zugehörige Diagramm aus. Erhöhen Sie eine der Zahlen, wird die entsprechende Säule höher.



Die beschriebenen Manipulationen erfordern größtenteils, dass Sie das Diagramm vollständig sehen. Das ist selbst bei kleinem Tabellenfenster möglich, wenn Sie nach Selektion irgendeines Diagrammteils per Anklicken den Befehl ANSICHT|DIAGRAMMFENSTER wählen (auch im Kontextmenü vorhanden, das erscheint, wenn Sie mit der rechten Maustaste auf eine Stelle knapp innerhalb des Diagrammrahmens klicken).

Dieser Befehl öffnet ein eigenes Fenster, in dem das Diagramm erscheint, das *immer* genau die zur vollständigen Darstellung des Diagramms benötigte Größe besitzt. Verkleinern Sie das Fenster, so verkleinert Excel auch das darin enthaltene Diagramm, vergrößern Sie das Fenster, so vergrößert Excel auch das Diagramm (Abbildung 22.9).

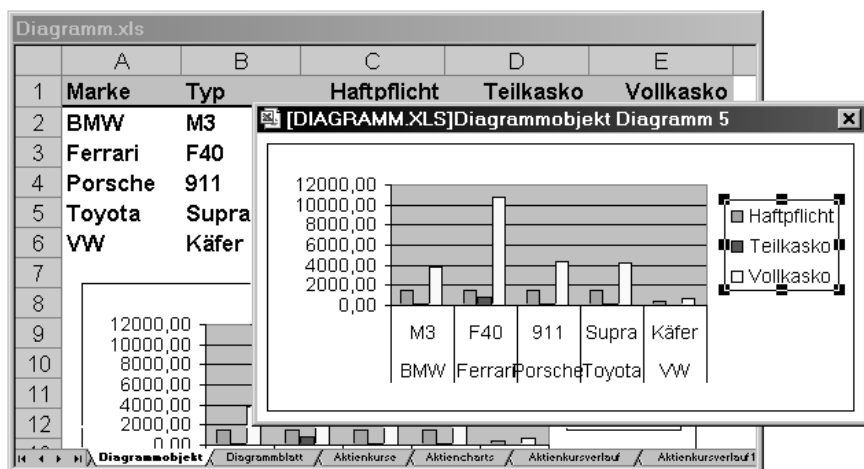


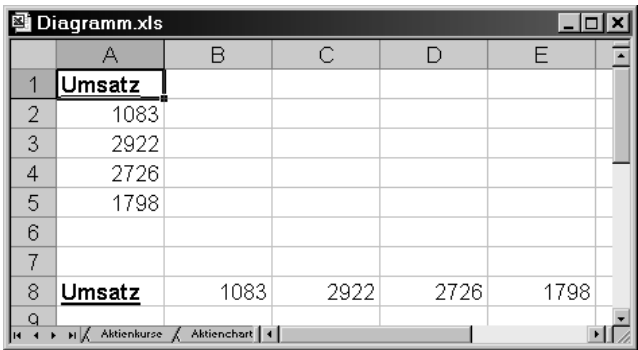
Abbildung 22.9:
Diagrammfenster

Das Diagrammfenster wird übrigens sofort wieder geschlossen, wenn Sie das Tabellenfenster anklicken.

22.3 Datenreihen und Rubriken

Im zweiten Schritt des Assistenten legen Sie nicht nur den dazustellenden Datenbereich fest, sondern auch die Interpretation der Datenreihen (siehe Abbildung 22.4). Eine Datenreihe kann sich in einer Spalte oder in einer Zeile befinden (Abbildung 22.10).

Abbildung 22.10:
Datenreihen-
anordnung



The screenshot shows an Excel window titled 'Diagramm.xls'. The active sheet contains a table with columns A through E. Row 1 has 'Umsatz' in column A. Rows 2 through 5 have numerical values in column A: 1083, 2922, 2726, and 1798 respectively. Row 8 has 'Umsatz' in column A, and columns B through E contain the values 1083, 2922, 2726, and 1798 respectively. The status bar at the bottom shows 'Aktienkurse' and 'Aktienchart'.

	A	B	C	D	E
1	Umsatz				
2	1083				
3	2922				
4	2726				
5	1798				
6					
7					
8	Umsatz	1083	2922	2726	1798

Der Assistent interpretiert beide Fälle korrekt. Erzeugen Sie ein Diagramm, wenn zuvor eine Zelle des Bereichs A1:A5 selektiert ist, so erkennt er, dass Spalte A eine Datenreihe enthält und unter *Reihe in* ist die Option *Spalten* vorselektiert.

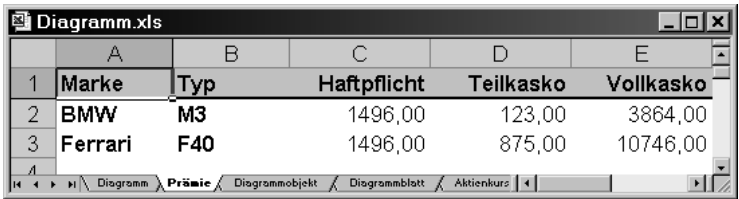
Erzeugen Sie ein weiteres Diagramm mit dem Bereich A8:E8, so erkennt er ebenfalls korrekt, dass nun Zeile 8 eine Datenreihe enthält und gibt entsprechend die Option *Zeilen* vor.



Bei dieser automatischen Erkennung geht der Assistent davon aus, dass es üblicherweise mehr Elemente gibt, die miteinander verglichen werden sollen, als Vergleichskategorien, mehr Rubriken als Datenreihen.

Diese Annahme trifft jedoch nicht immer zu. Nehmen wir an, Sie wollen nur zwei Autotypen miteinander vergleichen, zwei Rubriken, und zwar in Bezug auf drei Versicherungsarten, drei Kategorien (Abbildung 22.11).

Abbildung 22.11:
Mehr Rubriken als
Datenreihen



The screenshot shows an Excel window titled 'Diagramm.xls'. The active sheet contains a table with columns A through E. Row 1 has 'Marke' in column A, 'Typ' in column B, and 'Haftpflicht', 'Teilkasko', and 'Vollkasko' in columns C through E respectively. Rows 2 and 3 have numerical values in columns C through E: BMW M3 (1496,00, 123,00, 3864,00) and Ferrari F40 (1496,00, 875,00, 10746,00). The status bar at the bottom shows 'Diagramm', 'Prämie', 'Diagrammobjekt', 'Diagrammblatt', and 'Aktienkurs'.

	A	B	C	D	E
1	Marke	Typ	Haftpflicht	Teilkasko	Vollkasko
2	BMW	M3	1496,00	123,00	3864,00
3	Ferrari	F40	1496,00	875,00	10746,00

Mit den vorgegebenen Optionen haben Sie Pech: Erzeugen Sie ein Diagramm mit dem Bereich A1:E3 als Datenbasis, so wird nicht wie gewünscht

Spalten vorgegeben, sondern *Zeilen*. Der Assistent interpretiert die Zeilen, also die Autotypen, als Datenreihen, und nicht wie gewünscht die Spalten, die Versicherungsarten (Abbildung 22.12).

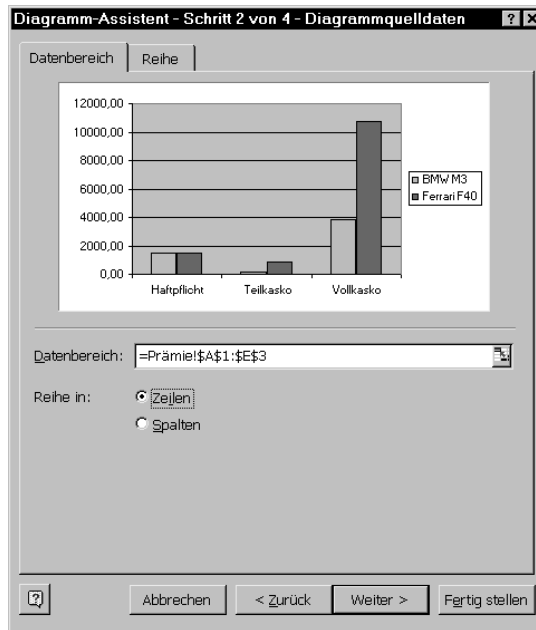


Abbildung 22.12:
Fehlinterpertation

Ursache dafür ist, dass der Datenbereich A1:E3 mehr Spalten als Zeilen enthält und der Assistent daher gemäß der geschilderten Annahme davon ausgeht, dass die Zeilen die Datenreihen enthalten, also die beiden Bereiche C2:E2 und C3:E3.

Um diese Fehlinterpretation zu korrigieren, aktivieren Sie einfach die Option *Spalten*, bevor Sie die Diagrammerzeugung fortsetzen. Die resultierenden Auswirkungen werden sofort angezeigt (Abbildung 22.13).

Zusätzlich können Sie in diesem Schritt im Register *Reihe* die Festlegung der Datenreihen und der in der Legende angezeigten Datenreihenbeschriftungen näher festlegen (Abbildung 22.14).

**Datenreihen-
beschriftungen**

Der Assistent zeigt im Listenfeld die von ihm erkannten drei Datenreihen an und rechts daneben den in der Legende anzuzeigenden Namen der Datenreihe (*Name*) bzw. den darzustellenden Wertebereich (*Werte*).

Beides können Sie für jede der drei Datenreihen individuell ändern. Dazu klicken Sie im Listenfeld einfach die betreffende Datenreihe an, beispielsweise »Haftplicht«. Unter *Name* erscheint nun der Bezug jener Zelle, die als Beschriftung verwendet wird, im Beispiel C1.

Abbildung 22.13:
Spalten als
Datenreihen

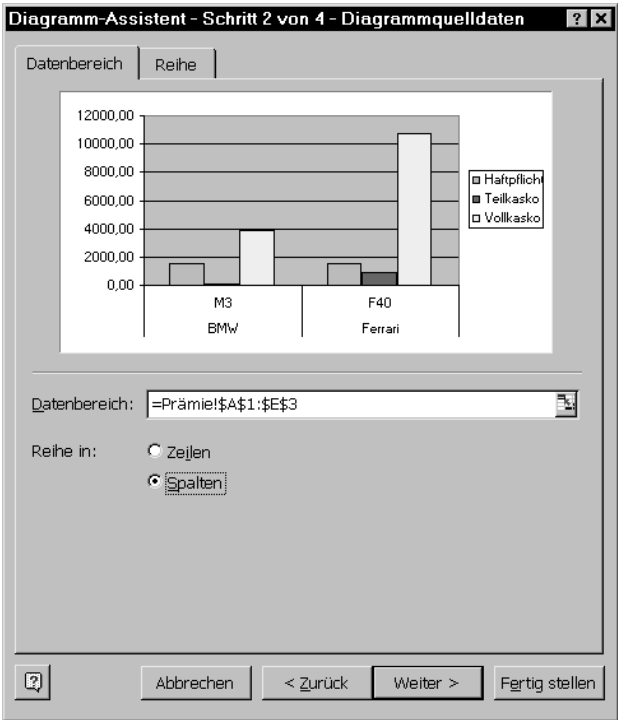
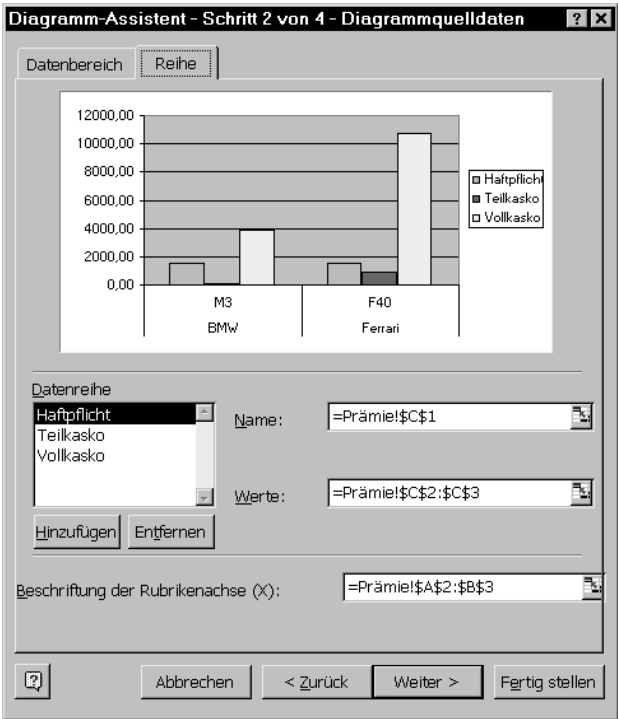


Abbildung 22.14:
Datenreihen-
beschriftungen



Um den Assistenten zu überreden, für diese Datenreihe in der Legende eine andere Beschriftung zu verwenden, tippen Sie den gewünschten Text, beispielsweise »Hallo«, einfach ein oder zeigen auf eine Zelle, die den zu verwendenden Text enthält.

Aktivieren Sie mit *Weiter* > den nächsten Schritt und kehren Sie dann mit < *Zurück* wieder zurück, erscheint im Listenfeld statt »Teilkasko« nun der neue Name, z.B. »Hallo«.

Unter *Werte* können Sie den Assistenten dazu bringen, für die Datenreihe statt C2:C3 einen anderen Zahlenbereich zu verwenden, zum Beispiel den Bereich G22:G26 (im Beispiel allerdings ein unsinniger Bereich, da leer), indem Sie einfach auf diesen Bereich zeigen.

Sie können dem Diagramm sogar zusätzliche Datenreihen hinzufügen, indem Sie auf *Hinzufügen* klicken: im Listenfeld wird ein Standardname wie »Reihe4« vorgegeben und Sie legen unter *Name* den Namen der Datenreihe und unter *Werte* die darzustellenden Werte fest. Mit *Entfernen* können Sie genau umgekehrt die zuvor im Listenfeld selektierte Datenreihe aus dem Diagramm entfernen.

Wann benötigen Sie dieses Register eigentlich? Sie benötigen es vor allem, wenn sich der Assistent bei seinen Analysen des Datenbereichs irrt, zum Beispiel, wenn der Autotyp nicht im Klartext in die zweite Spalte eingetragen ist, sondern stattdessen ein Typcode verwendet wird (Abbildung 22.15).

Der Assistent erkennt zwar korrekt, dass nur eine Zeile am oberen Bereichsrand der Rubriken Legendentexte enthält und gibt entsprechend im Register *Datenbereich* die Option *Spalten* vor, erkennt also, dass sich die Datenreihen in den einzelnen Spalten des Bereichs befinden.

Er erkennt jedoch nicht, dass am linken Bereichsrand *zwei* Spalten Rubrikenbeschriftungen enthalten, sondern gibt unter *Beschriftung der Rubrikenachse* (X) den Bereich A2:A6 vor und nicht wie gewünscht A2:B6.

Das Problem ist, dass Spalte B keine Texte, sondern Zahlen enthält, die der Assistent verständlicherweise für eine vierte Datenreihe namens »Typ« hält (siehe Legende und Listenfeld *Datenreihe*). Wie die Vorschau zeigt, würde das Resultat recht unglücklich ausfallen.

Sie müssen die fehlerhafte Interpretation korrigieren, indem Sie im Listenfeld *Datenreihe* die angebliche Datenreihe »Typ« selektieren, mit *Entfernen* aus dem Diagramm entfernen und zusätzlich unter *Beschriftung der Rubrikenachse* (X) den Bereich A2:B6 selektieren und den Assistenten damit zwingen, zwei Spalten ab dem linken Bereichsrand als Rubrikenbeschriftung aufzufassen, also Spalte A und Spalte B (Abbildung 22.16).

Abbildung 22.15:
Fehlerhafte
Bereichs-
interpretation

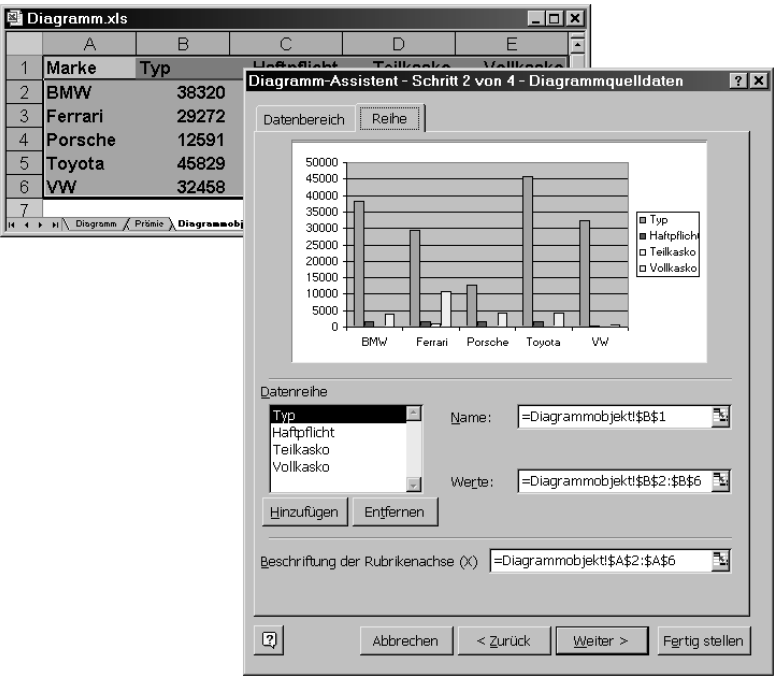
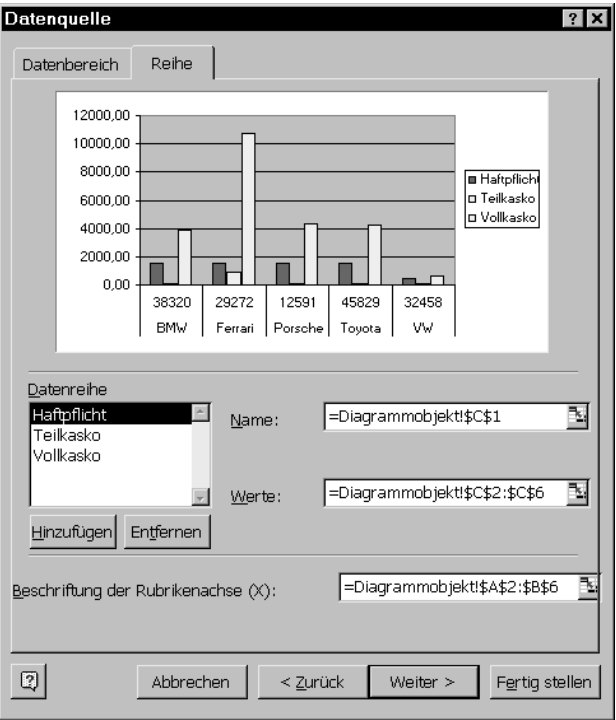


Abbildung 22.16:
Manuelle Korrektur



Nun werden die Codenummern korrekt als Rubrikenbeschriftungen verwendet und nicht wie zuvor fälschlich als zusätzliche Datenreihe.

Wollen Sie ein bereits bestehendes Diagramm verändern, so können Sie das Dialogfeld mit den Registern Datenbereich und Reihe nach Selektion des Diagramms übrigens auch direkt mit dem Befehl DIAGRAMM|DATEN-QUELLE... öffnen (nur verfügbar, wenn irgendein Diagrammteil selektiert ist), also ohne den Umweg über den Diagramm-Assistenten.



22.4 Diagrammblätter

Wählen Sie im letzten Schritt der Diagrammerzeugung *Als neues Blatt*, so legt Excel ein Diagrammblatt an, das ausschließlich für die Darstellung des betreffenden Diagramms verwendet wird (Abbildung 22.17).

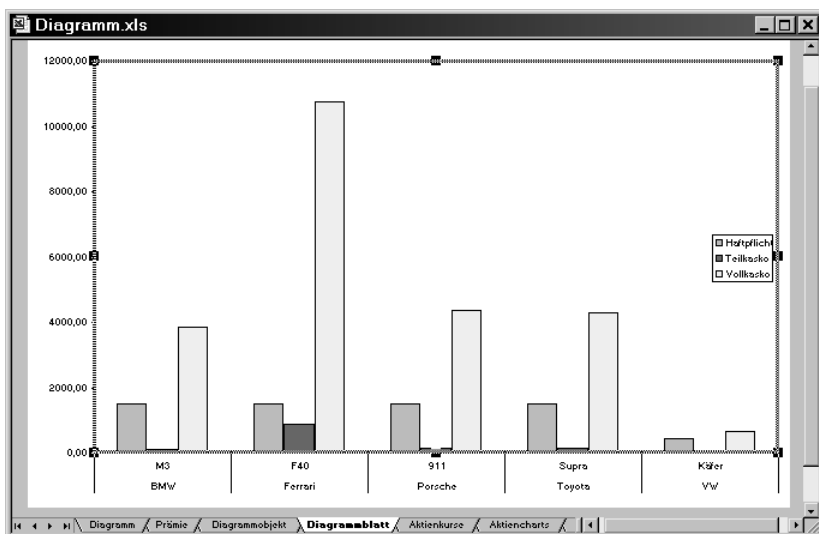


Abbildung 22.17:
Diagramm.xls,
»Diagrammblatt«

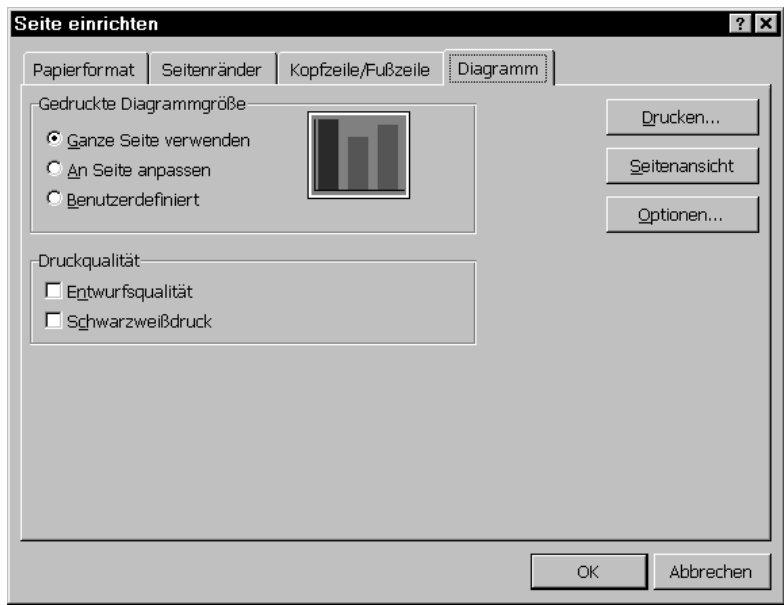
Um es später einmal in eine Tabelle einzufügen, wählen Sie DIAGRAMM|SPEICHERORT..., aktivieren die Option *Als Objekt in* und selektieren im zugehörigen Listenfeld die Tabelle, in die es als Objekt eingefügt werden soll. Alternativ dazu können Sie auch einfach erneut den Diagramm-Assistenten aktivieren und im letzten Schritt statt *Als neues Blatt* diesmal *Als Objekt in* wählen.

Um die Diagrammgröße zu ändern, benutzen Sie im Befehl DATEI|SEITE EINRICHTEN... das Register *Diagramm* (Abbildung 22.18).

*Diagramm-
Druckgröße*

Benutzerdefiniert ermöglicht Ihnen die manuelle Festlegung der Diagrammgröße.

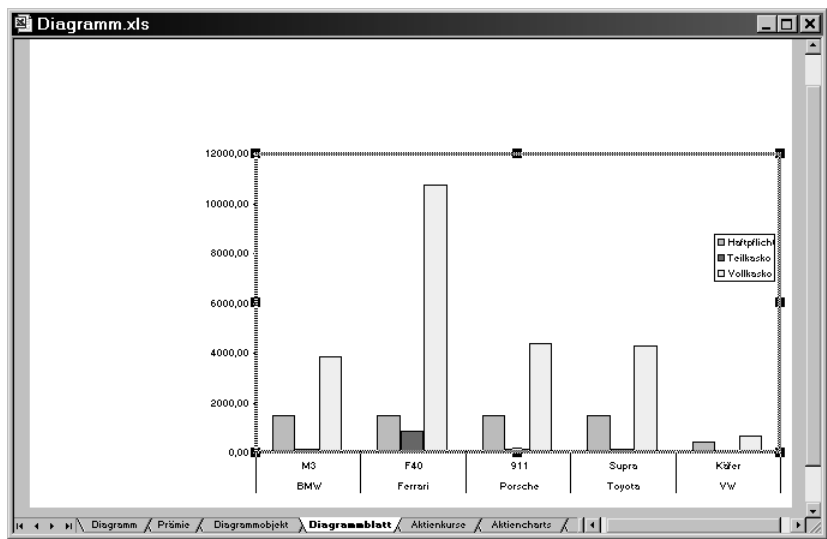
Abbildung 22.18:
Diagramm-
Druckoptionen



Ist diese Option aktiviert, können Sie die Druckgröße und -position anschließend mit der Maus festlegen: Sie selektieren das Diagramm per Klick auf einen leeren Punkt der Diagrammfläche. Die Anfasser erscheinen und Sie können das Diagramm *in den Grenzen des weiß dargestellten Blatts* verkleinern oder vergrößern.

Durch Ziehen an einer Randlinie können Sie sogar seine Position ändern (Abbildung 22.19).

Abbildung 22.19:
Diagramm
verkleinern und
verschieben



Die beiden anderen Optionen stellen automatisch eine meist geeignete Diagrammgröße ein: *An Seite anpassen* vergrößert das Diagramm in X- und Y-Richtung gleichermaßen, bis die maximal mögliche Breite oder Höhe erreicht ist. Das Breite-/Höhe-Verhältnis wird dabei unverändert beibehalten (Abbildung 22.20).

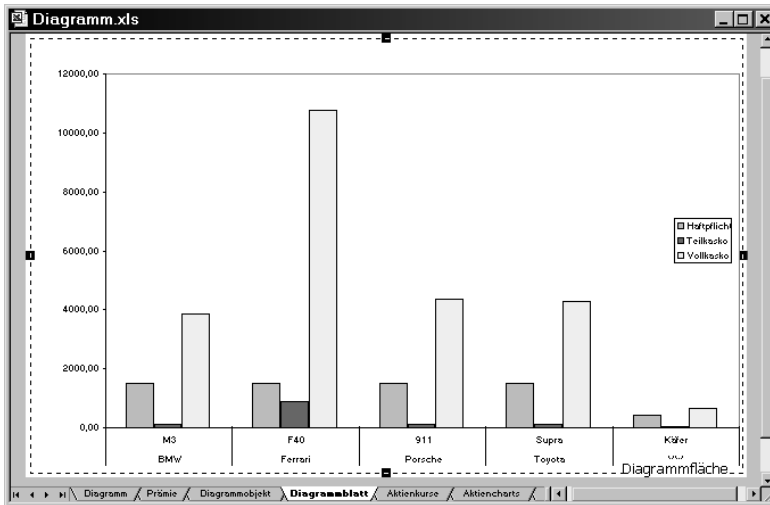


Abbildung 22.20:
An Seite ange-
passtes Diagramm

Ganze Seite verwenden dehnt das Diagramm in X- und in Y-Richtung so weit wie möglich aus, benutzt also das gesamte Blatt zur Darstellung, wobei sich das Breite-/Höhe-Verhältnis zwangsläufig ändert.

Diese Verzerrung stört besonders, wenn das Diagramm eine längliche Form besitzt, also deutlich breiter als hoch ist, und wenn Sie im Hochformat drucken. Verwenden Sie in solchen Fällen wie in den Abbildungen das Querformat, um die Proportionen wenigstens grob beizubehalten.



Entwurfsqualität druckt größer, aber schneller.

Schwarzweißdruck druckt alle Vordergrundelemente, die nicht vollkommen weiß sind, schwarz aus; alle Hintergrundelemente, die nicht vollkommen schwarz sind, werden weiß ausgedruckt.

22.5 Zusätzliche Diagrammelemente einfügen

Sie können in ein Diagramm jederzeit zusätzliche Elemente einfügen. Die am häufigsten benutzten fügen Sie ein, indem Sie das Diagramm durch Anklicken selektieren, auf das *Diagramm-Assistenten*-Symbol klicken und sich zum noch nicht erläuterten dritten Schritt vortasten. Darin steht jedes Register für ein anderes einzufügendes Diagrammelement.

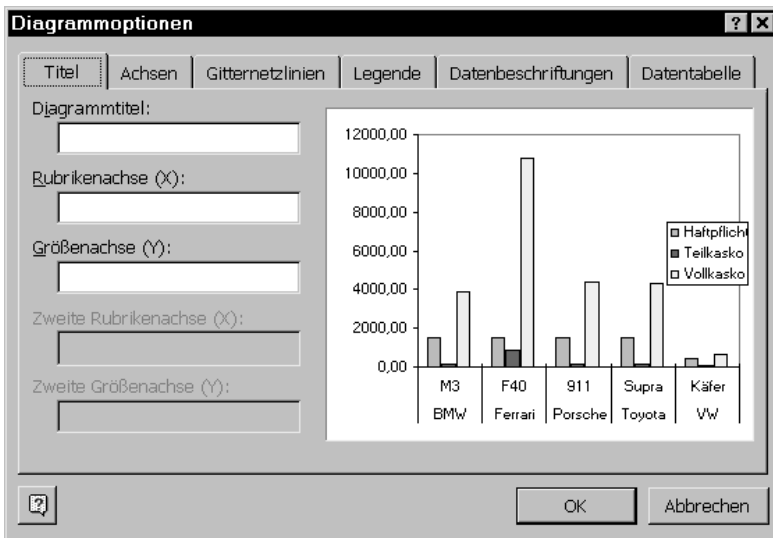
Sie können dieses Register auch direkt aufschlagen, indem Sie **DIAGRAMM|DIAGRAMMOPTIONEN...** wählen (nur verfügbar, wenn irgendein Diagrammteil selektiert ist).

Ich erläutere nun zuerst in den folgenden sechs Unterkapiteln dieses Register und zeige Ihnen danach, welche Elemente Sie darüber hinaus einfügen können ohne den Diagramm-Assistenten.

22.5.1 Diagramm- und Achsentitel

Im Register *Titel* können Sie unterschiedliche Arten von Titeltiteln einfügen, einen Diagrammtitel, einen Titel für die X-Achse und einen Titel für die Y-Achse (Abbildung 22.21).

Abbildung 22.21:
Titel



Tippen Sie den gewünschten Titel in ein Feld ein und verlassen Sie dieses Feld durch Selektion eines anderen Felds, so wird die Diagrammvorschau sofort entsprechend aktualisiert und der eingetragene Titel erscheint darin.

22.5.2 Achsen

Im Register *Achsen* legen Sie fest, ob die X- und die Y-Achse überhaupt angezeigt werden sollen. Sie können die Anzeige beider Achsen unabhängig voneinander ein- oder ausschalten (Abbildung 22.22).

Wesentlich interessanter ist diese Möglichkeit jedoch, wenn Sie sie auf ein Verbunddiagramm anwenden, das mehrere Diagrammtypen enthält (Abbildung 22.23).

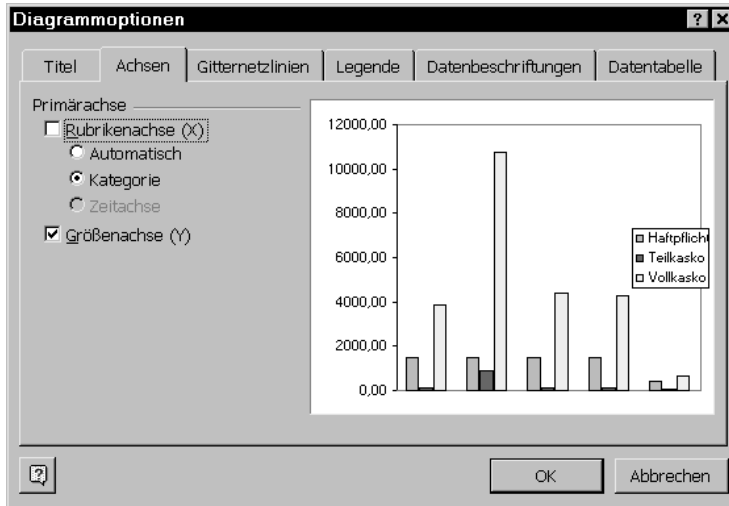


Abbildung 22.22:
Keine Rubriken-
achse anzeigen

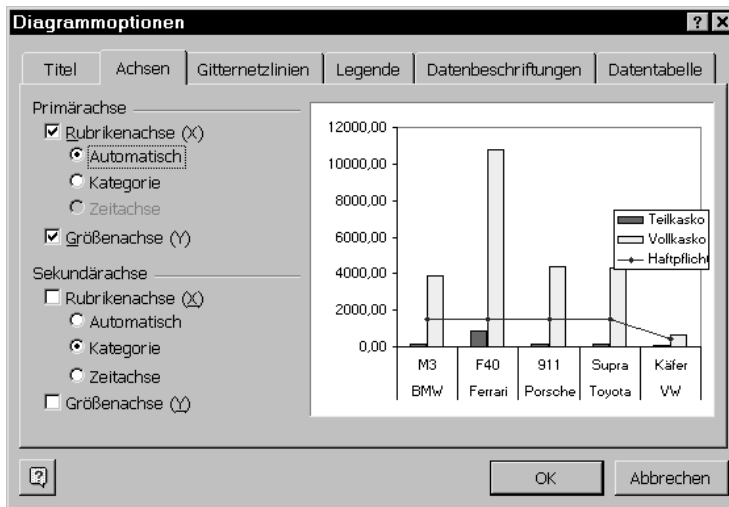


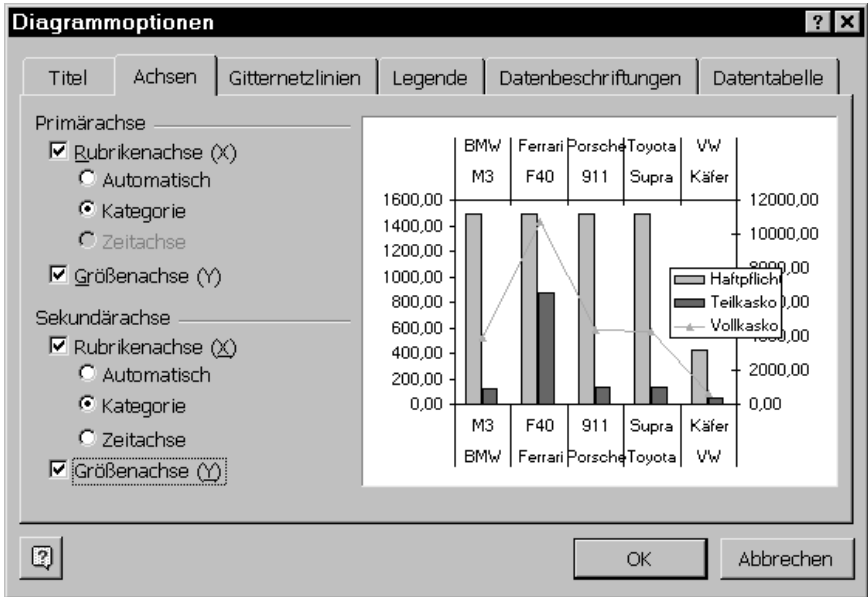
Abbildung 22.23:
Sekundärachsen

Dann können Sie für beide Diagrammtypen getrennt die Rubriken- und die Größenachse wahlweise aus- oder einblenden.

Blenden Sie zusätzlich unter *Sekundärachse* beide Achsen ein, so wird die Rubrikenachse am oberen Diagrammrand gespiegelt.

Vor allem aber erscheint am rechten Bildschirmrand eine zusätzliche Größenachse. Die daran aufgetragenen Werte beziehen sich ausschließlich auf das Nebendiagramm, im folgenden Beispiel ein Liniendiagramm mit genau einer als Linie dargestellten Datenreihe, »Vollkasko« (Abbildung 22.24).

Abbildung 22.24:
Zusätzliche Größen-
und Rubrikenachse
für das Linien-
diagramm



Wählen Sie unter *Rubrikenachse (X)* die Option *Zeitachse*, so wird *in jedem Fall* eine Zeitachse mit Datumsbeschriftungen wie »01.01.09«, »02.01.09« etc. eingeblendet, selbst wenn den Daten in Ihrer Tabelle, die als X-Achsenbeschriftung verwendet werden, kein Datumsformat zugewiesen ist.

Handelt es sich dabei um Daten wie »M3 BMW«, die Excel nicht als Datum interpretieren kann, so werden einfach Standard-Datumswerte wie »01.01.00«, »02.01.00« verwendet, die zwar nicht unbedingt sinnvoll sind, aber zumindest darauf hinweisen, dass es sich bei der Darstellung um eine Zeitreihe handelt.

22.5.3 Gitternetzlinien

Im Register *Gitternetzlinien* blenden Sie Gitternetzlinien ein bzw. aus (Abbildung 22.25).

Unter Hauptgitternetz sind die Linien zu verstehen, die sich von den Hauptstrichen der betreffenden Achse über das Diagramm erstrecken, und unter Hilfsgitternetz die Linien, die von den Hilfsstrichen ausgehen.

Wäre auch für die X-Achse die Option *Hilfsgitternetz* aktiviert worden, enthielte das Diagramm zusätzlich zwischen je zwei vertikalen Linien eine weitere vertikale Linie.

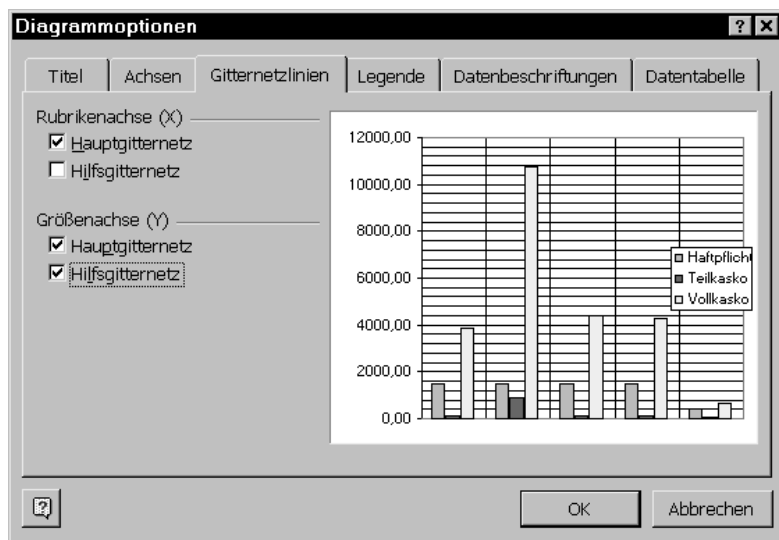


Abbildung 22.25:
Gitternetzlinien

22.5.4 Legende

Das Register *Legende* ermöglicht zunächst, mit dem Kontrollkästchen *Legende anzeigen* die Legende ein- bzw. auszublen- den (Abbildung 22.26).

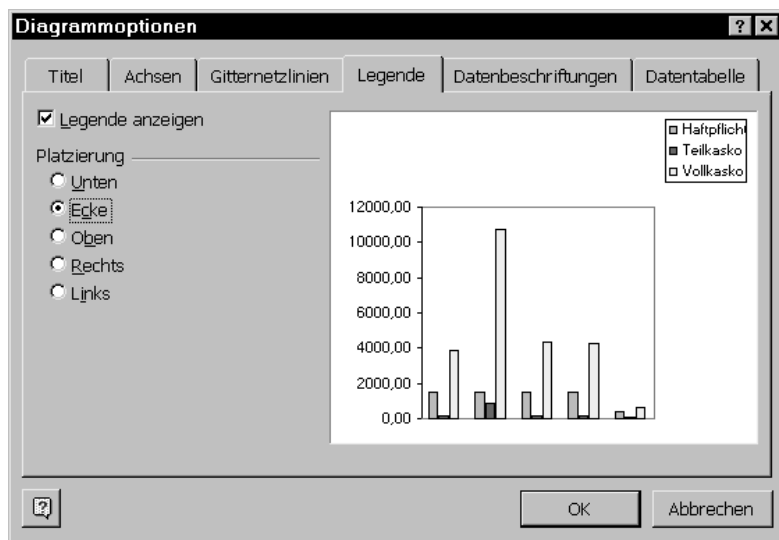


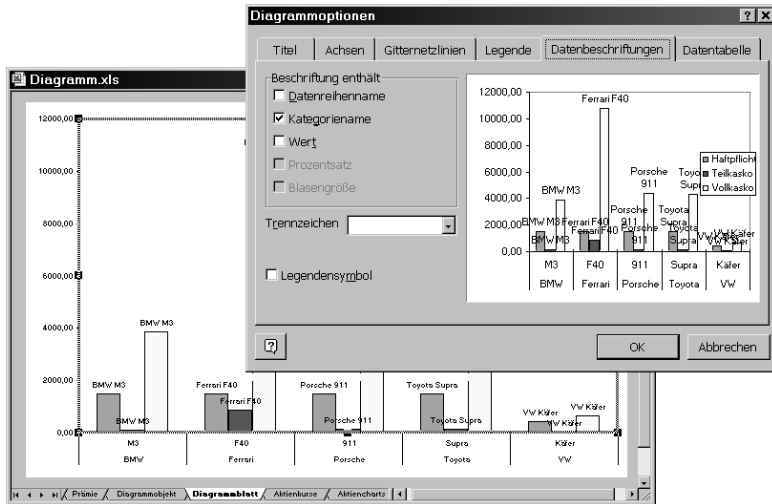
Abbildung 22.26:
Legende

Zusätzlich können Sie darin festlegen, wo sich die Legende befinden soll, oberhalb des Diagramms, unterhalb, rechts oder links davon oder aber (*Ecke*) wie in der Abbildung in der oberen rechten Ecke.

22.5.5 Datenbeschriftung

Im Register *Datenbeschriftung* wird festgelegt, welche Texte oder Zahlen neben den einzelnen Datenpunkten angezeigt werden (Abbildung 22.27).

Abbildung 22.27:
Daten-
beschriftungen



- ➔ *Datenreihenname* zeigt die zugehörige Datenreihenbeschriftung an, z.B. »Vollkasko«.
- ➔ *Kategorienname* zeigt die zugehörige Rubrikenbeschriftung an, z.B. »M3 BMW« (siehe Abbildung).
- ➔ *Wert* zeigt neben jedem Datenpunkt den Wert an, den dieser darstellt.
- ➔ *Prozentsatz* zeigt statt des absoluten Werts den jeweiligen prozentualen Anteil an der Datenreihe an.
- ➔ *Blasengröße* zeigt in Blasendiagrammen die Größe der einzelnen Blasen an.

Sie können beliebig viele dieser Beschriftungen auswählen und zusätzlich im Listenfeld *Trennzeichen* das Trennzeichen wählen, mit dem mehrere Beschriftungen eines Datenpunktes voneinander getrennt werden oder ob sie gar in einzelnen Zeilen untereinander erscheinen sollen.

Das Kontrollkästchen *Legendensymbol* fügt zusätzlich das Legendensymbol ein, das zur betreffenden Datenreihe gehört, das kleine farbige Kästchen, das sich in der Legende neben dem Datenreihennamen befindet.

Welche dieser Optionen verfügbar sind, hängt vom aktuellen Diagrammtyp ab. Beispielsweise ist Blasengröße verständlicherweise nur bei Blasendiagrammen verfügbar.

22.5.6 Datentabelle

Aktivieren Sie im Register *Datentabelle* das Kontrollkästchen *Datentabelle anzeigen*, wird unterhalb des Diagramms als Mini-Tabelle der Datenbereich angezeigt, auf dem das Diagramm basiert (Abbildung 22.28).

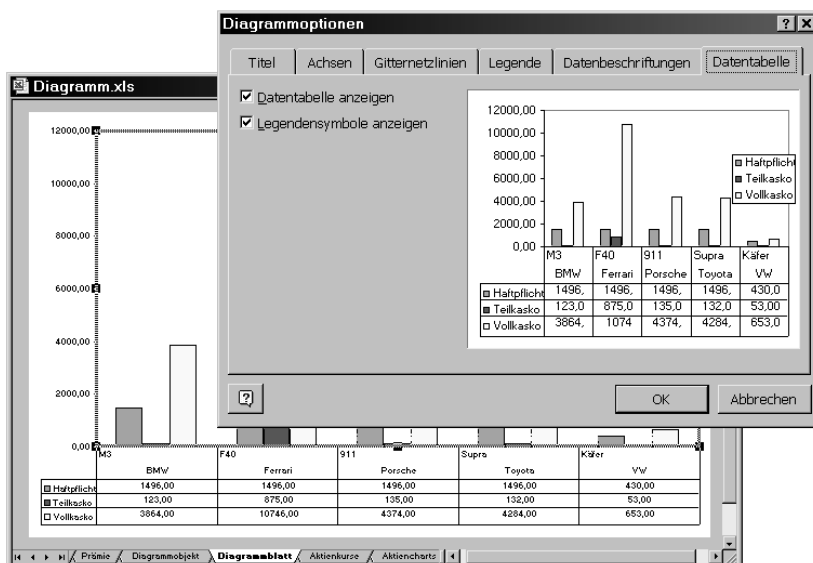


Abbildung 22.28:
Daten-Tabelle

Nur in diesem Fall ist *Legendensymbole anzeigen* aktivierbar, womit in der Minitabelle neben den Datenreihennamen auch die zugehörigen Legenden-Symbole angezeigt werden.

22.5.7 Ungebundene Texte

Ist ein Diagrammblatt oder aber – bei in Tabellen integrierten Diagrammen – das Diagramm aktiviert, können Sie jederzeit Text eintippen, der zunächst in der Bearbeitungsleiste erscheint. schließt die Eingabe ab und zeigt den Text im Diagramm an (Abbildung 22.29).

Sie können das Textfeld jederzeit durch Daraufklicken selektieren und wie gewohnt verschieben und seine Form oder seine Größe verändern.

22.5.8 Trendlinien

DIAGRAMM|TRENDLINIE HINZUFÜGEN... (nur verfügbar, wenn ein Diagrammelement selektiert ist; zusätzlich ist dieser Befehl jedoch auch über das Kontextmenü der Datenreihenbilder erreichbar) öffnet folgendes Dialogfeld (Abbildung 22.30):

Abbildung 22.29:
Ungebundener Text

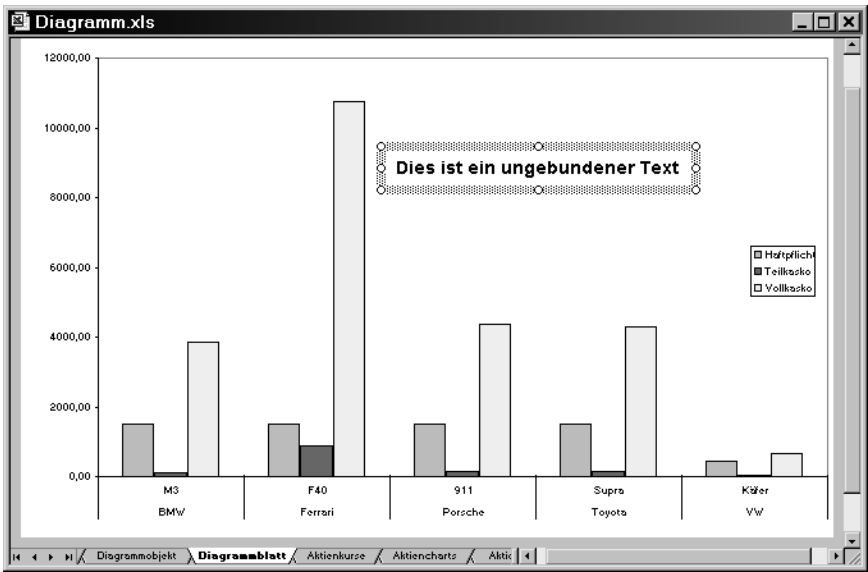
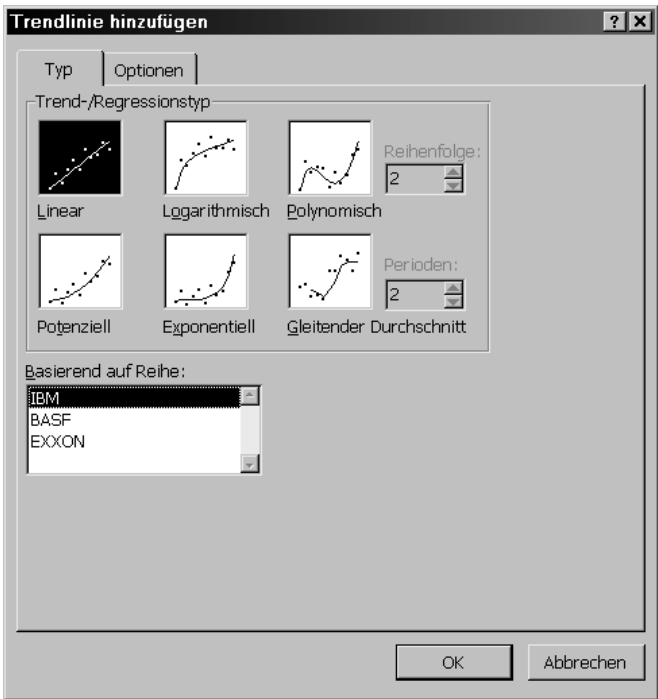


Abbildung 22.30:
Trendlinie
hinzufügen



Im Listenfeld wählen Sie die Datenreihe aus, deren Trend beschrieben werden soll, und darüber wählen Sie den Regressionstyp, der Ihrer Meinung nach diesem Trend zu Grunde liegt.

Excel errechnet die Funktion, die die vorhandenen Werte am besten beschreibt und zeichnet die resultierende Trendlinie ein, angewandt auf ein Diagramm, das drei Aktienkurse enthält (Abbildung 22.31).

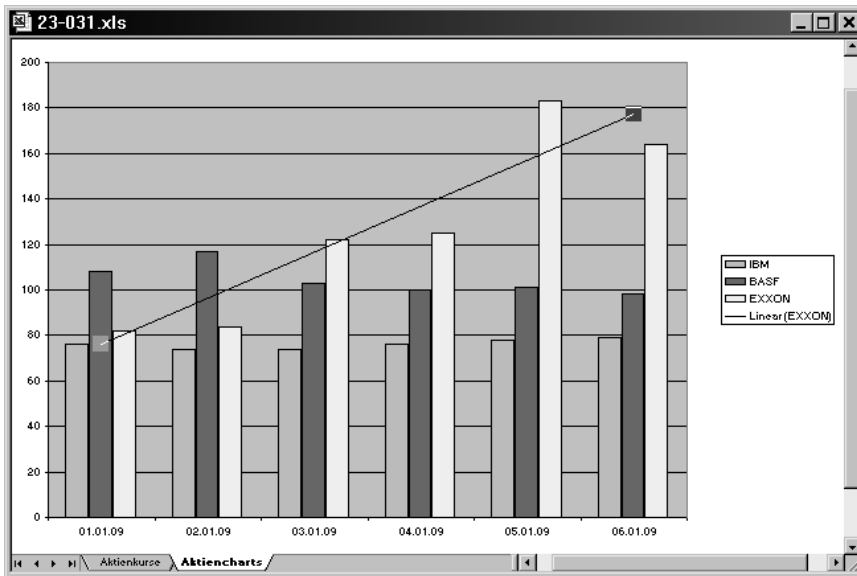


Abbildung 22.31:
Lineare Regression

JIL Für diese Abbildung selektierte ich die dritte Datenreihe, wählte DIAGRAMM|TRENDLINIE HINZUFÜGEN... und gab als Typ *linear* an: Excel ermittelte gemäß der »Kleinste Summe der Abweichungsquadrate«-Methode die lineare Funktion, die am besten an die Datenreihe angepasst ist, und zeichnet die resultierende Gerade ein.

Aktienexperten sollten übrigens die Option Gleitender Durchschnitt mit freier Wahl der beim Berechnen des Durchschnitts zu verwendenden Periodenanzahl beachten!

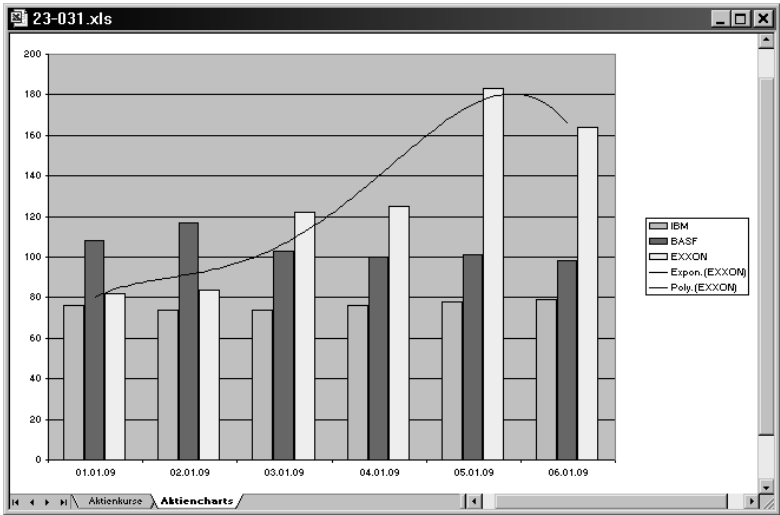


Im Beispiel liegt übrigens mit größter Wahrscheinlichkeit kein linearer Zusammenhang vor und eine lineare Regression ist daher nicht angebracht. Statt einer linearen wäre vermutlich eine Exponentialfunktion wesentlich besser an die vorhandenen Daten angepasst.

Also wählen Sie erneut DIAGRAMM|TRENDLINIE HINZUFÜGEN... (oder benutzen das TRENDLINIEN-Kontextmenü, indem Sie mit der rechten Maustaste auf die Trenlinie klicken), selektieren diesmal jedoch *Exponentiell*.

Das Resultat sieht bereits wesentlich vernünftiger aus, was jedoch mit einem Polynom vierter Ordnung noch gesteigert werden kann (Abbildung 22.32).

Abbildung 22.32:
Polynom vierter
Ordnung



Die Trendlinie ist nun hervorragend an die vorhandenen Daten angepasst – allerdings beweist das natürlich keineswegs, dass auch tatsächlich der vermutete Zusammenhang vorliegt und mit aus dieser Kurve resultierenden Schlüssen wie »Aha, mit EXXON geht's neuerdings abwärts« sollten Sie entsprechend vorsichtig sein!

Register *Optionen*

Das Register *Optionen* (Abbildung 22.33).

Abbildung 22.33:
Register »Optionen«

Trendlinie hinzufügen

Typ: **Optionen**

Name der Trendlinie:
☒ Automatisch: Linear (IBM)
☐ Benutzerdefiniert:

Trend:
Vorwärts: Perioden
Rückwärts: Perioden

☐ Schnittpunkt =
☐ Gleichung im Diagramm darstellen
☐ Bestimmtheitsmaß im Diagramm darstellen

OK Abbrechen

Unter *Name der Trendlinie* geben Sie der Trendlinie einen Namen, der in der Legende angezeigt wird.

Aktivieren Sie *Benutzerdefiniert*, können Sie im zugehörigen Eingabefeld einen frei wählbaren Namen eingeben

Automatisch verwendet den über dem Eingabefeld angezeigten Namen, der auf dem Legendentext der Datenreihe basiert.

Vorwärts und *Rückwärts* bestimmen, wie viele Rubrikeneinheiten der Trend in die Zukunft bzw. Vergangenheit projiziert werden soll.

Gleichung im Diagramm darstellen und *Bestimmtheitsmaß im Diagramm darstellen* zeigt die ermittelte Regressionsfunktion bzw. das Bestimmtheitsmaß an.

Der Wert im Eingabefeld von *Schnittpunkt* = definiert den Wert der Funktion beim X-Wert 0, also jenen Wert, bei dem die Trendlinie die Y-Achse am Ausgangspunkt trifft.

Lassen Sie den Trend beispielsweise zwei Einheiten in die Zukunft projizieren und benutzen Sie zusätzlich die erläuterten Beschriftungsoptionen für die Trendlinie, die Gleichung und das Bestimmtheitsmaß, erhalten Sie folgendes Resultat (Abbildung 22.34).

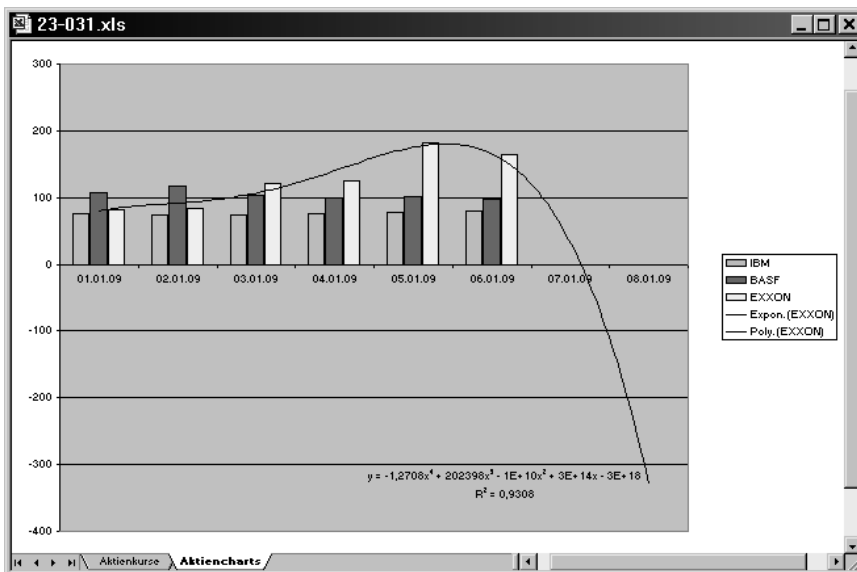


Abbildung 22.34:
Alle Optionen
aktiviert

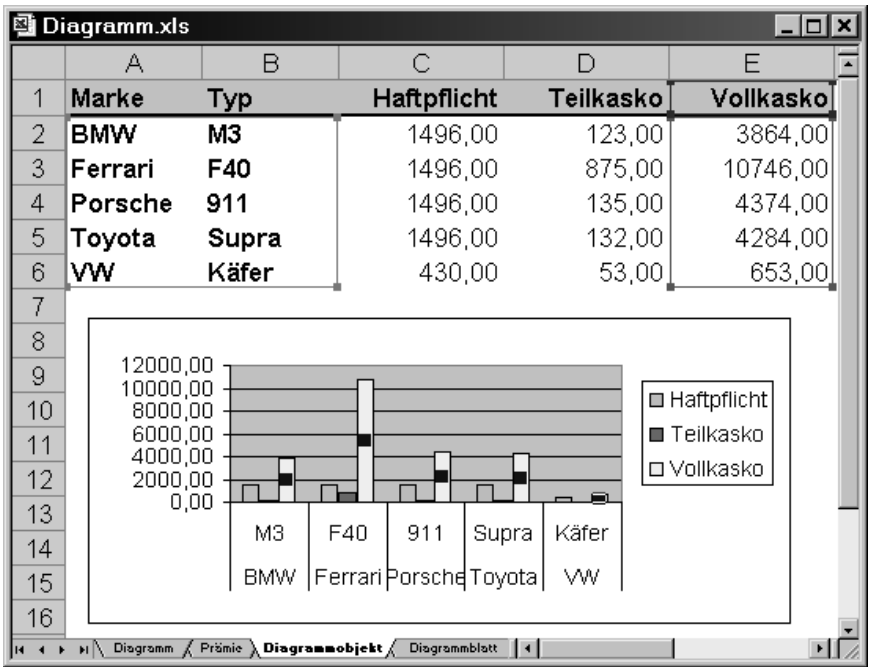
22.5.9 Zusätzliche Daten

Das Einfügen zusätzlicher Daten erspart Ihnen, Diagramme ständig neu erzeugen zu müssen, wenn sich die zugrundeliegende Datenbasis erweitert. Sie können sich unter verschiedenen Möglichkeiten jene aussuchen, die Ihnen am meisten zusagt.

Drag&Drop Die vermutlich einfachste: Markieren Sie die einzufügenden Daten nach dem Eintippen, ziehen Sie sie in den Diagrammbereich und lassen Sie sie dort fallen.

Alternativ dazu können Sie so vorgehen: Selektieren Sie durch Anklicken die zu verändernde Datenreihe. Excel hebt daraufhin die zugehörigen Daten durch farbige Rahmen hervor (Abbildung 22.35).

Abbildung 22.35:
Hervorhebung der
Datenquelle



Durch je einen Rahmen werden die Achsenbeschriftungen und der Bereich E1:E6 hervorgehoben, der die Legendenbeschriftung und die Daten der selektierten Reihe enthält.

Durch Ziehen am Kästchen in der rechten unteren Rahmenecke können Sie den Rahmen verlängern oder verkürzen und damit die dargestellte Datenreihe verändern, beispielsweise durch Verkürzen des Rahmens die drei letzten Daten ausschließen (Abbildung 22.36).

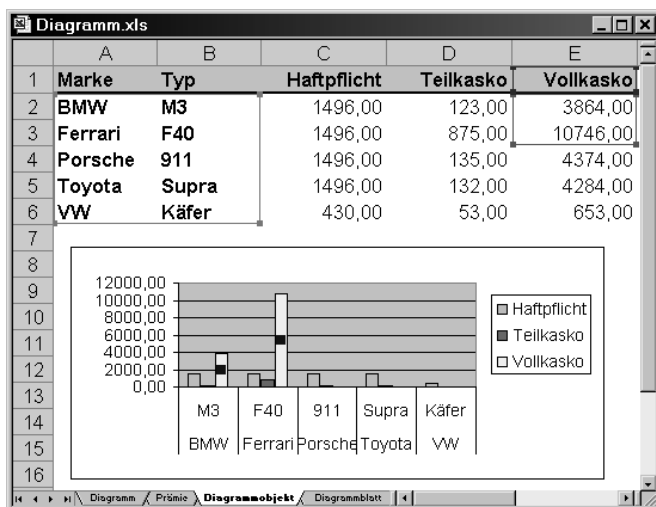


Abbildung 22.36:

Datenreihe
verändern

Genau umgekehrt wäre durch Ziehen des Kästchens nach unten der Rahmen verlängert worden und würde nun auch Daten umfassen, die sich in den Zellen unterhalb von E6 befinden und zu der Datenreihe hinzugefügt werden sollen.

Sie können den Rahmen sogar durch Ziehen an einem der Ränder komplett verschieben, um durch die betreffende Datenreihe einen völlig anderen Datenbereich der Tabelle darzustellen.

Per Drag&Drop können Sie sogar völlig neue Datenreihen einfügen: Sie selektieren den betreffenden Tabellenbereich, klicken danach auf den Rand des selektierten Bereichs und ziehen ihn auf das Diagramm. Nach dem Loslassen der Maustaste werden die betreffenden Daten eingefügt.

Die gleiche Funktion besitzt der Befehl **DIAGRAMM|DATEN HINZUFÜGEN...** (nur verfügbar, wenn ein Diagrammelement selektiert ist). Nehmen wir an, Sie erweitern die Tabelle »Prämie« um eine zusätzliche Datenreihe »Steuer«.

Per Menübefehl

Um sie dem Diagramm hinzuzufügen, wählen Sie **DIAGRAMM|DATEN HINZUFÜGEN...**. Ein Dialogfeld erscheint, in dem Sie den hinzuzufügenden Datenbereich inklusive eventuell vorhandener Spalten- und Zeilenüberschriften selektieren sollen (natürlich nur, wenn die Legende und die Rubrikenbeschriftungen entsprechend erweitert werden sollen) (Abbildung 22.37).

Sie aktivieren die Tabelle mit der Datenbasis und zeigen entsprechend der Abbildung auf die hinzuzufügende Reihe inklusive der Überschrift »Steuer«.

Der Bereichsbezug wird in das Dialogfeld eingefügt und das Diagramm nach dem Schließen um die zusätzliche Datenreihe und die Legende um die zusätzliche Überschrift erweitert (Abbildung 22.38).

Abbildung 22.37:
Datenbasis
erweitern

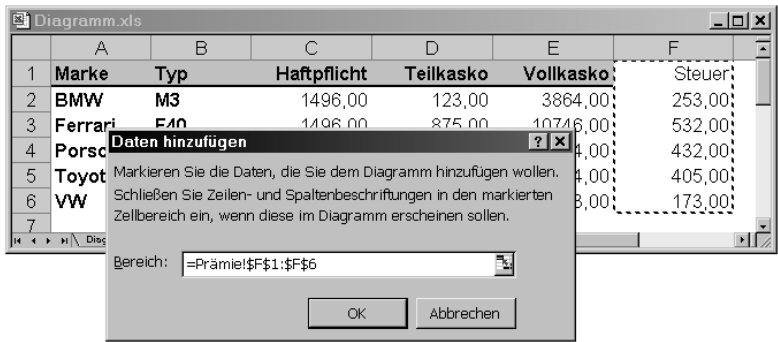
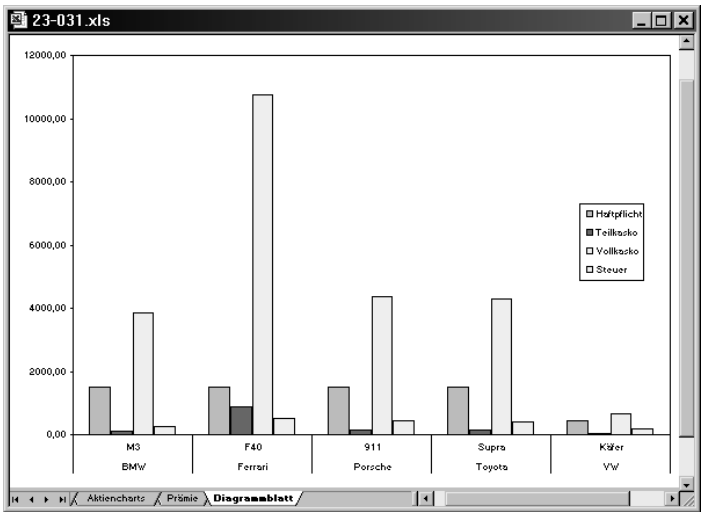


Abbildung 22.38:
Datenreihe einfügen



Analog dazu können Sie neue Rubriken einfügen, was in der Praxis wahrscheinlich häufiger vorkommt (Abbildung 22.39).

Ist Excel dabei unschlüssig, ob die selektierten Daten als zusätzliche Rubrik oder aber als zusätzliche Datenreihe zu interpretieren sind, erscheint folgendes Dialogfeld (Abbildung 22.40).

Im Beispiel lassen Sie alle vorselektierten Optionen unverändert: *Neue Datenpunkte*, weil die selektierten Daten als zusätzliche Rubrik und nicht als Datenreihe eingefügt werden sollen; *Spalten*, weil die Werte dieser Rubrik ja tatsächlich spalten- und nicht zeilenweise angeordnet sind.

Rubriken (X-Achsenbeschriftung) in erster Spalte ist jedoch problematisch, da sich die Rubrikenbeschriftung eigentlich in den ersten *beiden* Spalten befinden. Daher entfällt leider die in der zweiten Spalte enthaltene Rubrikenüberschrift »Astra«. Davon abgesehen wird das Diagramm jedoch wie gewünscht erweitert (Abbildung 22.41).

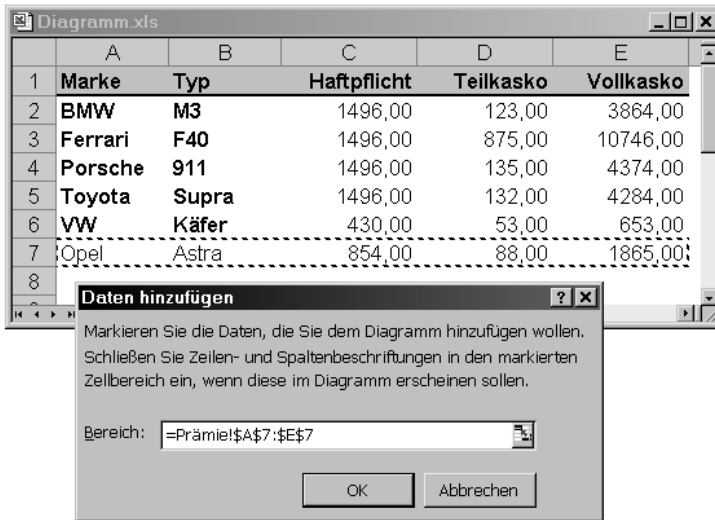


Abbildung 22.39:
Zusätzliche Rubrik

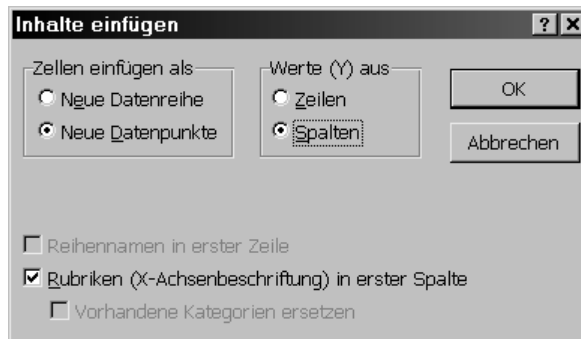


Abbildung 22.40:
Rubrik oder Datenreihe?

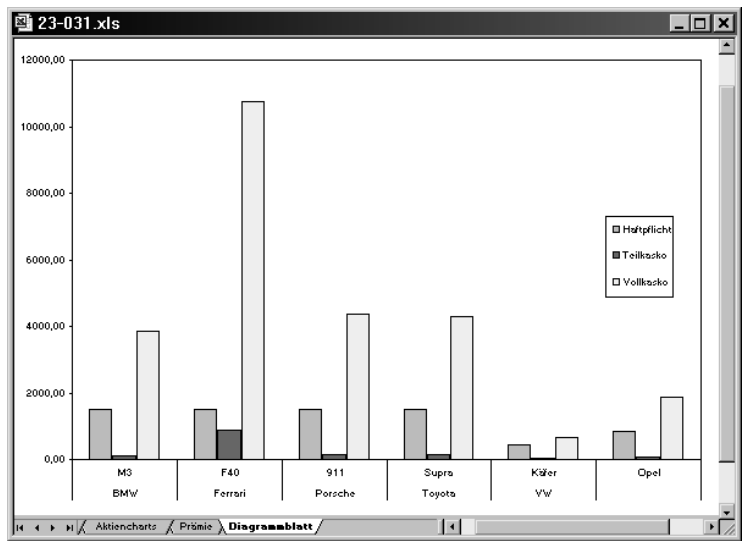
Die Zusatzfrage erschien nur, weil Excel im Beispiel vor ein sehr spezielles Problem gestellt wurde: Der nachträglich selektierte Bereich enthielt in den beiden linken Spalten Texte und rechts daneben drei Zahlen.

Wäre statt A7:E7 der nur eine Rubrikenbeschriftung enthaltende Bereich B7:E7 selektiert worden, wäre der Fall für Excel klar gewesen: Die linke Spalte enthielt die Rubrikenbeschriftung und die drei Zahlen rechts daneben waren zeilenweise angeordnete Werte der Rubrik.

Das Dialogfeld wäre nicht erschienen, sondern die zusätzliche Rubrik ohne Nachfrage korrekt eingefügt worden (abgesehen vom fehlenden Rubrikentext »Opel« in der nicht selektierten Zelle A7).



Abbildung 22.41:
Rubrik einfügen



23 Diagrammarten

Im folgenden Kapitel erläutere ich die verschiedenen Diagrammarten, die Ihnen Excel zur Verfügung stellt, und zeige Ihnen, wie Sie auch ohne den Assistenten nachträglich den Typ eines Diagramms äußerst bequem verändern können.

Jeder Diagrammtyp, beispielsweise das Säulendiagramm, steht in mehreren Varianten zur Verfügung. Daher erläutere ich zunächst die Eigenschaften der verschiedenen Varianten am Beispiel von Säulendiagrammen und bespreche anschließend die verschiedenen Diagrammtypen selbst: Balkendiagramme, Liniendiagramme, Netzdiagramme und so weiter.

Da fast alle Diagrammtypen auch in dreidimensionalen Varianten verfügbar sind, gehe ich anschließend auf diese »3D-Diagramme« ein und zeige beispielsweise, wie Sie die perspektivische Darstellung durch Rotieren des Diagramms um seine drei Achsen verändern können.

Anschließend bespreche ich die Möglichkeiten zur Feinmanipulation des Diagrammtyps. Sie müssen nämlich keineswegs immer dem gesamten Diagramm einen anderen Typ oder eine andere Variante zuweisen. Stattdessen können Sie auch Verbunddiagramme erzeugen, indem beispielsweise einige Datenreihen als Säulen dargestellt werden und andere wiederum als Linien, also mehrere Diagrammtypen in einem einzigen Diagramm kombiniert sind.

Zum Abschluss gehe ich noch auf Spezialdiagramme ein, die Kombinationen aus einem der Standard-Diagrammtypen in einer bestimmten Variante darstellen und darüber hinaus noch alle möglichen zusätzlichen Einstellungen enthalten: wo sich die Legende befinden soll, welche Schriftart für die Achsenbeschriftung verwendet wird, welche Farbe die Diagrammfläche besitzt etc.

Ich zeige Ihnen, wie Sie diesen Spezialdiagrammen Ihre eigenen fertig formatierten Diagramme als benutzerdefinierte Typen hinzufügen, um später einem neu zu erzeugenden Diagramm all Ihre Lieblingsformatierungen auf Knopfdruck zuzuweisen.

Sie können sogar Excels Standard-Diagrammtyp verändern, den der Assistent bei der Erzeugung eines neuen Diagramms als Vorlage verwendet und damit dafür sorgen, dass neue Diagramme automatisch entsprechend Ihren individuellen (Formatierungs)vorstellungen erzeugt werden!

23.1 Diagrammtypen und -varianten



Um aus einem Säulen- ein Flächen- oder ein Kreisdiagramm zu machen, klicken Sie auf das Diagramm und danach in der Diagramm-Symbolleiste auf das abgebildete Symbol (Abbildung 23.1).

Abbildung 23.1:
Diagramm-
Symbolleiste



Die zugehörige Liste enthält nahezu alle verfügbaren Diagrammarten. Klicken Sie eines davon an, so wird das Diagramm in den betreffenden Typ umgewandelt.



Die Kategorie Diagramm erstellen des Anpassen-Dialogfelds enthält noch einige weitere Symbole zur Umwandlung in verschiedene andere Diagrammtypen.

Die allgemeine Technik zum Ändern des Diagrammtyps besteht jedoch darin, das Diagramm durch Anklicken zu selektieren und danach den Diagramm-Assistenten zu aktivieren und im ersten Schritt einen anderen Typ oder eine andere Variante auszuwählen, oder den Befehl **DIAGRAMM|DIAGRAMMTYP...** zu wählen (nur verfügbar, wenn ein Diagrammelement selektiert ist), der das gleiche Dialogfeld öffnet.

Diagramm-varianten

Da die meisten Diagrammtypen in immer gleichen Varianten vorliegen, erläutere ich zunächst die für Säulendiagramme verfügbaren Varianten und beschränke mich bei den restlichen Diagrammtypen auf die Erläuterung jener Varianten, die sich davon unterscheiden (Abbildung 23.2).

Die sieben Varianten erläutere ich der Reihe nach von links nach rechts und von oben nach unten.



Die Standardvariante der Säulendiagramme kennen Sie bereits. Ein bestehendes Diagramm können Sie statt mit dem Assistenten auch per Anklicken des abgebildeten Symbols in ein Säulendiagramm dieser Art umwandeln.

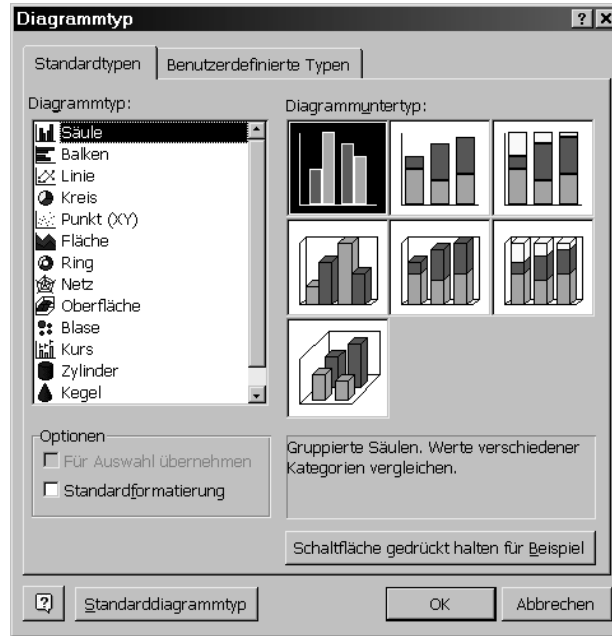


Abbildung 23.2:
Säulendiagramm-
Varianten

Die zweite Variante erzeugt gestapelte Säulen. Alle Säulen einer Rubrik werden übereinander gestapelt, sodass Sie die Gesamthöhen vergleichen können; im Beispiel sind das die für jeden Autotyp insgesamt zu zahlenden Prämien, wenn jeweils alle drei Versicherungsarten abgeschlossen werden (Abbildung 23.3).

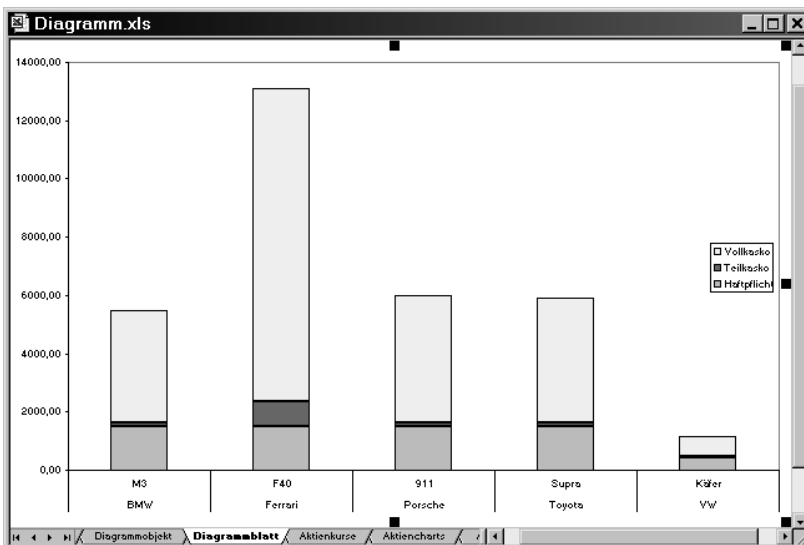
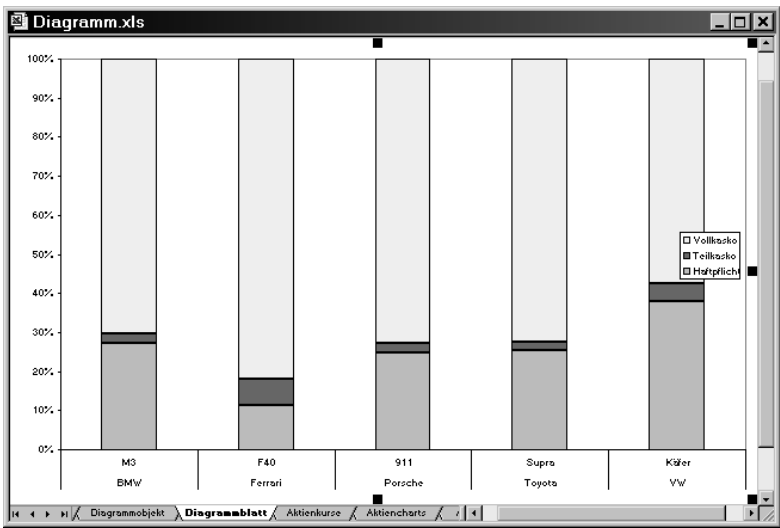


Abbildung 23.3:
Gestapelte Säulen

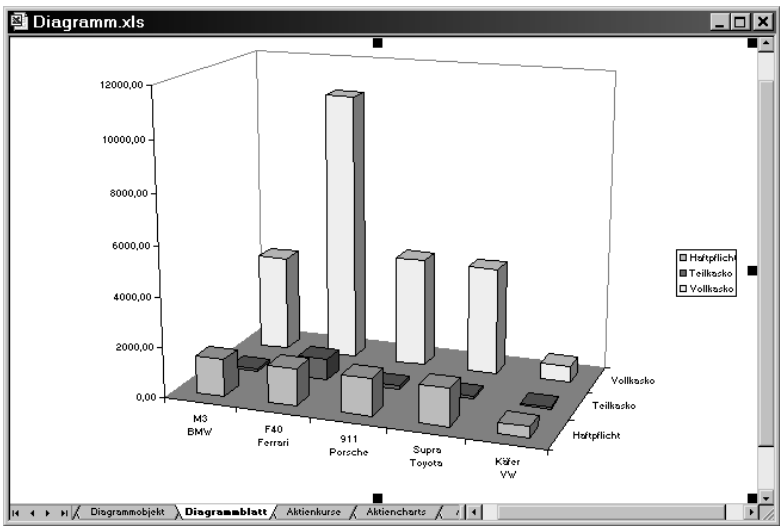
Die dritte Variante stapelt die Säulen einer Rubrik ebenfalls übereinander, erzeugt jedoch jeweils gleich hohe Säulen, die einen prozentualen Vergleich der einzelnen Werte anhand der anteiligen Höhen der verschiedenen Säulenabschnitte ermöglichen (Abbildung 23.4).

Abbildung 23.4:
Prozentual
gestapelte Säulen



Die folgenden Varianten entsprechen den ersten drei, sind jedoch 3D-Varianten, die einen perspektivischen Eindruck und damit einen Hauch von Tiefenwirkung erzeugen. Davon abgesehen entsprechen die vierte, fünfte und sechste Variante jedoch exakt den vorhergehenden. Am ausgeprägtesten ist die Tiefenwirkung bei der letzten 3D-Variante, bei der die Diagrammfläche zusätzlich gedreht ist (Abbildung 23.5).

Abbildung 23.5:
3D-Variante



Alle verfügbaren Diagrammtypen erläutere ich im Folgenden nach dem gleichen Schema: zuerst sehen Sie ein Bild, das die für den betreffenden Typ verfügbaren Varianten darstellt, danach ein Beispiel für ein derartiges Diagramm, wobei ich üblicherweise die erste Variante verwende.

Diagrammtypen

Säulendiagramme kennen Sie inzwischen zur Genüge, also beschäftigen wir uns gleich mit den eng damit verwandten Balkendiagrammen (Abbildung 23.6).

Balkendiagramme



Abbildung 23.6:
Balkendiagramm-
Varianten

Im Gegensatz zu Säulendiagrammen sind die Balken nicht horizontal, sondern vertikal ausgerichtet. Zusätzlich sind Rubriken- und Größenachse vertauscht (Abbildung 23.7).

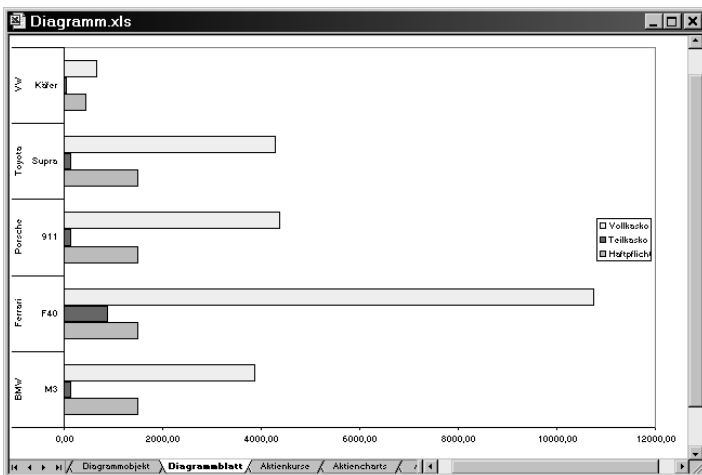
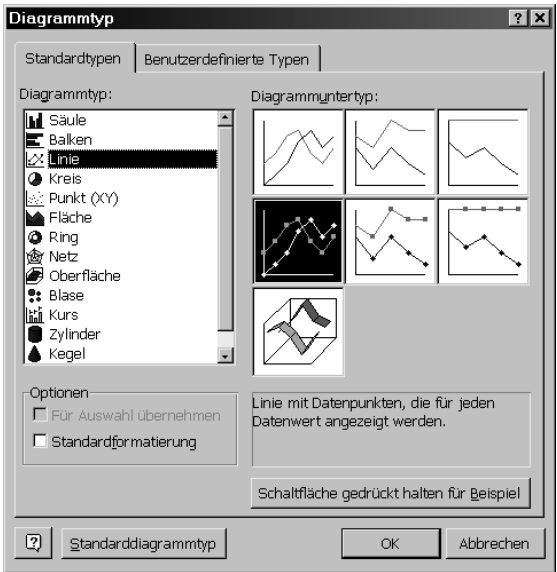


Abbildung 23.7:
Balkendiagramm

Liniendiagramme Liniendiagramme sind zur Darstellung von Daten geeignet, die sich nicht sprunghaft, sondern eher kontinuierlich in kleinen Schritten ändern (Abbildung 23.8).

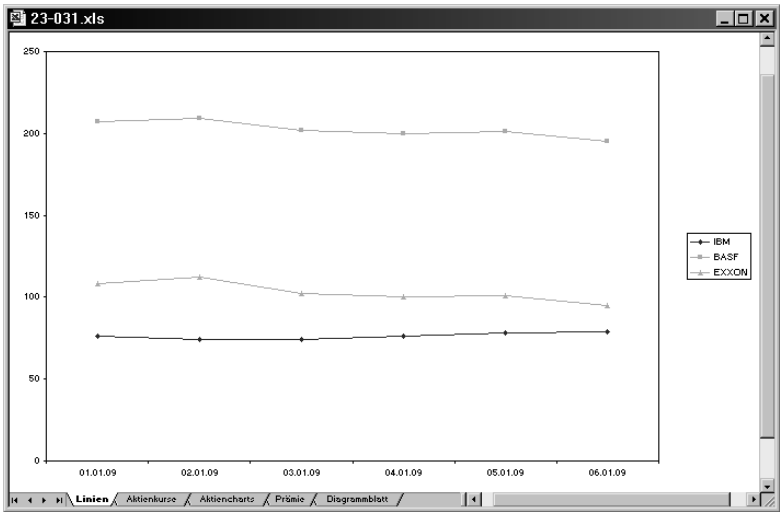
Abbildung 23.8:
Liniendiagramm-
Varianten



Das gilt zum Beispiel für Aktiencharts, evtl. um die über jeweils sechs Tage verfolgte Kurse dreier Aktien darzustellen, also dreier Datenreihen.

Mit der ersten Variante wird jede der drei Datenreihen als einzelne Linie dargestellt, wobei die einzelnen Datenpunkte durch eine Gerade miteinander verbunden sind (Abbildung 23.9).

Abbildung 23.9:
Aktienkurse als
Liniendiagramm



Flächendiagramme sind eng mit Liniendiagrammen verwandt (Abbildung 23.10).

*Flächen-
diagramme*

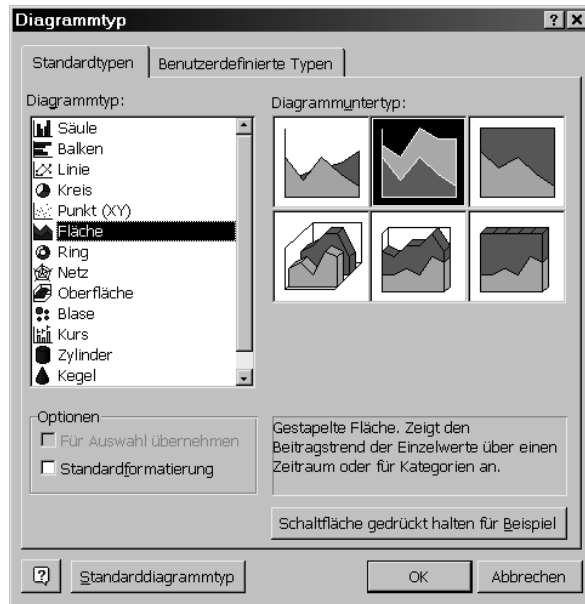


Abbildung 23.10:
Flächendiagramm-
Varianten

Der einzige Unterschied besteht darin, dass die Flächen zwischen den einzelnen Linien ausgefüllt sind (Abbildung 23.11).

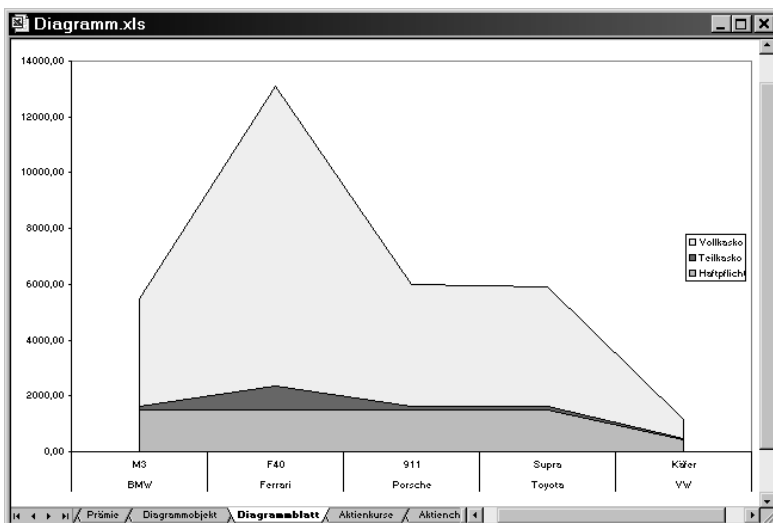


Abbildung 23.11:
Flächendiagramm

Kreisdiagramme stellen eine der am gängigsten Methoden dar, um *eine* Datenreihe grafisch darzustellen (Abbildung 23.12).

Kreisdiagramme

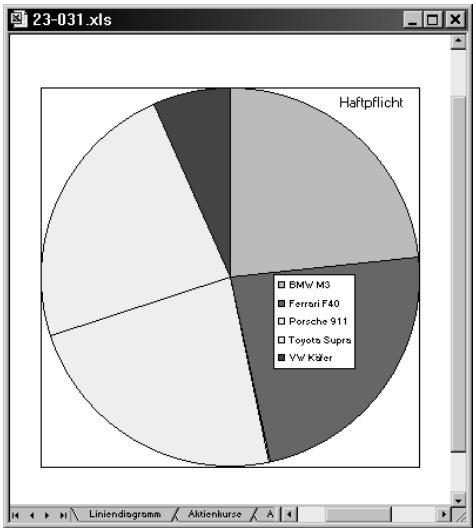
Abbildung 23.12:
Kreisdigramm-
Varianten



Jedes Kreissegment entspricht genau einem Datenpunkt der Reihe. Besteht die Datenbasis aus mehreren Datenreihen, verwendet Excel nur die *erste Datenreihe* zum Aufbau der Grafik. Kreisdigramme sind daher für die gleichzeitige Darstellung mehrerer Wertereihen ungeeignet.

Deutlich wird das bei einer Umwandlung des Versicherungsprämienbeispiels in ein Kreisdigramm, beispielsweise mit dem Listensymbol der Diagramm-Leiste. Wählen Sie aus der Liste das *Kreisdigramm*-Symbol aus, so wird die folgende Variante erzeugt, in der nur die erste Datenreihe dargestellt wird, »Haftpflcht« (Abbildung 23.13).

Abbildung 23.13:
Kreisdigramm





Sie können einzelne Kreissegmente durch Herausziehen aus dem Gesamtkreis hervorheben, indem Sie auf den Kreis klicken, wodurch zunächst alle Segmente selektiert werden, und danach an einem der Anfasser eines Segments ziehen.

Ein einzelnes Segment selektieren Sie, indem Sie nach der Selektion aller Segmente eine kleine Pause einlegen, um einen Doppelklick zu vermeiden, und danach auf das betreffende Segment klicken. Anschließend können Sie dieses einzelne Segment verschieben.

Mit diesen Techniken können Sie Kreise ähnlich formatieren wie mit der vierten oder der fünften Kreisvariante, allerdings weitaus individueller.



Die dritte (und die letzte) Variante ermöglichen Ihnen außer dem Hauptkreis einen damit verbundenen Unterkreis (bzw. gestapelte Balken) darzustellen. Diese Variante ist vorteilhaft, wenn verschiedene Kreissegmente im Hauptkreis zu schmal sind, um darin deutlich erkennbar zu sein.

Nach der Diagrammerstellung können Sie wählen, welche Segmente in welchem Kreis dargestellt werden, indem Sie mit den zuvor beschriebenen Techniken einzelne Segmente selektieren und vom einen in den anderen Kreis ziehen.

Ringdiagramme basieren auf konzentrischen Ringen um einen Kreismittelpunkt herum. Der innerste Ring stellt die erste Datenreihe dar, der zweite Ring von innen die zweite Datenreihe und so weiter (Abbildung 23.14).

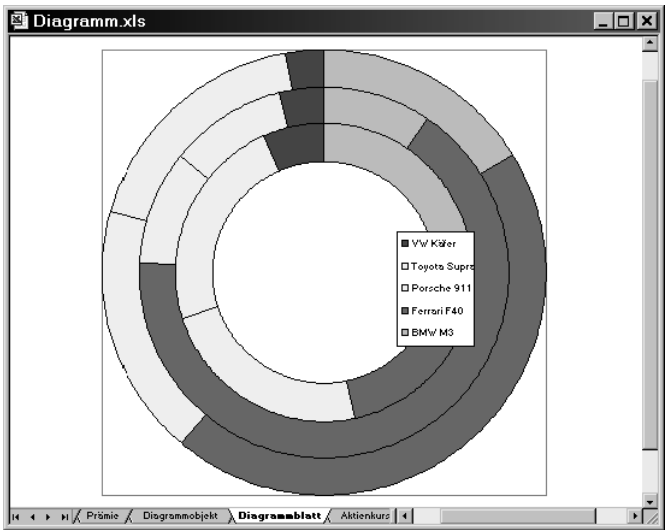
Ringdiagramme



Abbildung 23.14:
Ringdiagramm-
Varianten

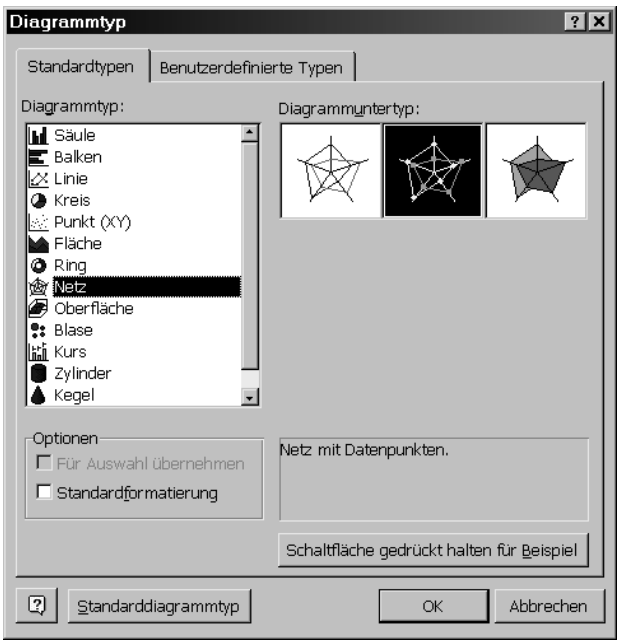
Die Länge eines Ringsegments ist dabei ein prozentuales Maß für die Größe des zugehörigen Werts, verglichen mit den restlichen Werten der Reihe, also den restlichen Segmenten des Rings (Abbildung 23.15).

Abbildung 23.15:
Ringdiagramm



Netzdiagramme In Netzdiagrammen wird die Größe eines Werts durch seinen Abstand zum Kreismittelpunkt dargestellt. Je größer der Wert ist, desto weiter außen befindet er sich (Abbildung 23.16).

Abbildung 23.16:
Netzdiagramm-
Varianten



Hier sehen Sie ein Netzdiagramm der ersten Variante (Abbildung 23.17):

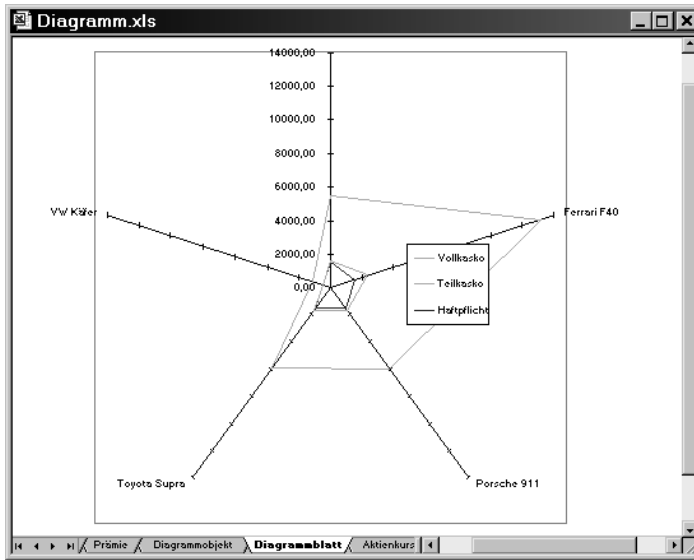


Abbildung 23.17:
Netzdiagramm

Punktdiagramme sind vor allem für statistische Zwecke interessant, um zum Beispiel Streuungen um einen Mittelwert herum optisch darzustellen (Abbildung 23.18).

Punktdiagramme

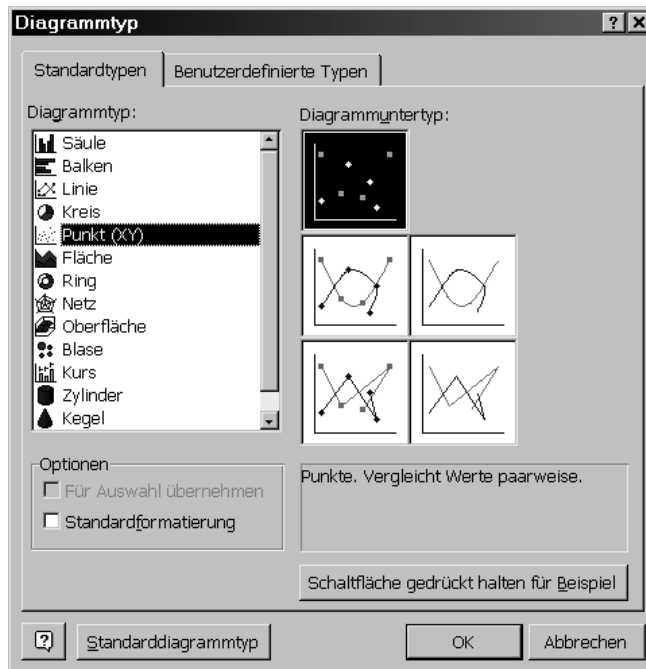
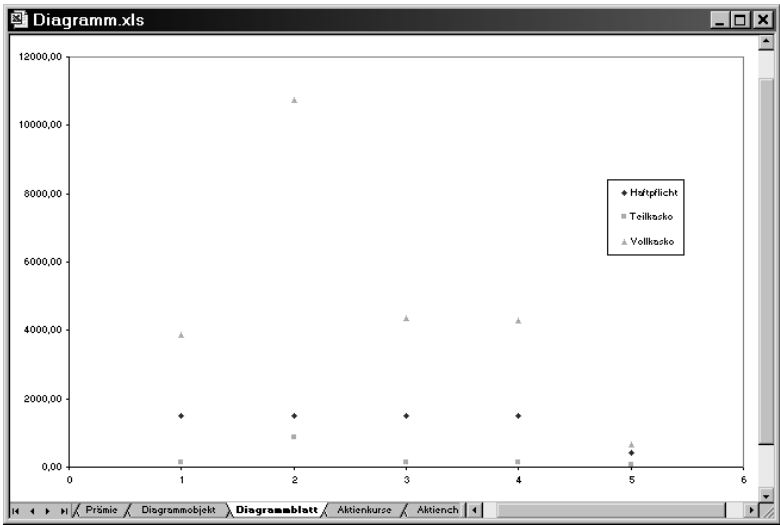


Abbildung 23.18:
Punktdiagramm-
Varianten

Das ist ein Punktdiagramm. In der ersten Variante (Abbildung 23.19):

Abbildung 23.19:
Punktdiagramm



Oberflächen-
diagramm Oberflächendiagramme setzen mindestens zwei Datenreihen voraus (Abbildung 23.20).

Abbildung 23.20:
Oberflächen-
diagramm-
Varianten



In der ersten Variante (Abbildung 23.21) sieht das so aus:

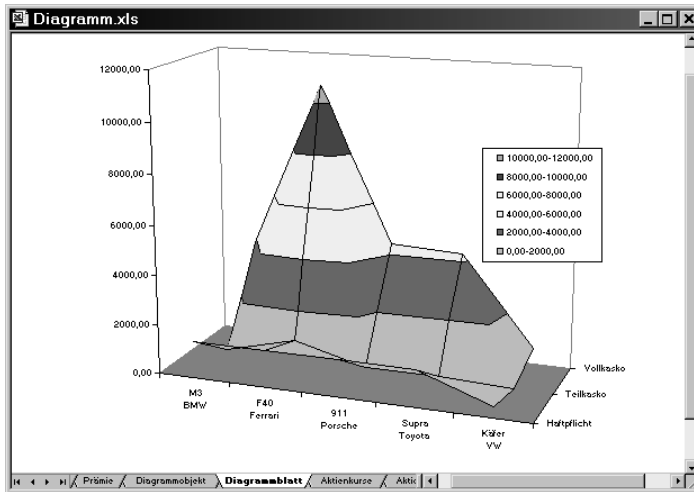


Abbildung 23.21:
Oberflächen-
diagramm

Blasendiagramme fassen jeweils zwei Werte zu einem einzigen Datenpunkt zusammen.

Blasen

Zur Erzeugung selektieren Sie zwei Datenreihen. Die erste Reihe bestimmt die Höhe der betreffenden Blase an der Größenachse, der Wert aus der zweiten Reihe legt den Durchmesser der Blase fest (Abbildung 23.22):

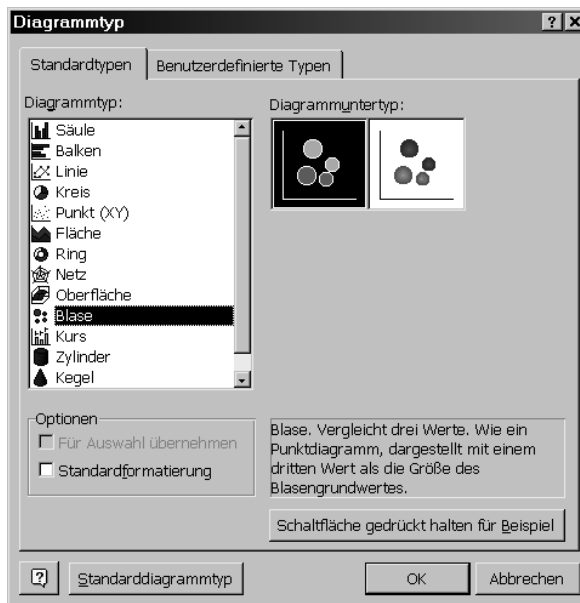
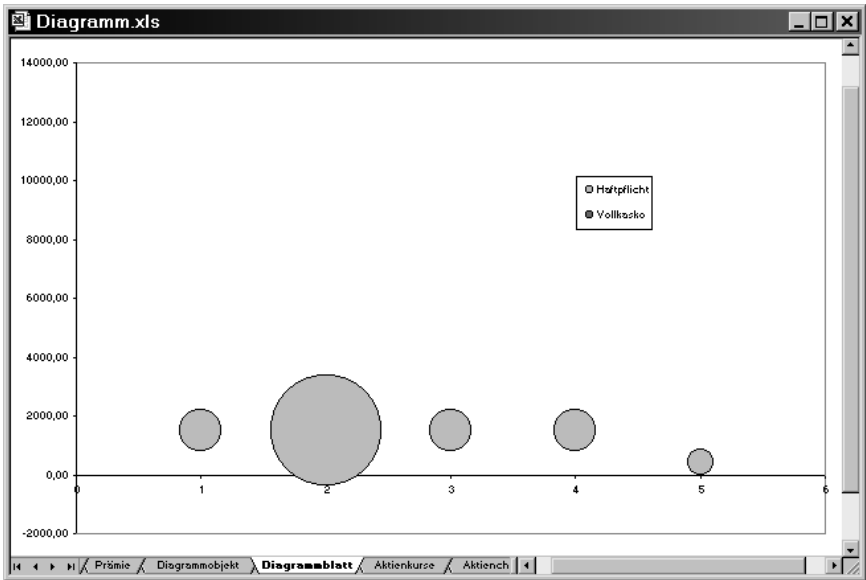


Abbildung 23.22:
Blasendiagramm-
Varianten

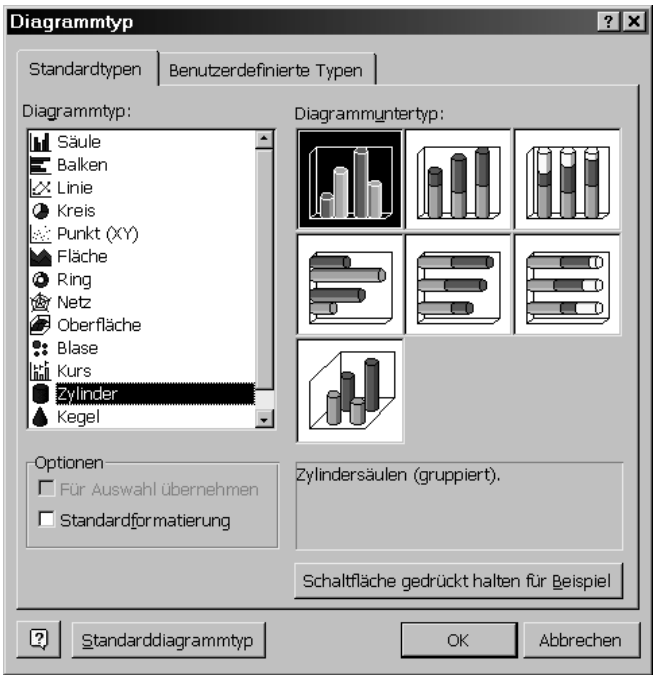
In der ersten Variante (Abbildung 23.23) sieht das so aus:

Abbildung 23.23:
Blasendiagramm



Zylinder Zylinderdiagramme entsprechen Säulendiagrammen, abgesehen davon, dass die Säulen die Form von Zylindern besitzen (Abbildung 23.24):

Abbildung 23.24:
Zylinderdiagramm-
Varianten



Hier sehen Sie die erste Variante (Abbildung 23.25):

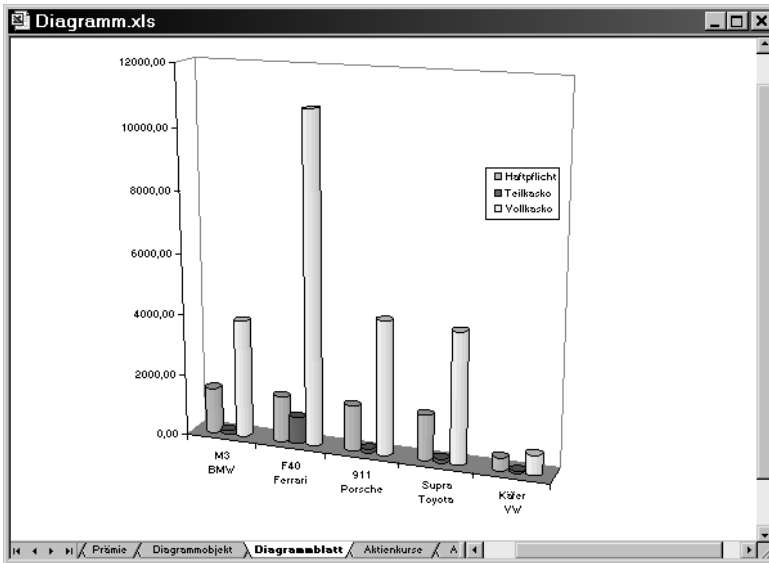


Abbildung 23.25:
Zylinderdiagramm

Das gleiche gilt für Kegeldiagramme, nur dass die Säulen in diesem Fall nach oben hin spitz zulaufen (Abbildung 23.26):

Kegel

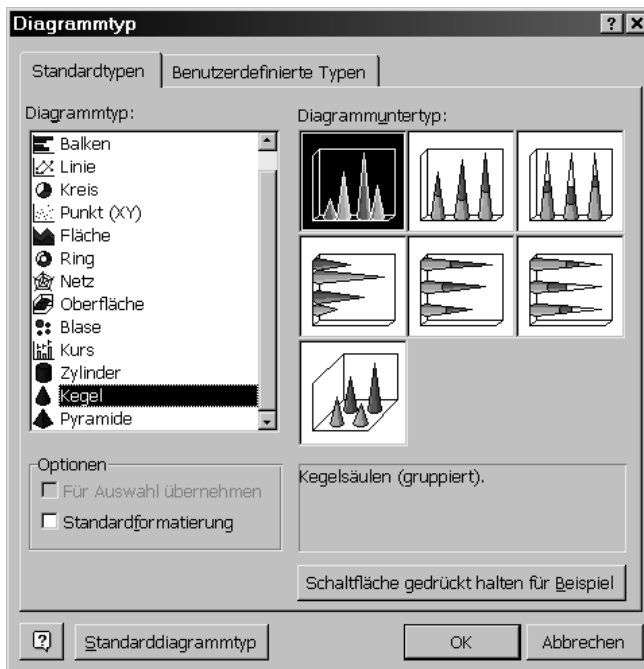
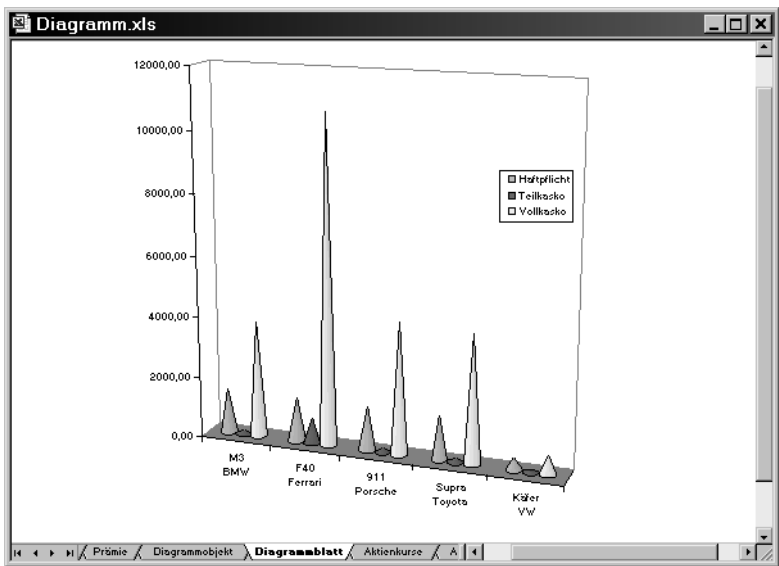


Abbildung 23.26:
Kegelvarianten

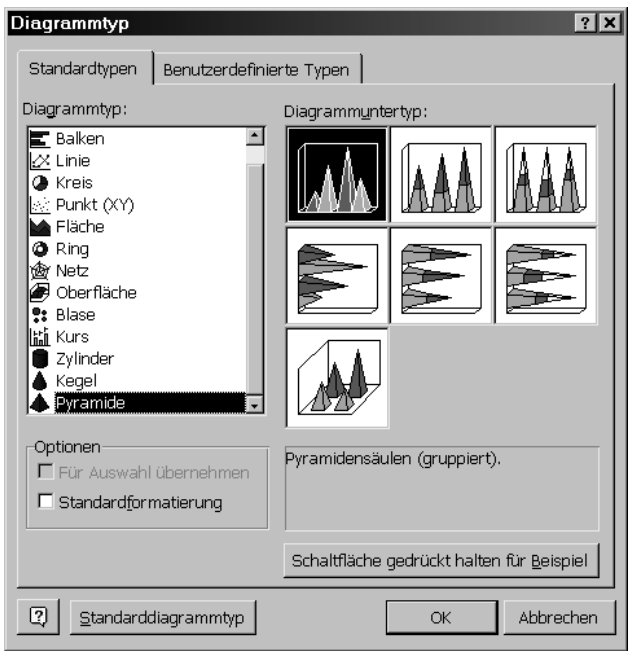
In der ersten Variante (Abbildung 23.27) sieht das so aus:

Abbildung 23.27:
Kegeldiagramm



Pyramiden Pyramidendiagramme unterscheiden sich von Kegeldiagrammen durch die dreieckige Form der Säulen (Abbildung 23.28):

Abbildung 23.28:
Pyramiden-
diagramm-
Varianten



Die erste Variante (Abbildung 23.29) sehen Sie hier:

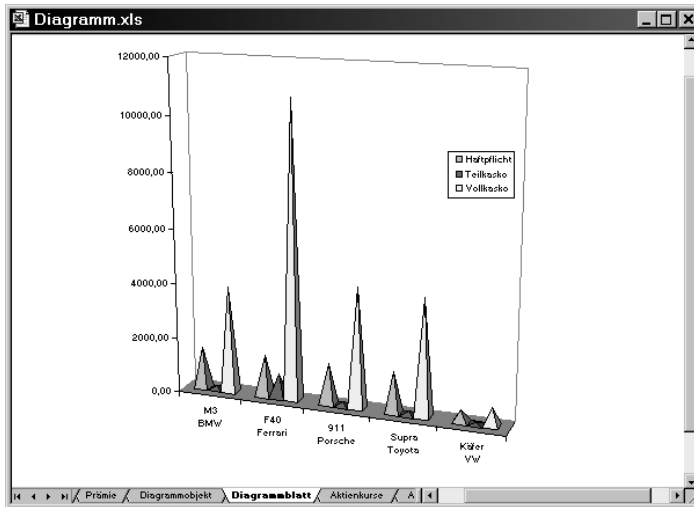


Abbildung 23.29:
Pyramiden-
diagramm

Kursdiagramme bestehen aus einer Überlagerung mehrerer Diagrammtypen (Abbildung 23.30).

Kurse

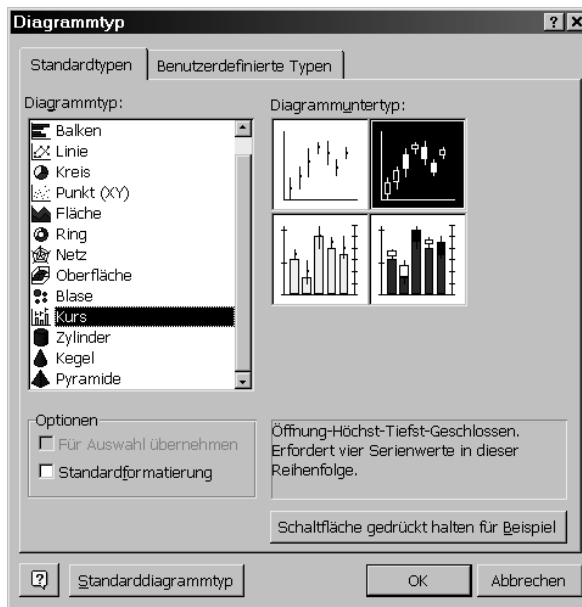


Abbildung 23.30:
Kursdiagramm-
Varianten

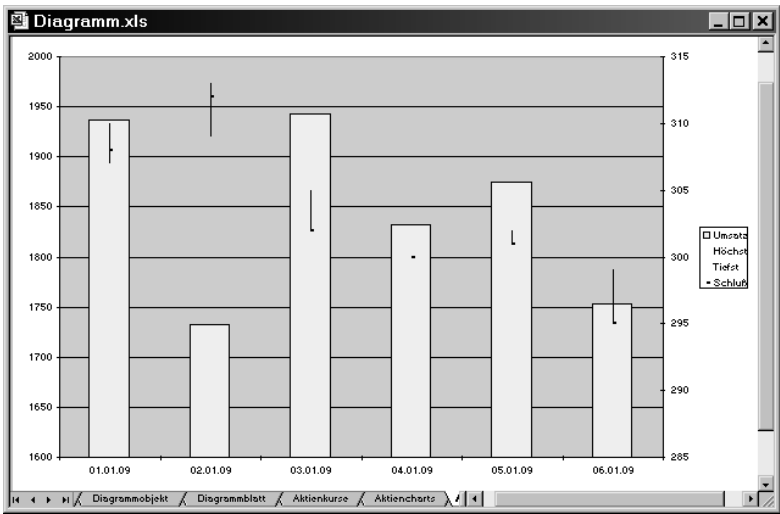
Am interessantesten speziell für Aktienkurse ist die dritte Variante. Sie setzt eine Tabelle mit einem Bereich wie F1:J7 in der folgenden Abbildung voraus, also einen Bereich, der vier Datenreihen mit dem Umsatz, den Höchst-, Tiefst- und den Schlusskursen einer Aktie enthält (Abbildung 23.31).

Abbildung 23.31:
Diagramm.xls,
»Aktienkurse«

	A	B	C	D	E	F	G	H	I	J
1	Datum	IBM	BASF	EXXON		Datum	Umsatz	Höchst	Tiefst	Schluß
2	01.01.09	76	207	108		01.01.09	1937	310	307	308
3	02.01.09	74	209	112		02.01.09	1732	313	309	312
4	03.01.09	74	202	102		03.01.09	1943	305	302	302
5	04.01.09	76	200	100		04.01.09	1832	300	300	300
6	05.01.09	78	201	101		05.01.09	1875	302	301	301
7	06.01.09	79	195	95		06.01.09	1753	299	295	295
8										
9										
10										
11		Datum	Umsatz	Eröffng.	Höchst	Tiefst	Schluß			
12		01.01.09	1937	307	310	307	308			
13		02.01.09	1732	310	313	309	312			
14		03.01.09	1943	305	305	302	302			
15		04.01.09	1832	300	300	300	300			
16		05.01.09	1875	301	302	301	301			
17		06.01.09	1753	297	299	295	295			

Daraus erzeugt sie zum einen Säulen, die dem Umsatz an den einzelnen Tagen entsprechen, und vertikalen Linien, die die Kursspannweiten darstellen. Das untere Linienende ist der Tiefstkurs am betreffenden Tag, das obere Ende der Höchstkurs und der Punkt dazwischen der Schlusskurs (Abbildung 23.32).

Abbildung 23.32:
Umsatz, Höchst-,
Tiefst- und Schluss-
kurse einer Aktie



Die letzte Variante ist für Tabellen gedacht, die wie der Bereich B11:G17 in Bild 23.31 zusätzliche Eröffnungskurse enthalten (Abbildung 23.33).

Die Rechtecke kennzeichnen die Spannweite zwischen Eröffnungs- und Schlusskurs, die vertikalen Linien eventuell darüber oder darunter hinausgehende Höchst- bzw. Tiefstkurse.

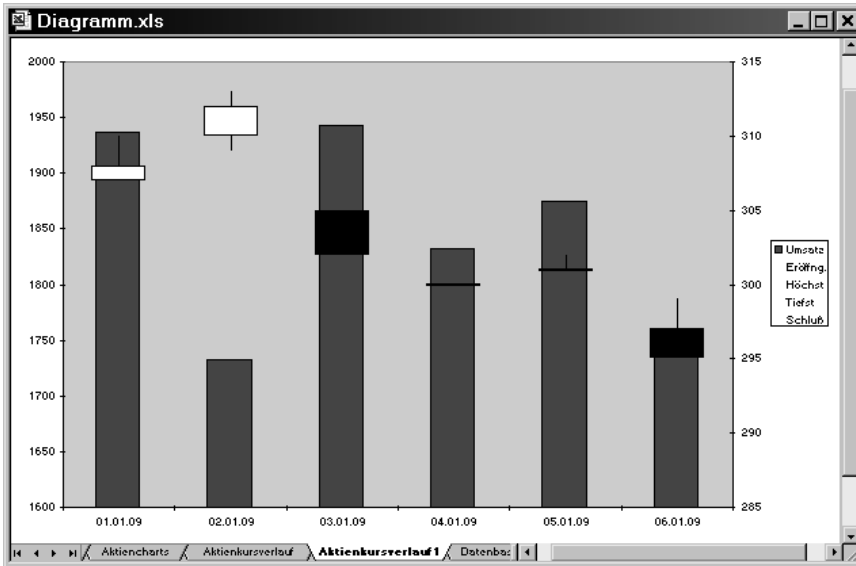


Abbildung 23.33:
Umsatz,
Eröffnungs-,
Höchst-, Tiefst- und
Schlusskurse

23.2 3D-Diagramme

Fast alle Diagrammtypen sind auch in dreidimensionalen Varianten verfügbar (Abbildung 23.34).

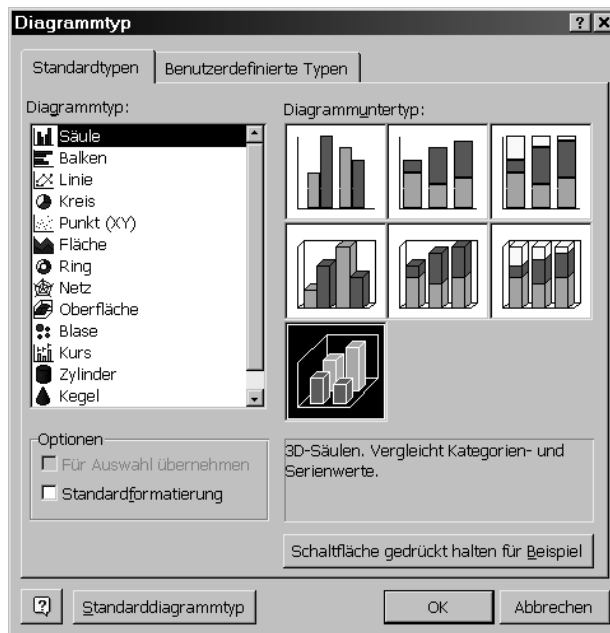


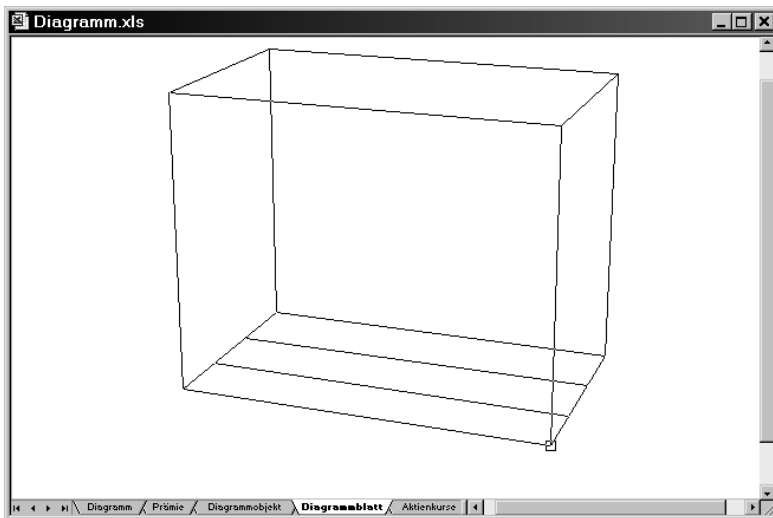
Abbildung 23.34:
3D-Säulendiagramm-Varianten

**Perspektiven-
kontrolle**

Bei all diesen Varianten können Sie die Perspektiven verändern: Sie selektieren die Diagrammecken, indem Sie eine Ecke der Diagrammfläche anklicken. An den Ecken erscheinen nun Anfasser.

Befindet sich der Cursor über einem davon, erscheint daneben der Hinweis »Ecken« und der Cursor wird zum Fadenkreuz. Durch Ziehen am Anfasser können Sie nun das gesamte Diagramm um eine Rotationsachse drehen (Abbildung 23.35).

Abbildung 23.35:
Rotieren eines
Drahtmodells



Während der Drehung erscheint nur ein Drahtmodell ohne Inhalt, das möglicherweise verwirrend ist. Einen realistischeren Überblick erhalten Sie, wenn Sie während der Rotation die **[Strg]**-Taste gedrückt halten (Abbildung 23.36).

Eine umständlichere Möglichkeit zur Perspektivenänderung bietet der Befehl **DIAGRAMM|3D-ANSICHT...** (nur verfügbar, wenn ein Diagrammelement selektiert ist).

Sein Dialogfeld enthält drei Paare von Richtungsknöpfen, die durch mehrfaches Anklicken beliebige Rotationen um drei rechtwinklig aufeinander stehende Achsen in jeweils entgegengesetzte Richtungen ermöglichen (Abbildung 23.37).

Die beiden linken Knöpfe kippen das Diagramm und die beiden unteren drehen es. Die beiden rechten Knöpfe bestimmen, wie ausgeprägt die Achsen fluchten (auf einen gemeinsamen Punkt zulaufen), wie stark der perspektivische Eindruck ausgeprägt ist, d.h. ob er eher gemäßigt wirkt, entsprechend dem Blick durch ein Teleobjektiv, oder übertrieben wie durch ein extremes Weitwinkelobjektiv.

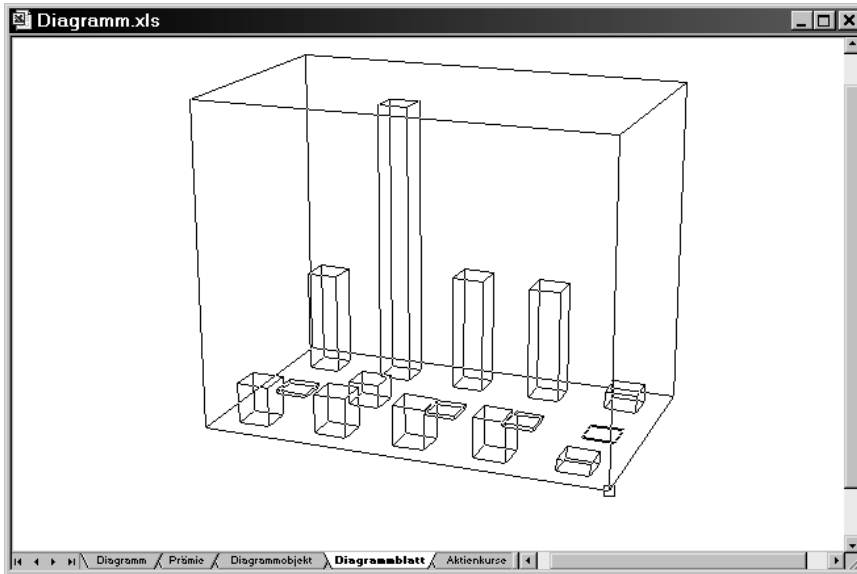


Abbildung 23.36:
Rotieren des
Diagramms

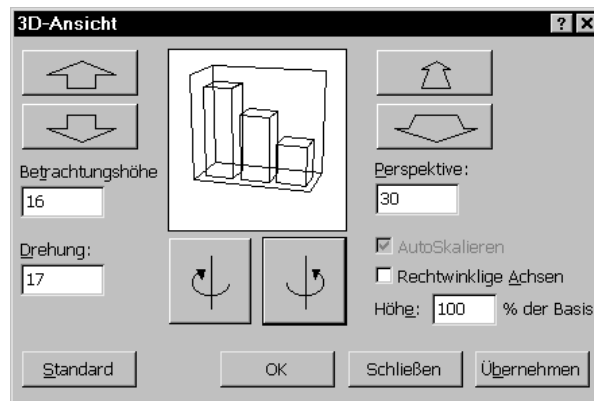


Abbildung 23.37:
Perspektiven-
kontrolle

Jede Manipulation führt zur Aktualisierung der zugehörigen Werte für *Betrachtungshöhe*, *Drehung* und *Perspektive*.

- ➔ *Übernehmen* überträgt die aktuelle Einstellung auf das Diagramm, ohne das Dialogfeld zu schließen, und erlaubt damit eine direkte Überprüfung der momentan definierten Perspektive.
- ➔ *Standard* stellt die Standardeinstellungen wieder her und macht Ihre Manipulationen rückgängig.
- ➔ *Höhe* definiert das Verhältnis zwischen Breite und Höhe der Grafik in Prozenten. Zum Beispiel bedeutet »50«, dass die Grafik halb so hoch wie breit ist.

- ➔ *Rechtwinklige Achsen* unterdrückt das perspektivische Fluchten der Achsen, was recht nützlich ist, wenn das Diagramm durch vorhergehendes Rotieren bereits recht verzerrt ist und wieder entzerrt werden soll.
- ➔ Zur Option *AutoSkalieren*: Wandeln Sie ein 2D- in ein 3D-Diagramm um, so ist es im Verhältnis dazu recht klein. Ist *AutoSkalieren* aktiviert (zusätzlich muss auch *Rechtwinklige Achsen* aktiviert sein), wird es bei einer Drehung von weniger als 45 Grad so skaliert, dass seine Größe etwa dem des vorhergehenden 2D-Diagramms entspricht.

23.3 Verbunddiagramme

Datenreihe selektieren

Statt dem gesamten Diagramm einen anderen Typ oder eine andere Variante zuzuweisen, können Sie selektiv eine oder mehrere Datenreihen formatieren und ihnen einen anderen Diagrammtyp zuweisen als den restlichen Reihen.

Dazu müssen Sie jedoch zunächst die betreffende Datenreihe des Diagramms selektieren:

- ➔ Mit der Maus klicken Sie irgendeinen Datenpunkt der betreffenden Reihe an, zum Beispiel eine Säule der Reihe »Vollkasko«, wenn Sie diese Reihe formatieren wollen.
- ➔ Mit der Tastatur drücken Sie einfach so oft eine der Pfeiltasten, bis die betreffende Reihe selektiert ist.

In beiden Fällen erkennen Sie die Selektion an kleinen Kästchen in den Säulen und daran, dass in der Bearbeitungsleiste nun die *DATENREIHE*-Formel zu sehen ist, die die Wertebasis dieser Säulen festlegt (Abbildung 23.38).



Anschließend klicken Sie in der Diagramm-Symbolleiste auf das *Diagrammtyp*-Symbol und wählen den gewünschten Diagrammtyp aus oder aber Sie wählen *DIAGRAMM|DIAGRAMMTYP...* (Abbildung 23.39).

Sie wählen wie gewohnt den gewünschten Typ und die Variante aus, aktivieren aber das erst jetzt aktivierbare Kontrollkästchen *Für Auswahl übernehmen*, das bewirkt, dass nur die zuvor ausgewählte Datenreihe entsprechend formatiert wird.

Selektieren Sie beispielsweise im Demodiagramm die Reihe »Vollkasko« und wählen Sie als Typ ein Kreisdiagramm, so wird nur die Vollkasko-Reihe als Kreisdiagramm dargestellt. Die beiden anderen Datenreihen erscheinen weiterhin als Säulendiagramm (Abbildung 23.40).

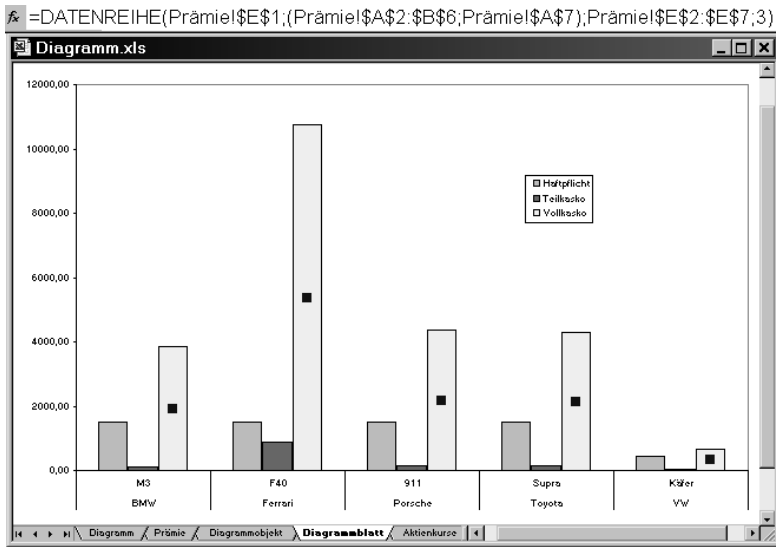


Abbildung 23.38:
Datenreihe
selektieren

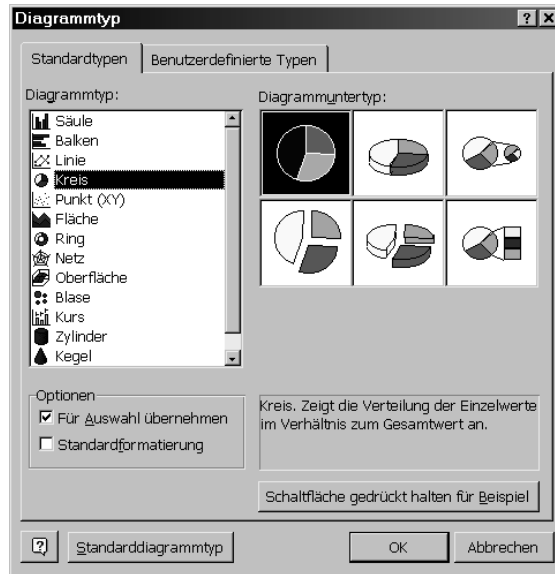


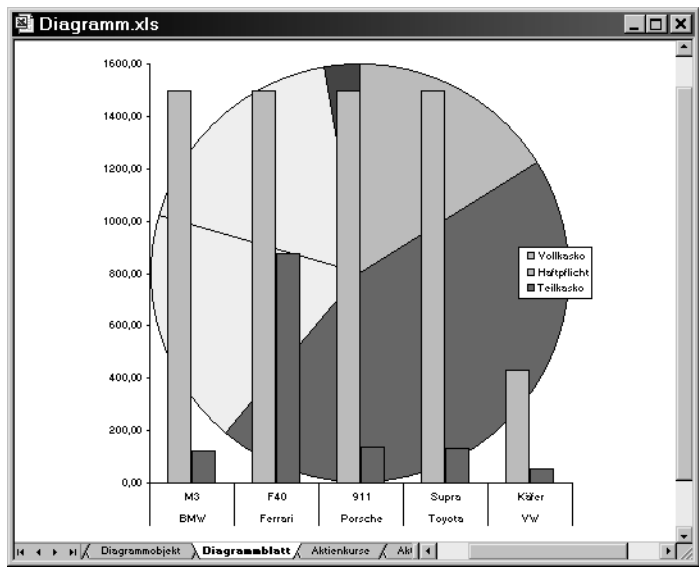
Abbildung 23.39:
Diagrammtyp
formatieren

Sie haben nun ein Verbunddiagramm erzeugt, einen Verbund aus zwei unterschiedlichen Diagrammen, die auf der gleichen Diagrammfläche dargestellt werden.

3D-Diagramme lassen sich nicht mit anderen Diagrammtypen kombinieren. Wählen Sie für die betreffende Datenreihe einen 3D-Typ, kann Excel daher – auf Nachfrage – nur das gesamte Diagramm entsprechend formatieren.



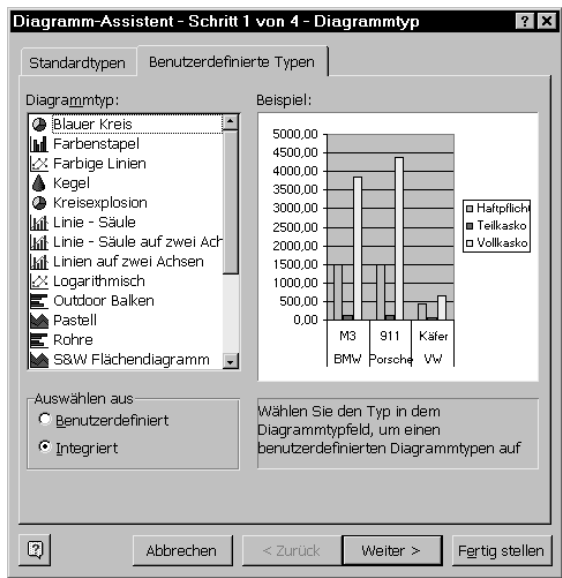
Abbildung 23.40:
Verbunddiagramm



23.4 Benutzerdefinierte Diagrammtypen und der Standard-Diagrammtyp

Außer den Standard-Diagrammtypen verfügt Excel über eine ganze Menge weitaus individuellerer Diagrammarten. Aktivieren Sie bitte den Diagramm-Assistenten und schlagen Sie das Register *Benutzerdefinierte Typen* auf (Abbildung 23.41).

Abbildung 23.41:
Benutzerdefinierte
Typen



Ist die Option *Integriert* aktiviert, werden alle verfügbaren Spezialdiagramme angezeigt. Diese Diagramme sind Kombinationen aus einem der Standard-Diagrammtypen in einer bestimmten Variante.

Darüber hinaus sind darin jedoch alle möglichen zusätzlichen Einstellungen enthalten: wo sich die Legende befinden soll, welche Schriftart für die Achsenbeschriftung verwendet wird, welche Farbe die Diagrammfläche besitzt und vieles mehr.

Kurz: Zu einem solchen vorgegebenen Diagramm gehören alle möglichen mehr oder weniger kunstvollen Spezialformatierungen, die Sie auch mit dem Diagramm-Assistenten festlegen können oder mit den in Kapitel 24, »Diagramme formatieren«, beschriebenen Techniken. Wählen Sie einen dieser Diagrammtypen aus, werden Ihrem Diagramm all diese Formatierungen gleichzeitig zugewiesen.



Zusätzlich können Sie mithilfe dieses Registers sogar eigene fertig formatierte Diagramme als benutzerdefinierte Typen speichern und später auf Knopfdruck einem neu zu erzeugenden Diagramm Ihre Lieblingsformatierungen zuweisen.

Diese benutzerdefinierten Diagramme werden angezeigt, wenn Sie die Option *Benutzerdefiniert* aktivieren. Wählen Sie aus der zugehörigen Liste einen von Ihnen definierten Diagrammtyp aus, so wird er anschließend dem momentan aktiven Diagramm zugewiesen.

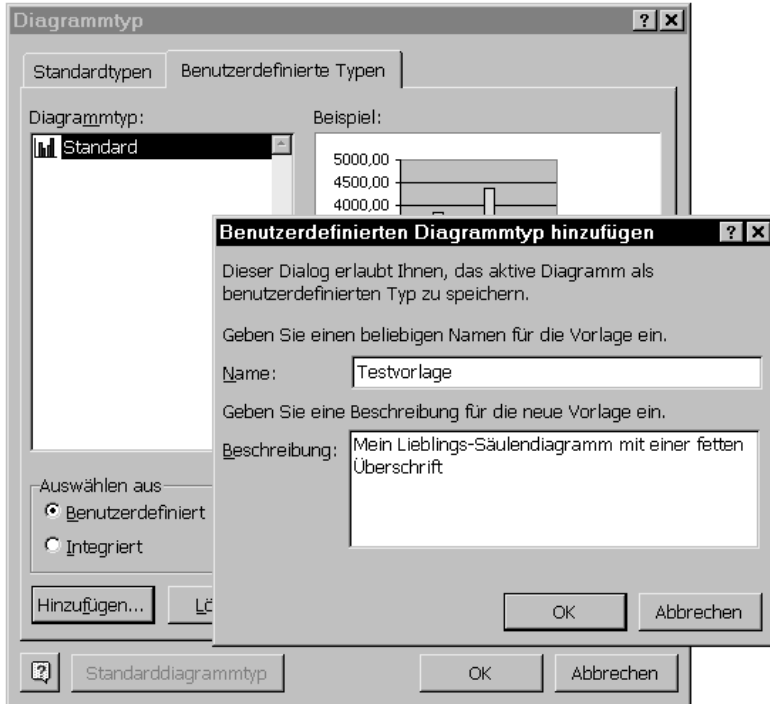
Die Liste enthält zunächst jedoch nur einen einzigen Diagrammtyp, »Standard«, den Standard-Diagrammtyp von Excel, der den Vorgaben bei der Erzeugung eines neuen Diagramms entspricht und den Sie übrigens wie gleich beschrieben verändern können, um diese Vorgaben selbst festzulegen.

Um unter Ihren eigenen Lieblingsdiagrammtypen auswählen zu können, müssen Sie zunächst einmal ein oder mehrere Diagramme gestalten und der Liste als benutzerdefinierte Diagrammtypen hinzufügen.

Haben Sie ein Diagramm erzeugt und formatiert und finden Sie es besonders gelungen, müssen Sie dazu dieses Dialogfeld öffnen – allerdings nicht durch Aktivierung des Diagramm-Assistenten, sondern mit dem Befehl **DIAGRAMM|DIAGRAMMTYP...** (erst nach Selektion eines Diagrammelements verfügbar).

Nur dann enthält das Register *Benutzerdefinierte Typen* die zusätzlichen Schaltflächen *Hinzufügen...* und *Löschen*, wenn darin die Option *Benutzerdefiniert* selektiert ist (Abbildung 23.42).

Abbildung 23.42:
Benutzerdefinierten
Typ hinzufügen



Hinzufügen... öffnet ein Dialogfeld, in dem Sie dem hinzuzufügenden Diagrammtyp einen Namen wie »Testvorlage« geben und zusätzlich einen längeren Kommentar wie »Mein Lieblings-Säulendiagramm mit fetter Überschrift« eingeben können. *OK* fügt in die Liste der benutzerdefinierten Diagrammtypen entsprechend den Eintrag »Testvorlage« ein.

Ab jetzt können Sie beim Formatieren eines Diagramms jederzeit in diesem Dialogfeld das Register *Benutzerdefinierte Typen* aufschlagen und die Option *Benutzerdefiniert* aktivieren, um in der zugehörigen Liste Ihren Lieblingsdiagrammtyp »Testvorlage« oder einen anderen von Ihnen definierten Diagrammtyp auszuwählen und dem aktiven Diagramm alle zugehörigen Formatmerkmale zuzuweisen.

Wurde das Dialogfeld nicht mit dem Diagramm-Assistenten, sondern mit dem Befehl *DIAGRAMM|DIAGRAMMTYP...* geöffnet, enthalten beide Register zusätzlich die Schaltfläche *Standard Diagrammtyp*.

Klicken Sie darauf, so wird der momentan in der Liste der Standard- oder der benutzerdefinierten Diagrammtypen selektierte Diagrammtyp zum neuen Standardtyp, der ab da bei der Erzeugung eines neuen Diagramms vorgegeben wird.

24 Diagramme formatieren

Im letzten Kapitel über Diagramme geht es um die Feinarbeit. Sie können beliebige zuvor selektierte Diagrammelemente formatieren, beispielsweise um Schriftart und -größe der Legendentexte zu verändern oder die Größenachse logarithmisch statt linear zu skalieren.

Zunächst zeige ich Ihnen, wie Sie die zu formatierenden Elemente selektieren, da einfaches Anklicken nicht immer genügt (beispielsweise selektiert das Anklicken eines einzelnen Datenpunkts nicht diesen, sondern die gesamte Datenreihe).

Zum Formatieren eines Diagrammobjekts benutzen Sie objektspezifische Dialogfelder, deren Möglichkeiten vom ausgewählten Objekt abhängen.

Auf diese Objekte und die zugehörigen Formatiermöglichkeiten gehe ich im Detail ein, angefangen von den harmlosesten Elementen »Zeichnungsfläche« und »Diagrammfläche« bis hin zu »Fehlerindikatoren« und den unglaublich vielfältigen Möglichkeiten zur Formatierung der Diagrammachsen.

Ein paar Beispiele, die die vielfältigen Formatiermöglichkeiten andeuten: Sie können unter anderem

- ➔ die Größe, Farbe und das Hintergrundmuster der Zeichnungs- und der Diagrammfläche verändern,
- ➔ die Legende frei verschieben und sogar Farbe und Muster der optionalen Legendenschlüssel beeinflussen (kleine Symbole vor den einzelnen Legendentexten),
- ➔ wählen, ob neben den einzelnen Säulen, Kreissegmenten etc. der dadurch dargestellte Wert angezeigt werden soll,
- ➔ die Farbe und das Muster der Datenreihen festlegen und bestimmen, ob zusätzliche Fehlerindikatoren angezeigt werden, die einen Unsicherheitsfaktor oder eine Bandbreite widerspiegeln,
- ➔ den Abstand zwischen den einzelnen Datenpunkten definieren und damit Säulen beispielsweise näher zusammenrücken, bis hin zur Überlappung; oder Verbindungslinien zwischen den einzelnen Säulen einzeichnen lassen

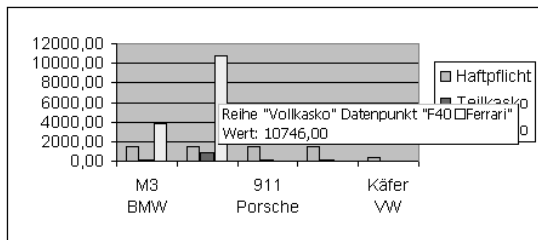
- ➔ und angeben, dass außer Primär- auch Sekundärachsen angezeigt werden; und Sie können sowohl die Ausrichtung der Achsen als auch ihre Beschriftung und ihre Skalierung bestimmen und festlegen, wie extrem hohe oder niedrige Werte behandelt werden sollen.

24.1 Selektieren

Sie können beliebige zuvor selektierte Elemente eines Diagramms mit objektspezifischen Dialogfeldern formatieren.

Excel hilft Ihnen bei der Selektion durch die Anzeige verschiedenster Informationen zu jenem Diagrammelement, auf dem sich gerade der Cursor befindet. Befindet er sich auf einer Säule, werden Sie beispielsweise über die Datenreihe informiert, zu der der Datenpunkt gehört, über den Namen des Datenpunkts und über die Größe des dargestellten Werts (Abbildung 24.1).

Abbildung 24.1:
Diagramm-Tipps



Sie können folgende Elemente selektieren:

- ➔ das gesamte Diagramm
- ➔ den Diagrammtitel
- ➔ die Diagrammfläche, das heißt die von den Achsen umschlossene Fläche
- ➔ die Achsen
- ➔ die Achsenbeschriftungen
- ➔ eine Datenpunktreihe, zum Beispiel die Säulen, die eine bestimmte Datenreihe
- ➔ einzelne Datenpunktbilder, zum Beispiel eine einzelne Säule oder ein Kreissegment
- ➔ die gesamte Legende
- ➔ die einzelnen Legendentexte
- ➔ die zugehörigen Legenden-Symbole
- ➔ und zusätzliche ungebundene Elemente wie Texte oder Pfeile



Mit der Tastatur selektieren Sie ein Element durch wiederholte Betätigung von $\rightarrow$ beziehungsweise $\leftarrow$. Das jeweils selektierte Objekt wird von Anfassern umgeben.

Drücken Sie beispielsweise nach der Aktivierung des Diagrammfensters einmal $\rightarrow$, wird das gesamte Diagramm selektiert. Drücken Sie erneut $\rightarrow$, ist stattdessen die Diagrammfläche selektiert und so weiter.

Allgemein: Mit $\rightarrow$ blättern Sie von Objekt zu Objekt weiter, mit $\leftarrow$ zurück (oder alternativ dazu mit $\uparrow$ / $\downarrow$).

Mit der Maus wählen Sie das betreffende Objekt im Listefeld der Diagramm-Symbolleiste aus oder klicken es ganz einfach an:

- ➔ Um eine Achse zu selektieren, klicken Sie auf die zugehörige Linie oder auf eine der Beschriftungen an den Achsenteilstrichen.
- ➔ Die Diagrammfläche selektieren Sie durch Anklicken eines beliebigen Punkts innerhalb dieser von den Achsen umschlossenen Fläche.
- ➔ Um das gesamte Diagramm zu selektieren, klicken Sie auf einen Punkt außerhalb dieser Fläche.
- ➔ Eine Datenreihe selektieren Sie durch Anklicken irgendeines der zugehörigen Datenpunktbilder, zum Beispiel einer Säule, die zu Reihe 3 gehört. Excel selektiert daraufhin alle Säulen dieser Reihe und zeigt in der Bearbeitungsleiste die zugehörige **DATENREIHE**-Formel an, die die darzustellenden Werte auswählt (Abbildung 24.2).

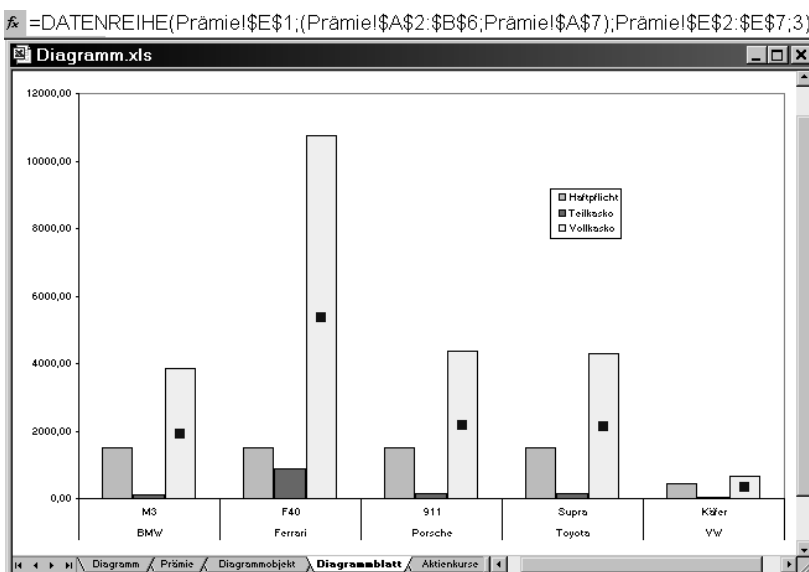


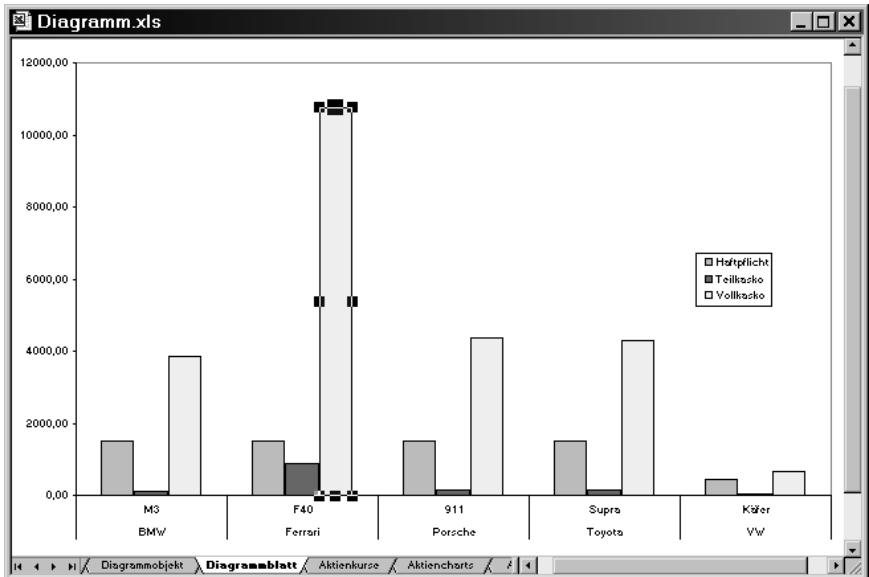
Abbildung 24.2:
Datenreihe
selektieren

**Unterobjekte
selektieren**

Wollen Sie eine einzelne Säule der Datenreihe selektieren, gehen Sie zunächst genauso vor, klicken also auf irgendeine Säule der Reihe, worauf alle Säulen dieser Reihe selektiert werden.

Danach warten Sie einen Moment, um einen Doppelklick zu vermeiden, und klicken die Sie eigentlich interessierende Säule an. Nun ist nur noch diese eine Säule selektiert (Abbildung 24.3).

Abbildung 24.3:
Datenpunkt
selektieren



Anschließend können Sie auf eine beliebige Säule der gleichen Reihe (im Beispiel Reihe 3) klicken, um stattdessen diese zu selektieren.

Diese Selektion eines Objekts in einer Gruppe wird aufgehoben, wenn Sie ein Objekt anklicken, das zu einer anderen Gruppe gehört, beispielsweise eine Säule von Reihe 2. Dann ist zunächst Reihe 2 komplett selektiert. Warten Sie einen Moment und klicken danach auf eine einzelne Säule dieser Reihe, so ist erneut nur diese eine Säule selektiert.

Das gleiche Prinzip können Sie auch auf die Legende anwenden: Anklicken selektiert zunächst die gesamte Legende. Klicken Sie nach einer kurzen Pause ein darin enthaltenes Objekt an, ist nur dieses einzelne Objekt selektiert, zum Beispiel ein einzelner Legendentext.

Um die Selektion eines Diagrammelements wieder aufzuheben, drücken Sie **[Esc]** oder klicken auf irgendeinen Punkt des Diagrammfensters, der sich außerhalb des Diagramms befindet.

24.2 Elemente löschen

Sie löschen ein Diagrammelement, indem Sie

- ➔ es selektieren und **[Entf]** drücken
- ➔ oder es selektieren und BEARBEITEN|LÖSCHEN|ALLES wählen
- ➔ oder es mit der *rechten* Maustaste anklicken und im zugehörigen Kontextmenü LÖSCHEN bzw. MARKIERUNG LÖSCHEN wählen.

Sie können auf diese Weise die Legende löschen oder eine selektierte Datenreihe und vieles mehr. Sogar die gesamte Zeichnungsfläche, wenn Sie sie zuvor selektieren – oder gar die komplette Diagrammfläche.

24.3 Formatier-Dialogfelder

Um ein Diagrammelement zu formatieren, müssen Sie mit einer der drei folgenden Techniken das zugehörige *Formatier*-Dialogfeld öffnen:

- ➔ Sie selektieren das gewünschte Element und wählen danach den ersten Befehl des FORMAT-Menüs, der – abhängig vom selektierten Objekt – beispielsweise FORMAT|MARKIERTE ACHSE... lautet, FORMAT|MARKIERTE DATENREIHEN... oder – nur bei 3D-Diagrammen – FORMAT|MARKIERTE DIAGRAMMWÄNDE...
- ➔ Ein Doppelklick auf das zu formatierende Element *selektiert es und öffnet zusätzlich dieses Dialogfeld*.
- ➔ Sie klicken ein Element mit der rechten Maustaste an, um es zu selektieren und gleichzeitig das zugehörige Kontextmenü zu öffnen. Abhängig vom betreffenden Elementtyp enthält es einen Befehl wie ACHSEN FORMATIEREN..., LEGENDE FORMATIEREN... oder DATENREIHEN FORMATIEREN..., der ebenfalls das zuständige *Formatier*-Dialogfeld öffnet.

Klicken Sie beispielsweise eine Achse mit der rechten Maustaste an, so öffnet sich folgendes Kontextmenü (Abbildung 24.4).

Bei einer Legende erscheint stattdessen das Kontextmenü (Abbildung 24.5).

Wie gesagt, der Inhalt des jeweiligen *Formatier*-Dialogfelds hängt vom zu formatierenden Diagrammelement ab.

Abbildung 24.4:
Achsenspezifisches
Kontextmenü

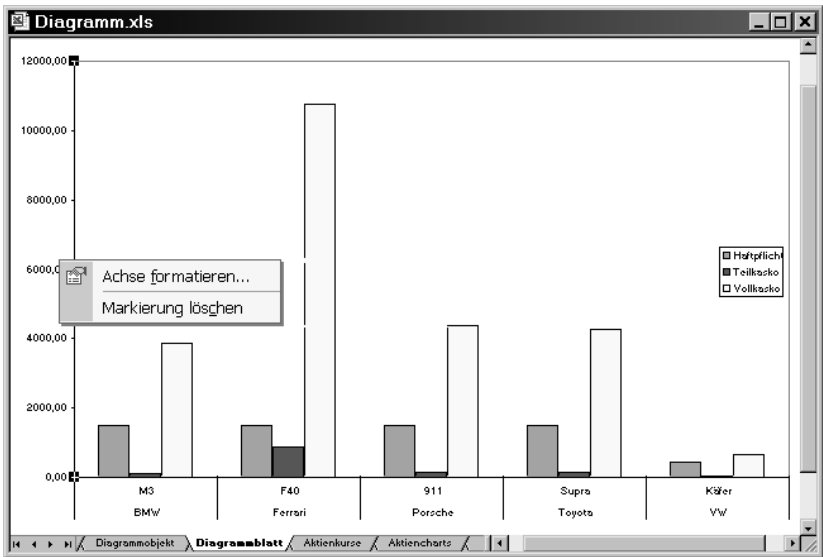
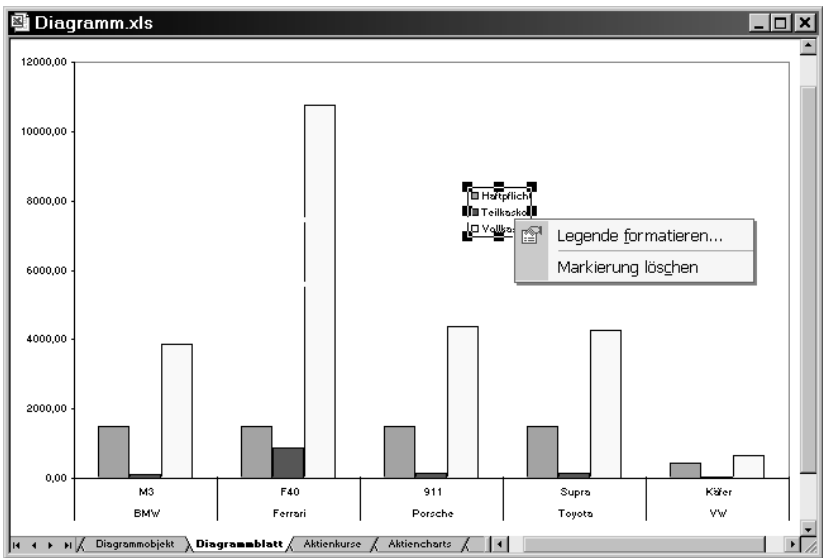


Abbildung 24.5:
Legendenspezifisches
Kontextmenü



Für Datenreihen ist beispielsweise dieses Dialogfeld zuständig (Abbildung 24.6).

Einige seiner Register werden Ihnen immer wieder begegnen, vor allem das abgebildete Register *Muster*, da mit den darin unter *Rahmen* enthaltenen Optionen die Strichart/-stärke/-farbe so unterschiedlicher Elemente wie

- ➔ des Rahmens, der eine Säule umgibt,
- ➔ des Rahmens um die Legende herum,
- ➔ der Achsenlinien

und vieler anderer formatiert wird und mit den Optionen unter *Fläche* entsprechend all diese Objekte mit einer Farbe und/oder einem Muster gefüllt werden.

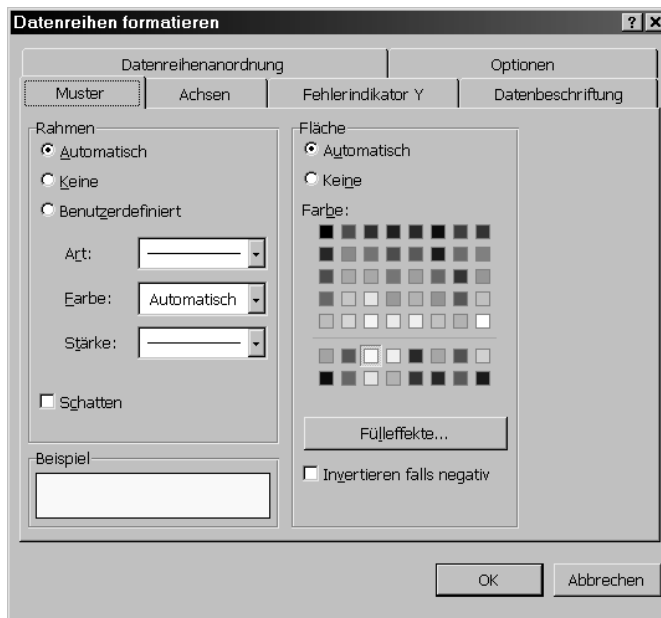


Abbildung 24.6:
Elementspezifisches Formatier-Dialogfeld

Register wie dieses, die Ihnen immer wieder begegnen und in immer gleicher Weise auf unterschiedlichste Objekte angewendet werden, erläutere ich nur einmalig. Ich sehe keinen Sinn darin, ein Register wie Muster ein Dutzend Mal wiederzukaufen – und Sie haben sicher gleichermaßen wenig Interesse daran, die letzten elf dieser Erläuterungen jedesmal überblättern zu müssen.



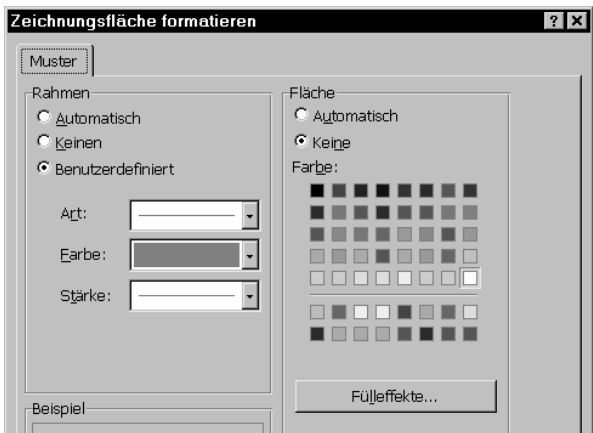
Lesen Sie daher die folgenden Kapitel bitte der Reihe nach! Im ersten erläutere ich das Register *Muster* und in den folgenden setze ich es dann als bekannt voraus. Im zweiten erläutere ich *Schrift* und setze anschließend auch dieses Register als bekannt voraus etc.

Steigen Sie daher erst im zehnten der folgenden Kapitel ein, wird dort möglicherweise ein Register erwähnt und als bekannt vorausgesetzt, von dem Sie noch nie zuvor hörten!

24.4 Zeichnungsfläche

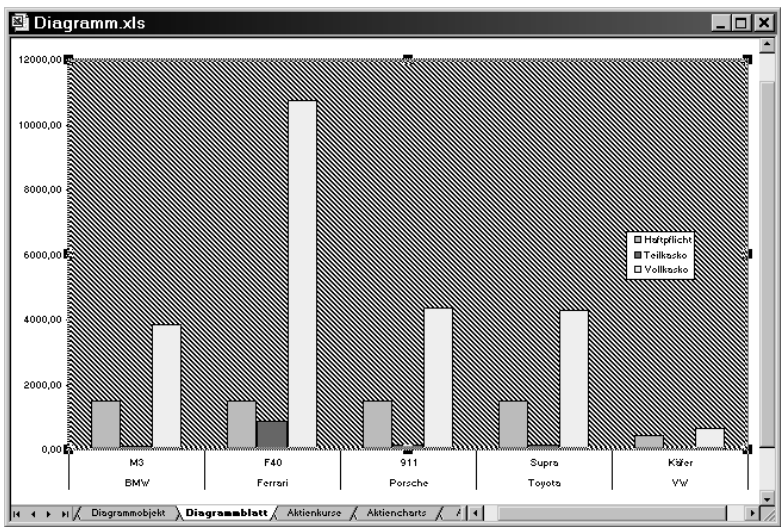
Register Muster Die Zeichnungsfläche ist die Fläche *innerhalb* der Achsen. Selektieren Sie sie per Doppelklick auf irgendeinen Punkt dieser Fläche (bitte auf einen »leeren« Punkt, an dem sich keine Säule oder ein sonstiges Objekt befindet!), so öffnet sich das folgende Dialogfeld (Abbildung 24.7).

Abbildung 24.7:
Register »Muster«



Das abgebildete Register *Muster* ist altbekannt, sowohl vom Formatieren von Tabellen als auch vom Formatieren von Objekten.

Abbildung 24.8:
Zeichnungsfläche
formatieren



Angewandt auf die Zeichnungsfläche können Sie die Farbe und Stärke des Rahmens um diese Fläche herum bestimmen und die Fläche selbst mit einer Farbe und einem zusätzlichen Muster ausfüllen (Abbildung 24.8).

24.5 Diagrammfläche

Die Diagrammfläche ist die Fläche *außerhalb* der Achsen, also die Fläche um die Zeichnungsfläche herum. Das zugehörige *Formatier*-Dialogfeld enthält außer *Muster* das Register *Schrift* (Abbildung 24.9).

Register *Schrift*

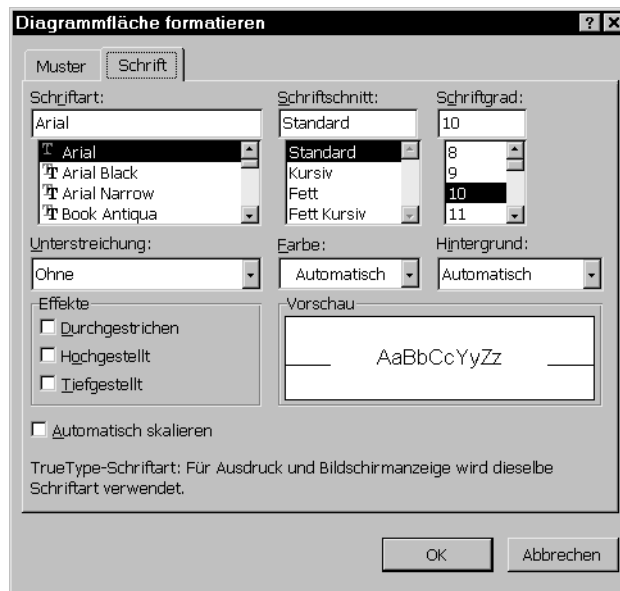


Abbildung 24.9:
Register »Schrift«

Wählen Sie darin eine andere Option, beispielsweise einen anderen Schriftgrad, so werden *alle im gesamten Diagramm enthaltenen Texte* entsprechend umformatiert. Zusätzlich können Sie der Diagrammfläche ein eigenes Muster verleihen (Abbildung 24.10).

Zusätzlich enthält es (allerdings nicht bei Diagrammblättern) das bereits von Grafikobjekten her bekannte Register *Eigenschaften*, mit dem sich vor allem die Bindung des Grafikobjekts an die darunter liegende Tabelle festlegen lässt (Abbildung 24.11).

Abbildung 24.10:
Diagrammfläche
formatieren

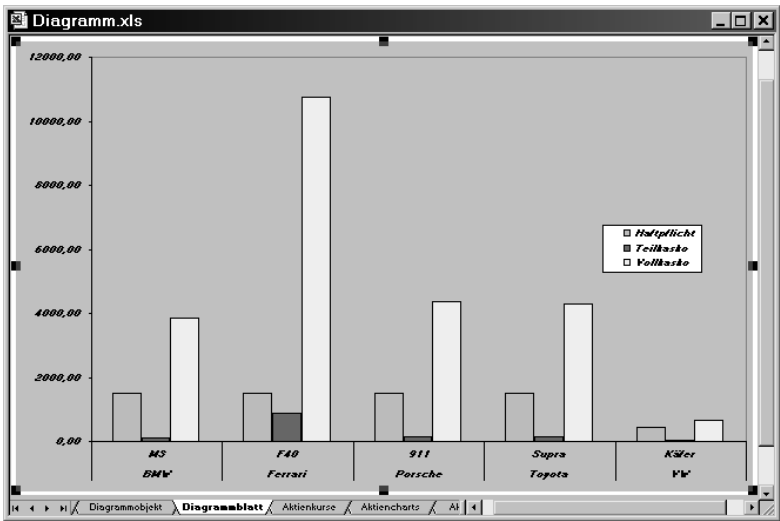


Abbildung 24.11:
Register
»Eigenschaften«

Diagrammfläche formatieren

Muster | Schrift | **Eigenschaften**

Objektposition

- ☒ Von Zellposition und -größe abhängig
- ☐ Nur von Zellposition abhängig
- ☐ Von Zellposition und -größe unabhängig

☒ Objekt drucken

☒ Gesperrt

Eine Sperre von Objekten wird nur dann wirksam, wenn das Blatt geschützt ist. Wählen Sie aus dem Menü "Extras" den Befehl "Dokument schützen" und dann "Blatt". Das Kennwort ist optional.

OK Abbrechen

24.6 Legende

Zunächst einmal können Sie die Legende frei verschieben, indem Sie sie anklicken und bei gedrückter Maustaste zur gewünschten Position ziehen.

Darüber hinaus öffnet ein Doppelklick auf die Legende ein Dialogfeld, das außer den Registern *Muster* (den Legendenrahmen oder das Rahmeninnere formatieren) und *Schrift* (die für die Legendentexte zu verwendende Schrift) das Register *Platzierung* enthält (Abbildung 24.12).

Register
Platzierung

Die einzelnen Optionen sind praktisch selbsterklärend. In allen Fällen wird sie am betreffenden Rand zentriert dargestellt, mit *Unten* beispielsweise so (Abbildung 24.13):

Die einzige Ausnahme ist die Option *Ecke*, die die Legende in die rechte obere Diagrammecke verschiebt, wo sie am wenigsten stört.

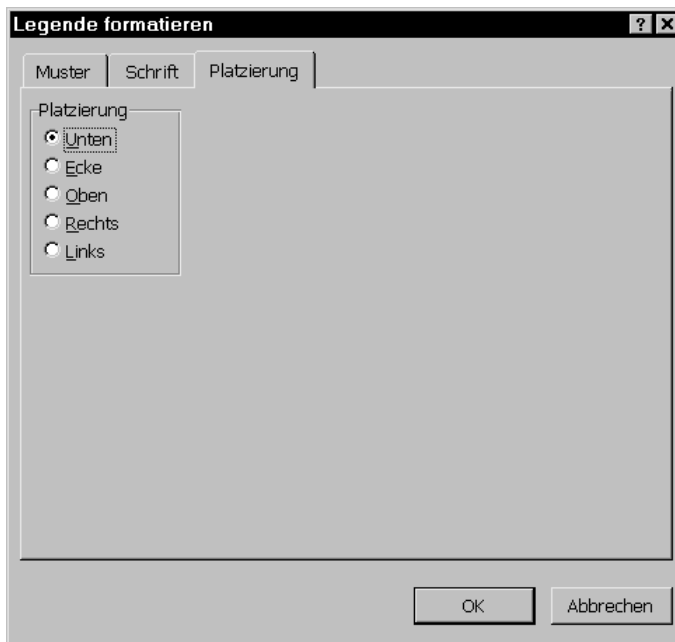


Abbildung 24.12:
Register
»Platzierung«

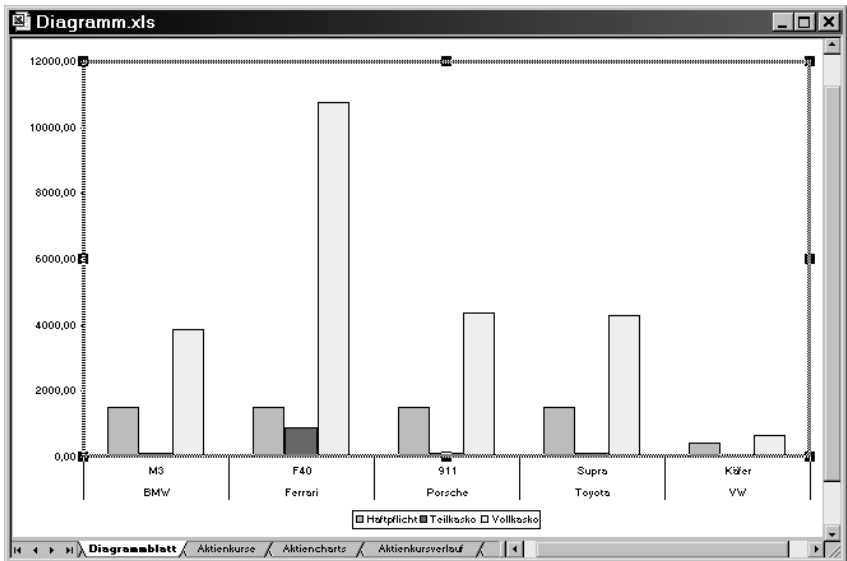
Sie können übrigens auch einzelne Legendentexte oder Legenden-Symbole formatieren: Zunächst selektieren Sie die Legende, nach kurzer Wartepause klicken Sie darin gezielt auf das betreffende Objekt.



Öffnen Sie danach das zugehörige *Formatier*-Dialogfeld (Doppelklick auf das Objekt, Anklicken mit der rechten Maustaste oder Wahl von **FORMAT|MARKIERTER LEGENDENEINTRAG...**), so enthält es nur noch ein

Register: *Schrift*, falls ein Legendentext selektiert wurde bzw. *Muster* bei einem Legenden-Symbol, um damit die Farbe bzw. das Muster des zugehörigen Kästchens festzulegen.

Abbildung 24.13:
Legende
formatieren



24.7 Datenpunkte

Einen einzelnen Datenpunkt selektieren Sie ebenso wie ein einzelnes Legendenobjekt: Durch Anklicken irgendeines Punkts (z.B. einer Säule) der betreffenden Reihe selektieren Sie zunächst die gesamte Reihe, danach – nach einer kurzen Wartepause – die gewünschte Säule durch einen gezielten Klick darauf.

Anschließend öffnen Sie das zugehörige *Formatier*-Dialogfeld (Doppelklick auf das Objekt, Anklicken mit der rechten Maustaste oder Wahl von FORMAT[MARKIERTER DATENPUNKT...]).

Register Daten-
beschriftung

Mit dem bereits bekannten Register *Muster* können Sie den Rahmen verändern, der eine Säule umgibt, oder die Säulenfläche ausfüllen. Zusätzliche Möglichkeiten eröffnet das Register *Datenbeschriftung* (Abbildung 24.14).



Die verschiedenen Beschriftungsoptionen wurden bereits beim Formatieren von Datenreihen erläutert (siehe Kapitel 22.5.5, »Datenbeschriftungen«) und können nach dem Selektieren eines einzelnen Datenpunkts verwendet werden, um ihn individuell zu beschriften.

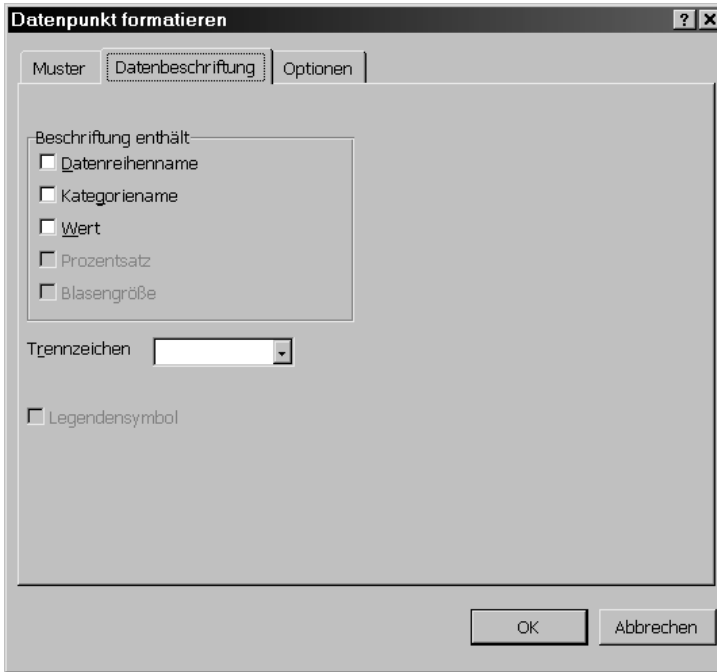


Abbildung 24.14:
Register »Daten-
beschriftung«

Für die folgende Abbildung selektierte ich die rechte Säule und aktivierte sowohl *Wert* als auch *Legenden-Symbol* (Abbildung 24.15).

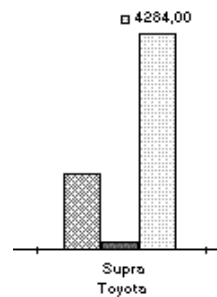


Abbildung 24.15:
Datenpunkt
formatieren

Ist eine Säule selektiert und bewegen Sie den Cursor zum oberen Säulenrand, wird er zu einem Doppelpfeil. Sie können den Rand nun anklicken und nach oben oder unten ziehen, um die Höhe der Säule zu verändern. Da das Diagramm mit den zu Grunde liegenden Daten verbunden ist, aktualisiert Excel den zugehörigen Tabellenwert automatisch.



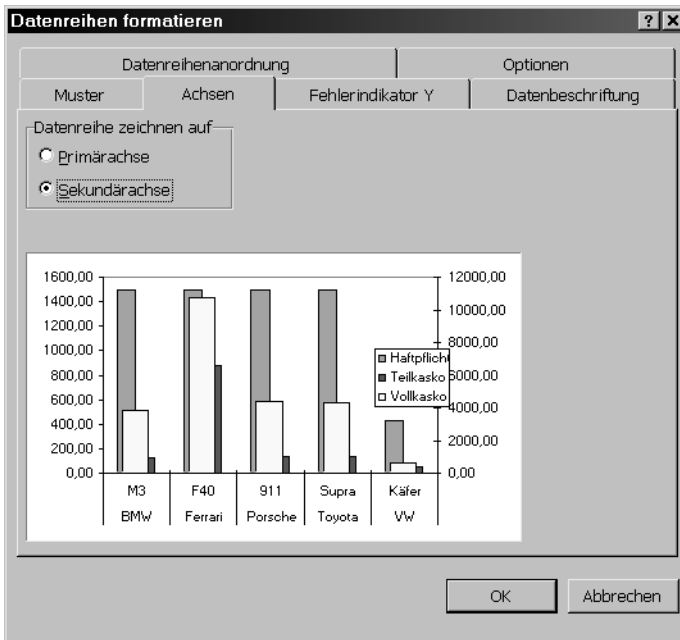
Analog dazu können Sie in Balkendiagrammen am rechten Rand eines Balkens ziehen, um ihn zu verkürzen oder zu verlängern, in Punktdiagrammen Punkte verschieben etc.

24.8 Datenreihen

24.8.1 Achsen

Das Dialogfeld zum Formatieren von Datenreihen enthält außer *Muster* und *Datenbeschriftung* noch einige weitere Register, zum Beispiel *Achsen* (Abbildung 24.16).

Abbildung 24.16:
Register »Achsen«



Sekundärachse trägt die Werte der selektierten Reihe nicht an der linken primären Größenachse auf, sondern an der rechten sekundären (siehe Abbildung). Existiert keine sekundäre Größenachse, so wird sie erzeugt. Anschließend können Sie eine weitere Reihe selektieren und auch diese an der sekundären statt an der primären Größenachse darstellen lassen.

Zum Sinn dieses Features: Enthält eine Datenreihe einen oder mehrere Werte, die sich extrem von den Werten der restlichen Reihen unterscheiden, sollten Sie für die betreffende Reihe eine sekundäre Größenachse verwenden. Da sich die Skalierung der Größenachse zwangsläufig immer am größten darzustellenden Wert orientiert, werden sonst die restlichen Werte nivelliert und die Differenzen zwischen diesen Werten sind kaum noch sichtbar.

Ein schönes Beispiel dafür stellt das Versicherungsdiagramm dar, in dem die Reihe »Vollkasko« einen außerordentlich hohen Wert für die Vollkaskoprämie des Ferrari F40 enthält, 10.746 DM.

Der daran orientierte Maßstab der Größenachse führt dazu, dass die Differenzen zwischen den Teilkaskoprämien für den BMW (123 DM) und den Porsche (135 DM) nicht unterscheidbar sind.

Um dieses Problem zu lösen, weisen Sie der Datenreihe mit den extrem von den restlichen Reihen abweichenden Werten einen anderen Diagrammtyp zu und geben diesem Diagramm anschließend *eine eigene Größenachse*.

Dazu selektieren Sie im Beispiel die Reihe »Vollkasko«, wählen DIAGRAMM|DIAGRAMMTYP... und als Darstellungsform beispielsweise das Liniendiagramm, um diese Reihe als Linie darzustellen..

Anschließend öffnen Sie das Dialogfeld zum Formatieren der Datenreihe (z.B. per Doppelklick auf die Linie) und schlagen darin das Register *Achsen* auf (Abbildung 24.17).

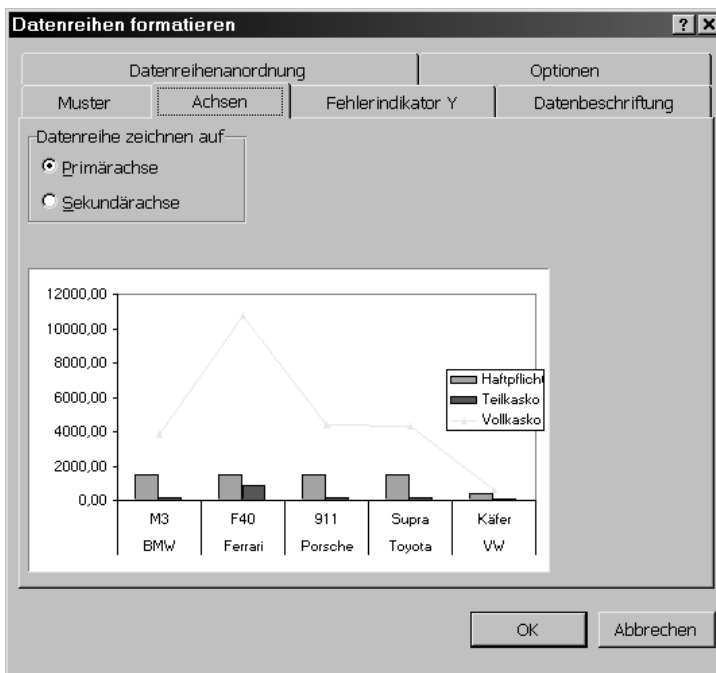
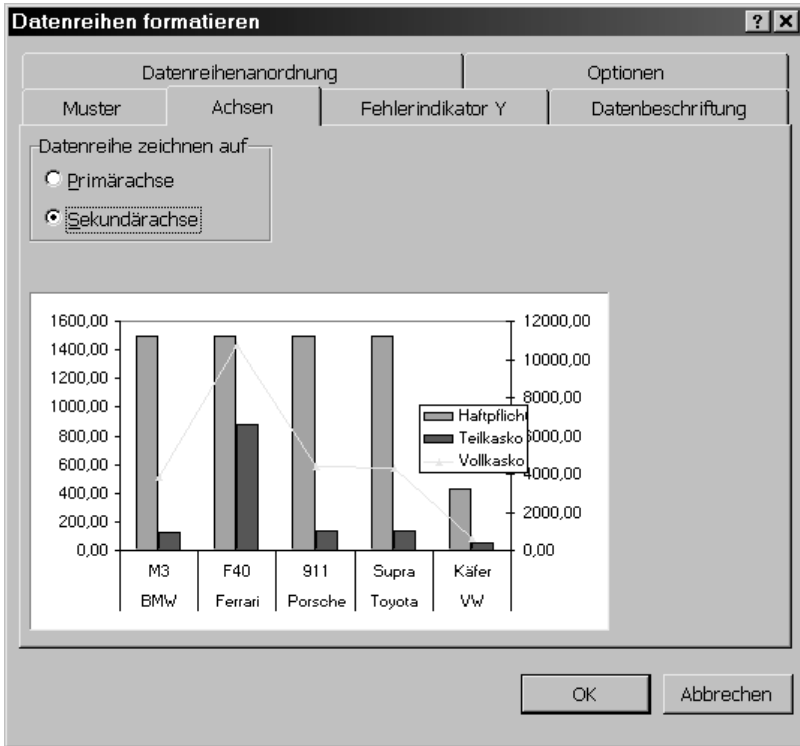


Abbildung 24.17:
Register »Achsen«

Darin ist *Primärachse* selektiert, da die Werte der selektierten Reihe »Vollkasko« (des Liniendiagramms) momentan ebenso wie die Werte der beiden anderen Datenreihen auf der linken Primärachse aufgetragen sind. Aktivieren Sie stattdessen jedoch *Sekundärachse*, erhalten Sie folgendes Resultat (Abbildung 24.18).

Abbildung 24.18:
Sekundärachse für
zweite Diagramm-
typgruppe



Für die betreffende Diagrammgruppe wird am rechten Diagrammrand eine eigene Größenachse aufgetragen, deren Skalierung sich am höchsten Wert *dieser Gruppe* orientiert.

Die Skalierung der linken Größenachse orientiert sich jetzt nur noch am höchsten Wert der restlichen Datenreihen und ist daher von der extremen Vollkaskoprämie vollkommen unabhängig. Nun sind auch annähernd gleiche Werte dieser Reihen optisch gut unterscheidbar.

24.8.2 Fehlerindikatoren

Bei Datenreihen, die nicht auf völlig gesicherten Werten basieren, bei denen also ein Unsicherheitsfaktor vorliegt, können Sie mit Hilfe des Registers *Fehlerindikator Y* den vermuteten Unsicherheitsbereich einzeichnen lassen (Abbildung 24.19).

Ist *Keine* aktiviert, werden keine Fehlerindikatoren eingezeichnet.

Die restlichen Optionen erläutere ich an einem Beispiel: Nehmen wir an, die Vollkaskoprämien stellen Durchschnittswerte dar, denen die bei fünf Versicherungen erfassten Prämiensätze zu Grunde gelegt wurden. In der Praxis

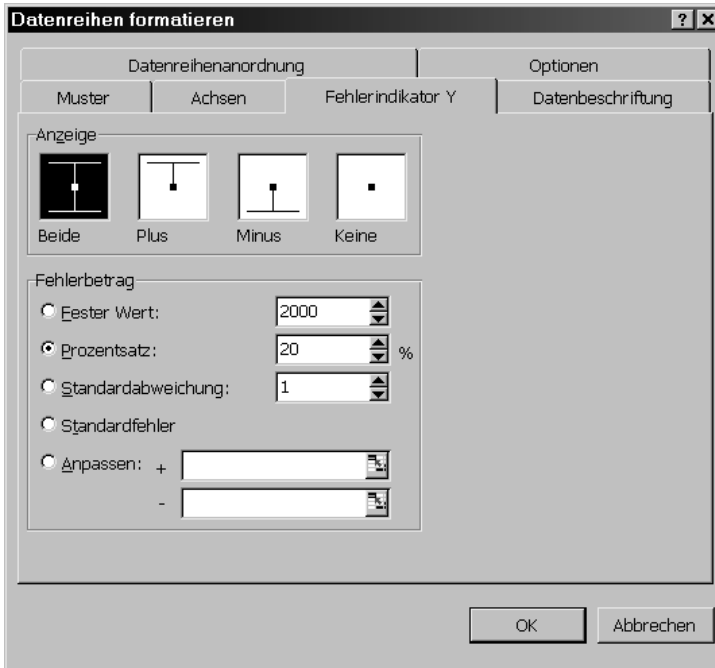


Abbildung 24.19:
Register »Fehler-
indikator Y«

kann die Vollkaskoprämie jedoch bei der Versicherung XYZ von diesem Durchschnitt abweichen, sagen wir um 20%.

Um diesen Sachverhalt grafisch darzustellen, aktivieren Sie unter *Anzeige* die Option *Beide* und unter *Fehlerbetrag* die Option *Prozentsatz*. Im zugehörigen Scrollfeld geben Sie den Wert 20% ein.

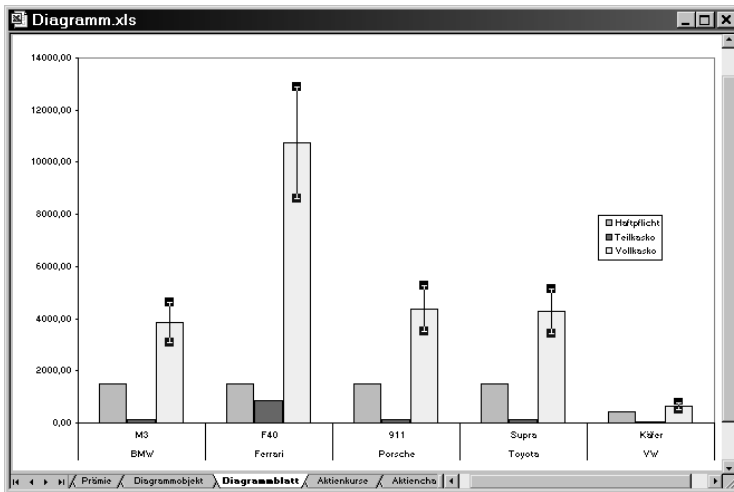
Sie verwenden also exakt jene Optionen, die in der vorhergehenden Abbildung zu sehen sind und erhalten damit folgendes Ergebnis (Abbildung 24.20):

Anhand der vom jeweiligen oberen Säulenrand (nennen wir den zugehörigen Wert kurz »SR«) aus gezeichneten Indikatoren wird der in beiden Richtungen vorhandene Unsicherheitsbereich wunderschön grafisch dargestellt; durch Indikatoren, die bis $SR+20\%$ reichen bzw. bis $SR-20\%$.

Die restlichen Optionen unter *Anzeige*:

- ➔ *Plus* würde nur Indikatoren vom oberen Säulenrand nach oben einzeichnen, in Richtung höherer Werte (nur zu $SR+20\%$).
- ➔ *Minus* würde nur Indikatoren in Richtung niedrigerer Werte einzeichnen (nur zu $SR-20\%$).

Abbildung 24.20:
Prozentindikator



Unter *Fehlerbetrag*:

- ➔ *Fester Wert* hebt bei allen Säulen den – in absoluten Zahlen – gleichen Unsicherheitsbereich hervor.

Der Wert 1000 würde bei einer Säule, der der Wert 3864 (Vollkaskoprämie BMW M3) zu Grunde liegt, von 2864 bis 4864 reichende Indikatoren einzeichnen. Bei einer Säule mit dem Basiswert 10746 (Ferrari F40) würden sich die Indikatoren dagegen von 9746 bis 11746 erstrecken.

- ➔ *Standardabweichung*: Unter Standardabweichung, kurz »SD«, versteht man die durchschnittliche Abweichung der Werte von ihrem Mittelwert, kurz »MW«, also die Streuung um diesen Mittelwert herum.

Mit der Vorgabe 1 zeichnet Excel zu jeder Säule Indikatoren ein, die vom Wert $MW - SD$ bis $MW + SD$ reichen, mit der Vorgabe 0,5 dagegen Indikatoren, die von $MW - SD/2$ bis $MW + SD/2$ reichen.

Sie sehen danach auf einen Blick, ob ein bestimmter Wert um mehr als die durchschnittliche Abweichung vom Mittelwert abweicht oder nicht (Abbildung 24.21).

- ➔ *Standardfehler* ermittelt die gleiche Spannweite, zeichnet sie jedoch ausgehend von der individuellen Säulenhöhe nach oben bzw. unten ein, also bis zu $SR + SD$ bzw. $SR - SD$.
- ➔ *Anpassen*: Unter »+« eingetragene Werte wie 1000 definieren vom oberen Säulenrand SR aus nach oben gezeichnete Indikatoren (bis zum Wert $SR + 1000$), unter »-« eingetragene definieren von da aus nach unten gezeichnete Indikatoren (bis zum Wert $SR - 1000$).

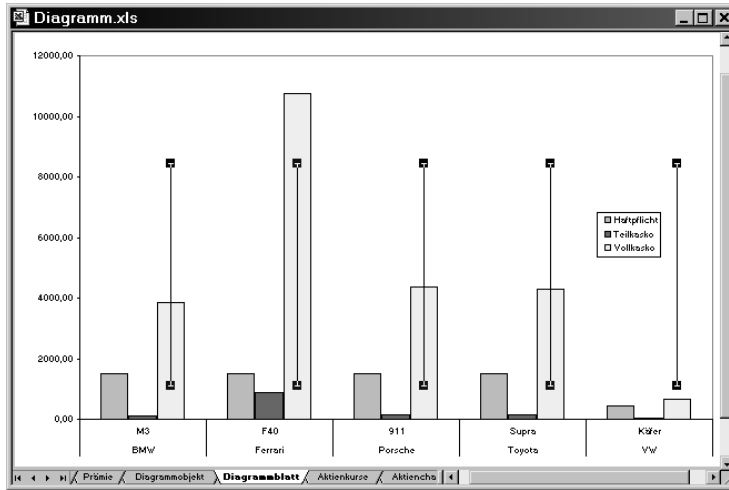


Abbildung 24.21:
Mit Standard-
abweichung 1

Nach dem Einfügen von Indikatoren sind diese zunächst selektiert. Sie können sie jedoch auch später jederzeit durch Anklicken selektieren. Drücken Sie anschließend die rechte Maustaste, so erscheint folgendes Kontextmenü (Abbildung 24.22).

Es ermöglicht Ihnen das Entfernen der Indikatoren oder das Öffnen eines Dialogfeldes mit dem erläuterten Register *Fehlerindikator Y*. Zusätzlich enthält es das Register *Muster*, mit dem Sie die Linienart/-stärke/-farbe bestimmen und zwischen zwei unterschiedlichen Arten von Endpunkten wählen können.

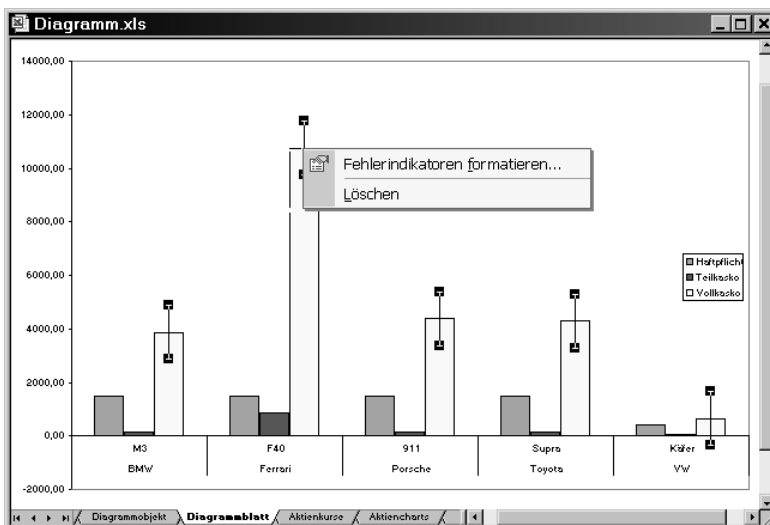
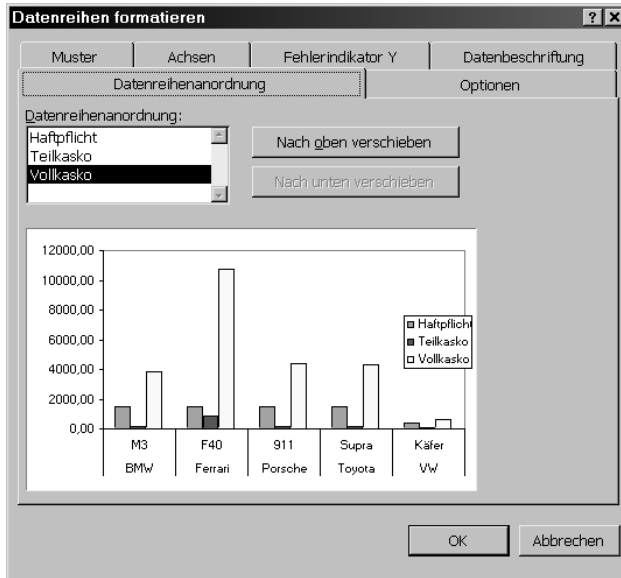


Abbildung 24.22:
Fehlerindikatoren-
Kontextmenü

24.8.3 Datenreihenanzordnung

Im Register *Datenreihenanzordnung* können Sie die Reihenfolge der Datenreihen verändern (Abbildung 24.23).

Abbildung 24.23:
Register »Datenreihenanzordnung«



Angenommen, Sie wollen die Reihenfolge »Haftpflcht«, »Teilkasko«, »Vollkasko« umkehren, in »Vollkasko«, »Teilkasko«, »Haftpflcht«. Dazu selektieren Sie die Reihe »Haftpflcht« und klicken solange auf *Nach unten verschieben*, bis sich diese Reihe am unteren Listenende befindet. Danach selektieren Sie »Vollkasko« und klicken einmal auf *Nach oben verschieben*, um die Reihe an das obere Listenende zu verschieben.

Bei jeder Änderung der Reihenfolge aktualisiert Excel das Vorschaufenster, sodass Sie über die Auswirkungen sofort informiert sind. OK überträgt die geänderte Datenreihenfolge auf das Diagramm (Abbildung 24.24).

24.8.4 Optionen

Im Register *Optionen* können Sie die Anordnung der einzelnen Datenpunktbilder beeinflussen, zum Beispiel den Abstand zwischen den einzelnen Säulen oder Balken (Abbildung 24.25).

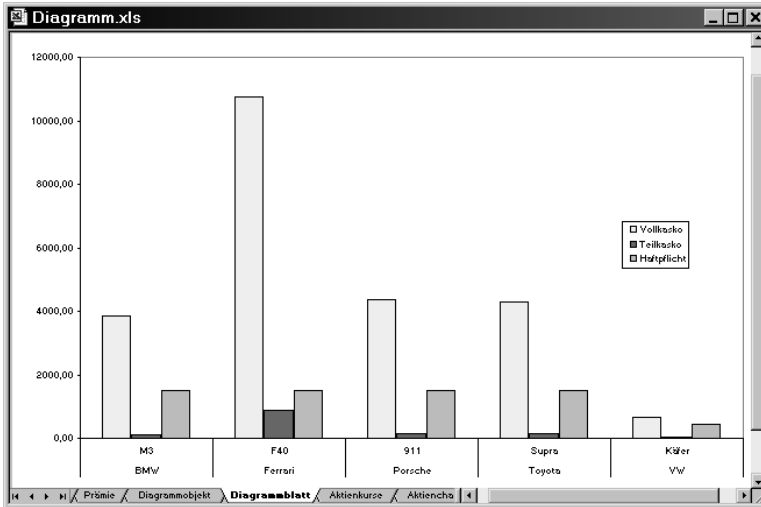


Abbildung 24.24:
Reihenfolge
umkehren

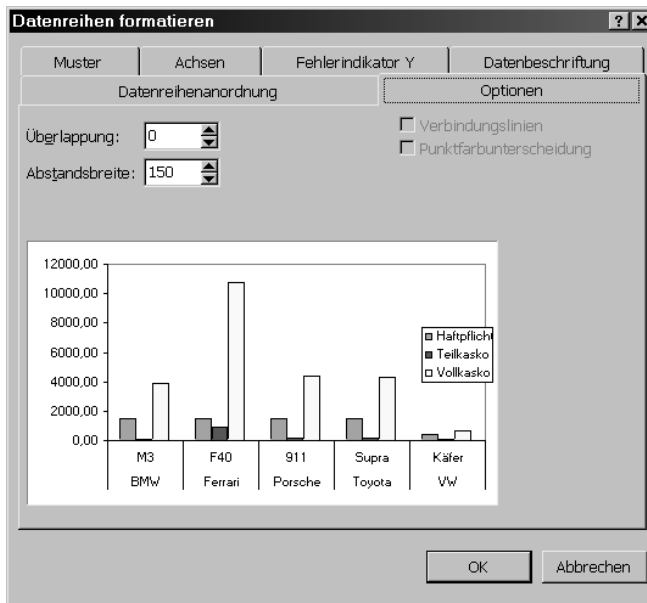


Abbildung 24.25:
Register »Optionen«

Die verfügbaren Optionen hängen davon ab, welcher Diagrammtyp aktiv ist! Für Kreisdiagramme gibt es andere Optionen als für Säulendiagramme, für diese wiederum andere als für Ringdiagramme etc.



Die Abbildung zeigt die Optionen, die für Datenreihen verfügbar sind, die als Balken-, Säulen-, Zylinder-, Kegel- oder Pyramidendiagramme dargestellt werden.

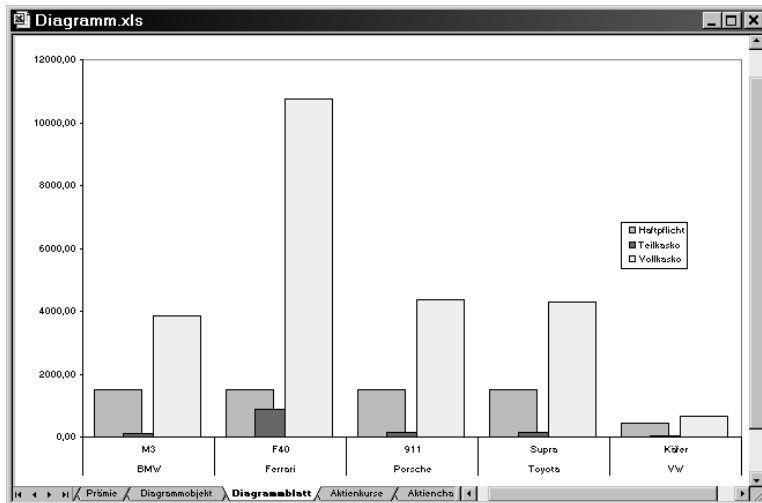
**Balken-, Säulen-,
Zylinder-, Kegel-
und Pyramiden-
diagramme**

- ➔ *Überlappung* ist nur für Balken- und Säulendiagramme verfügbar und legt den Abstand zwischen den einzelnen Balken einer Rubrik fest.
- ➔ *Abstandsbreite* bestimmt den Abstand zwischen den Balken zweier benachbarter Rubriken. Beide Werte sind Prozentangaben, die sich auf die Breite der Balken/Säulen beziehen.

Die möglichen Extremfälle:

- ➔ Überlappung -100%: Der Abstand zwischen den einzelnen Säulen einer Rubrik beträgt genau eine Säulenbreite.
- ➔ Überlappung 0%: Zwischen den Säulen gibt es keinen Abstand, sie grenzen unmittelbar aneinander.
- ➔ Überlappung 100%: Die Säulen überdecken einander vollkommen, sodass pro Rubrik nur noch eine Säule sichtbar ist.
- ➔ Abstand 500%: Benachbarte Rubriken trennen genau fünf Säulenbreiten Abstand.
- ➔ Abstand 0%: Zwischen den Säulen benachbarter Rubriken gibt es keinerlei Abstand (Abbildung 24.26).

Abbildung 24.26:
Säulenabstände



Für diese Abbildung wurde der Abstand zwischen den Säulen innerhalb einer Rubrik – die Überlappung – auf 40% und der Abstand zwischen den Rubriken auf 50% verringert.

Verbindungslinien ist nur für gestapelte Balken oder Säulen verfügbar und verbindet die einzelnen Punkte jeder Datenreihe durch eine Linie (Abbildung 24.27).

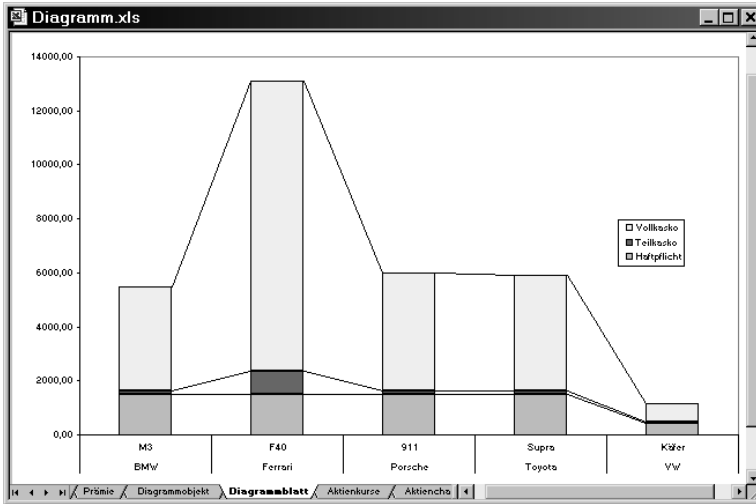


Abbildung 24.27:
Verbindungslinien

Für 3D-Diagramme stehen folgende abstandsbestimmende Optionen zur Verfügung (siehe Abbildung 24.28).

3D-Diagramme

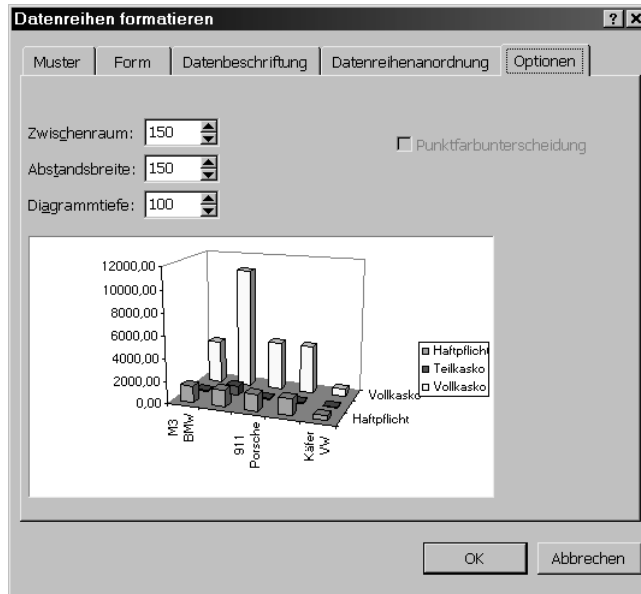


Abbildung 24.28:
3D-Diagramm-
Optionen

Zwischenraum (0 bis 500; Säulendiagramme: 20 bis 2000) bestimmt den Abstand zwischen den Datenpunkten – beispielsweise Säulen – einer Rubrik, z.B. zwischen den drei Säulen für die Prämien »Haftpflicht«, »Teilkasko« und »Vollkasko« der Rubrik »VW Käfer«.

100% bedeutet, dass der Abstand dieser drei Säulen voneinander exakt der Tiefe der einzelnen Säulen entspricht. Durch Erhöhen dieses Werts wächst das Diagramm jedoch nicht wie erwartet in die Tiefe; stattdessen wird die Tiefe der Säulenreihen verringert, die Diagrammtiefe selbst bleibt unverändert.

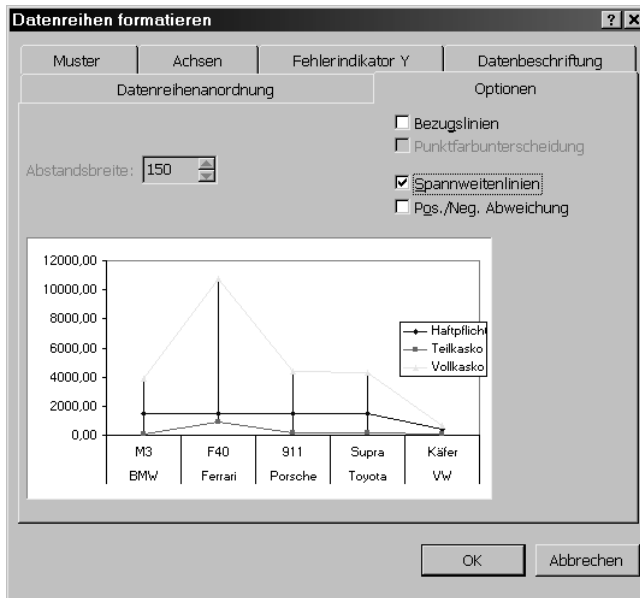
Sie wird durch *Diagrammtiefe* festgelegt, einen Wert zwischen 0 und 500, der die Tiefe des Diagramms im Verhältnis zu seiner Breite definiert.

Abstandsbreite legt den Abstand zwischen den einzelnen Datenpunkten einer Reihe fest, beispielsweise zwischen den Säulen in der hintersten Reihe »Vollkasko«.

Liniendiagramme

Für Liniendiagramme, speziell in Börsenanwendungen, gibt es ebenfalls einige besondere Optionen (Abbildung 24.29).

Abbildung 24.29:
Liniendiagramm-
Optionen



Spannweitenlinien ziehen (siehe Abbildung) senkrechte Linien vom höchsten zum tiefsten Wert jeder Rubrik, sodass beispielsweise die maximale Differenz des abgebildeten Kursverlaufs sichtbar wird.

Bezugslinien (auch für Flächendiagramme verfügbar) ziehen Linien von jedem Datenpunkt bis herunter zur Rubrikenachse. Angewandt auf die letzte Abbildung würden die vertikalen Linien daher weiter nach unten reichen, bis zu dieser Achse.

Pos./Neg. Abweichung ist unter anderem für Börsendiagramme geeignet und zeichnet vom Eröffnungsbis zum Schlusskurs reichende Rechtecke ein.

Im Gegensatz zu Spannweitenlinien wird dadurch nicht die maximale Differenz sichtbar gemacht, sondern die Kursdifferenz zwischen Börsenbeginn und Börsenschluss (Abbildung 24.30).

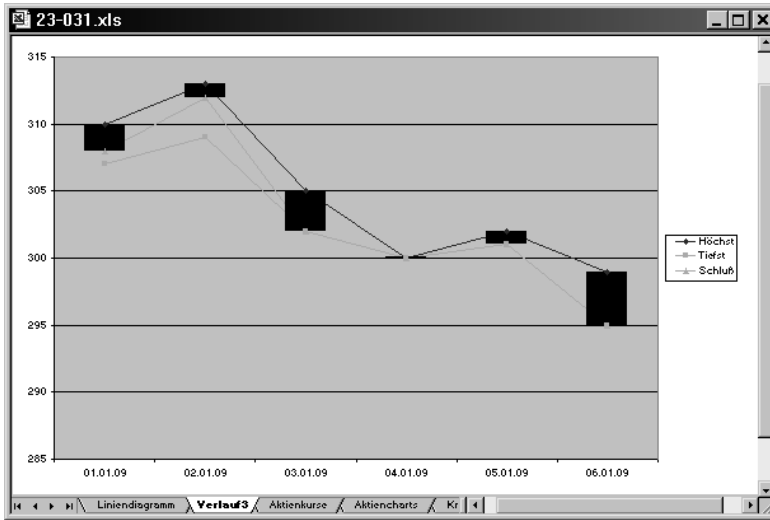


Abbildung 24.30:
Zeitliche
Kursdifferenzen

Spezielle Optionen für Ring- und Kreisdiagramme (Abbildung 24.31) sehen Sie hier:

Ring- und Kreisdiagramme

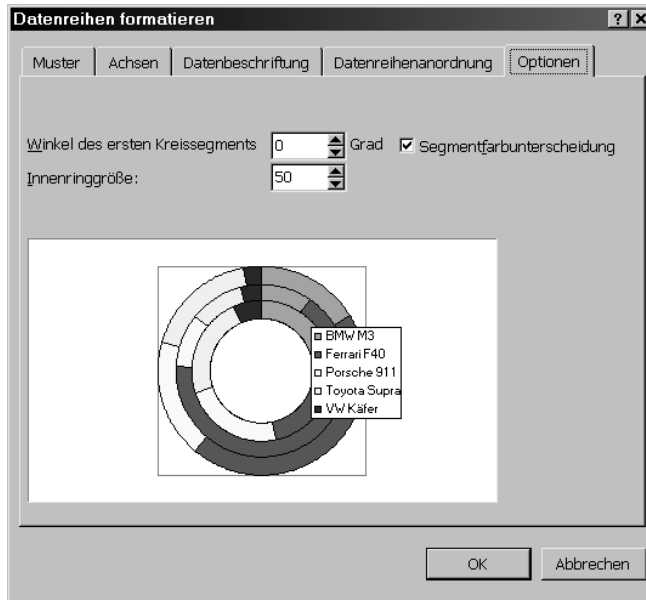


Abbildung 24.31:
Kreis- und
Ringdiagramm-
Optionen

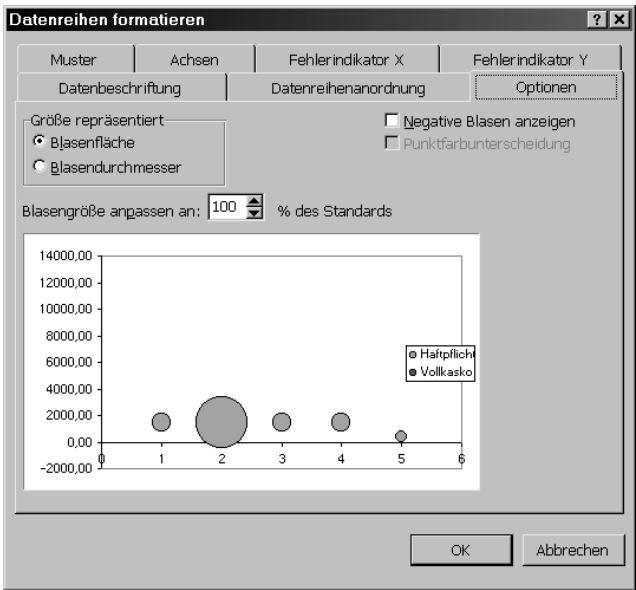
Innenringgröße (10 bis 90) ist eine Ringdiagramm-Option und nicht für Kreisdiagramme verfügbar. Je höher der Wert, desto größer ist der Durchmesser des inneren Rings, umso schmäler jedoch genau umgekehrt die einzelnen Ringe, sodass der Gesamtdurchmesser unverändert bleibt.

Winkel des ersten Kreissegments (0 bis 360 Grad) bestimmt die Position dieses Segments im Kreis bzw. im Ring: 0 Grad bedeutet, dass es an der 12-Uhr-Position beginnt; 90 Grad, dass es an der 3-Uhr-Position beginnt etc. Anders ausgedrückt: Diese Option dreht den Ring oder Kreis um den angegebenen Winkel.

Netzdiagramme Für Netzdiagramme ist die Option *Rubrikenachsenbeschriftungen* verfügbar, die Beschriftungen für die Rubrikenachsen in Netzdiagrammen anzeigt.

Blasendiagramme Das sind spezielle Optionen für Blasendiagramme (Abbildung 24.32):

Abbildung 24.32:
Blasendiagramm-
Optionen



Beispielsweise ermöglicht Ihnen *Blasengröße anpassen an*, die Größe einer zuvor im Diagramm selektierten Blase festzulegen.

24.9 Achsen

24.9.1 Ausrichtung

Eine Achse selektieren Sie durch Anklicken der Achsenlinie (genaues Zielen notwendig) oder einer der zugehörigen Rubrikenbeschriftungen bzw. eines Punktes zwischen diesen Beschriftungen.

Das zugehörige *Formatier*-Dialogfeld enthält außer dem *Schrift*-Register, mit dem Sie die Achsenbeschriftungen formatieren können (die Werte bzw. die Rubrikennamen), unter anderem das Register *Ausrichtung* (Abbildung 24.33).

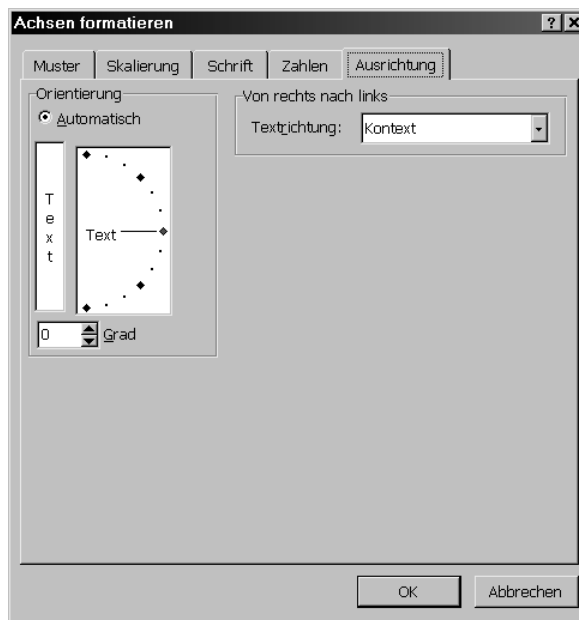


Abbildung 24.33:

Register
»Ausrichtung«

Analog zu Zellinhalten von Tabellen können Sie damit die Achsenbeschriftungen horizontal oder vertikal ausrichten oder beliebige dazwischen liegende Winkel festlegen, beispielsweise 45 Grad. Bei mehrzeiligen Beschriftungen wie im Beispiel wird jedoch nur die Ausrichtung der ersten Zeile beeinflusst (Abbildung 24.34).

Nur beim Formatieren der Rubrikenachse enthält das Dialogfeld die Option *Versetzt*, die den Beschriftungsabstand zum Blattrand festlegt, und die Option *Ausrichtung*, die die Beschriftung rechts-/linksbündig oder zentriert ausrichtet (beides funktioniert erneut nur bei einzeiligen Beschriftungen optimal).

Abbildung 24.34:
Achsentexte
ausrichten

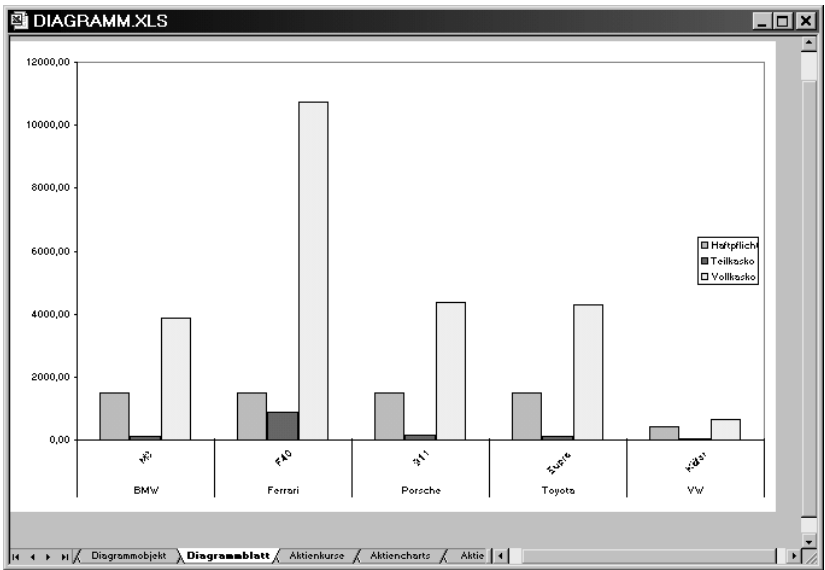
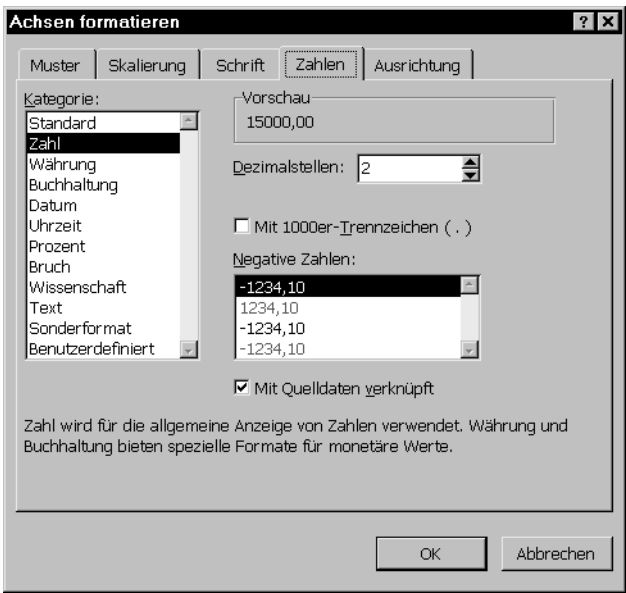


Abbildung 24.35:
Numerische
Achsenbeschriftungen
formatieren



24.9.2 Achsenwerte

Ist eine Achse mit numerischen Beschriftungen selektiert (die Größenachse oder eine eventuell vorhandene sekundäre Größenachse), so können Sie die dargestellten Zahlenwerte mit dem Register *Zahlen* wie gewohnt formatieren (Abbildung 24.35).



Dass Mit Quelldaten verknüpft aktiviert ist, bedeutet, die Achsenbeschriftungen sind mit den zugehörigen Daten in der Tabelle verknüpft und werden aktualisiert, wenn sich diese ändern.

Formatieren Sie die Beschriftungen, weisen Sie ihnen beispielsweise ein anderes Zahlenformat zu, so wird diese Verknüpfung jedoch aufgehoben. Sollen die Achsenbeschriftungen weiterhin mit den Quelldaten verknüpft bleiben, müssen Sie dieses Kontrollkästchen erneut aktivieren, bevor Sie das Dialogfeld schließen!

24.9.3 Achsenmuster

Das *Muster*-Register unterscheidet sich bei Achsen deutlich vom gewohnten entsprechenden *Tabellen*-Dialogfeld (Abbildung 24.36).



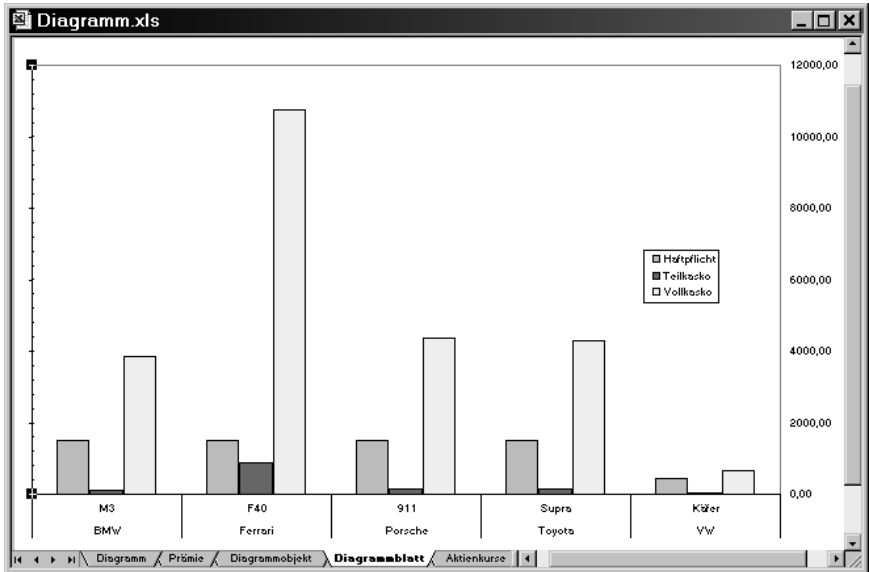
Abbildung 24.36:
Register »Muster«

Hauptstriche sind die winzigen Querstriche an den Achsen, unmittelbar neben den zugehörigen Achsenbeschriftungen, die eine bessere Zuordnung von Werten und Rubrikennamen ermöglichen.

Hilfsstriche sind zusätzliche Querstriche zwischen jeweils zwei Hauptstrichen, die zu einer noch feineren Unterteilung der Achse führen.

Für beide Teilungsarten können Sie wählen, ob die Striche überhaupt nicht dargestellt werden sollen (*Keine*) oder – wenn doch – an den Innenseiten der Achsen, den Außenseiten oder beiden Seiten.

Abbildung 24.37:
Auswirkungen der
Formatierung



Die Optionsgruppe *Teilstrichbeschriftungen* legt ebenfalls den Ort der Beschriftungen fest. Mit dem Standard *Achsennah* befinden sich die Beschriftungen direkt an den Achsen. *Hoch* verlagert sie dagegen an den entgegengesetzten Diagrammrand.

Zusammen mit zusätzlichen Hilfsstrichen *Innen* (an der Diagramminnen-seite der Achse) ergibt sich Abbildung 24.37.

Angewandt auf die Rubriken- statt auf die Größenachse würden deren Beschriftungen oberhalb der Diagrammfläche erscheinen.

24.9.4 Skalierung der Größenachse

Das Register *Skalierung* enthält unterschiedliche Optionen, je nachdem ob die Größen- oder die Rubrikenachse selektiert ist. Angewandt auf die Größenachse ergibt das folgende Bild (Abbildung 24.38).

Zu jedem Eingabefeld gehört ein eigenes Kontrollkästchen (unterhalb von *Automatisch*). Ändern Sie den im zugehörigen Eingabefeld automatisch vorgegebenen Wert, so wird dieses Kontrollkästchen von Excel deaktiviert.

Aktivieren Sie das Kontrollkästchen wieder, verwendet Excel erneut den Standardwert – was jedoch leider nicht sofort im zugehörigen Eingabefeld angezeigt wird. Das führt nach der Aktivierung leicht zum Eindruck, dass sich gar nichts geändert hätte.

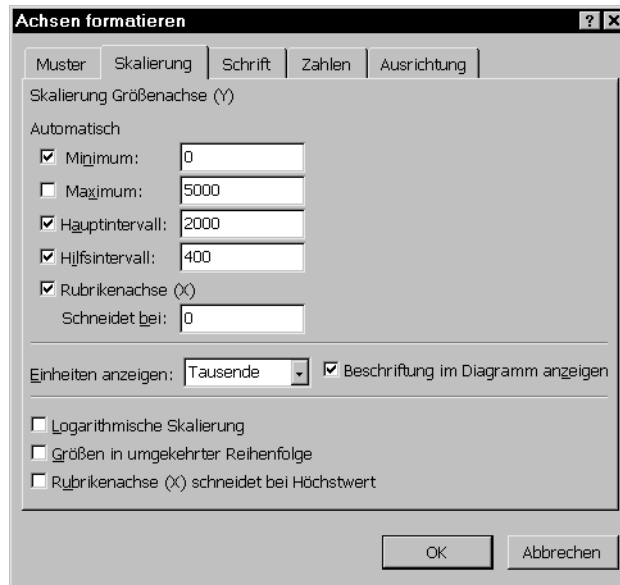


Abbildung 24.38:

Register
»Skalierung«

Ignorieren Sie diesen Eindruck einfach und vertrauen Sie darauf, dass die Aktivierung tatsächlich wieder den Standardwert setzt. Sie können sich davon überzeugen, indem Sie das Dialogfeld anschließend erneut öffnen – nun ist der zuvor vermisste Standardwert tatsächlich wieder vorgegeben.

Minimum und *Maximum* bestimmen die Skalierung an den beiden Enden der GröÙenachse. Die Standardwerte entsprechen dem kleinsten bzw. dem größten Datenpunkt, das heißt alle Werte werden exakt angezeigt.

Extremwert-
darstellung

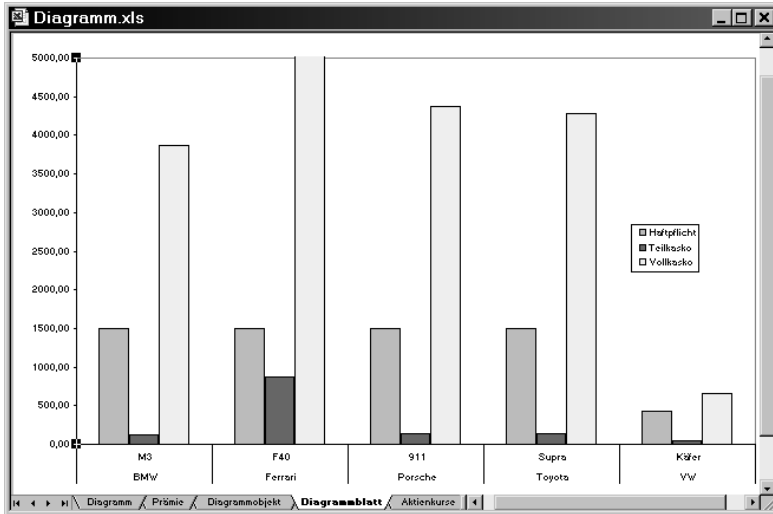
Im Normalfall ist das auch sinnvoll. Eine Ausnahme liegt vor, wenn eine Datenpunktreihe einen außergewöhnlich kleinen oder großen Wert enthält, wie im Beispiel die Vollkaskoprämie für den Ferrari F40. Dadurch ergeben sich Diagramme, die zwar auch diesen Extremwert korrekt darstellen, in denen jedoch die Differenzen zwischen den restlichen, wesentlich kleineren Werten einfach untergehen.

Um das zu ändern, sollte für *Maximum* ohne Berücksichtigung des Extremwerts zum Beispiel entsprechend der vorhergehenden Abbildung 5000 gewählt werden (Abbildung 24.39).

Dieses Diagramm ist eigentlich falsch. Denn die Säule für den F40 reicht nur noch bis zum Wert 5000, dem höchsten im Diagramm dargestellten – obwohl die zugehörige Prämie tatsächlich weitaus höher ist. Dafür ist nun ein vernünftiger Vergleich der restlichen Prämiensätze möglich.

Dass die Säule abgeschnitten ist, zeigt Excel durch die fehlende Rahmenlinie an der oberen Begrenzung der betreffenden Säule an.

Abbildung 24.39:
Abschneiden von
Extremwerten



**Teilungsstrich-
abstände**

Mit *Hauptintervall* und *Hilfsintervall* bestimmen Sie den Abstand zwischen den Teilungsstrichen auf der Größenachse.

Die Vorgabe hängt vom kleinsten bzw. größten dargestellten Wert ab. Ersetzen Sie die Hauptstrichvorgabe zum Beispiel durch 1000, erscheinen doppelt so viele Hauptstriche, nämlich bei den Werten 0, 1000, 2000, 3000 und so weiter.

Entsprechend bestimmt *Hilfsintervall* die Anzahl und den Abstand der – nicht beschrifteten – Hilfsstriche. Wurden diese im Register *Muster* mit irgendeiner Option außer *Keine* eingeblendet, erscheinen zwischen zwei Hauptstrichen fünf Hilfsstriche.

Eine Erhöhung des Standardwerts zeigt entsprechend mehr bzw. eine Verringerung weniger Hilfsstriche an (Abbildung 24.40).

Für diese Abbildung wurde für Hauptintervall 1000 und für Hilfsintervall 500 angegeben und im Register *Muster* für die Hilfsstriche die Option *Außen* gewählt, sodass sie auf der Diagrammaußenseite angezeigt werden.

**Achsenschnitt-
punkte**

Die Standardvorgabe für *Rubrikenachse (X)* *schneidet bei* ist 0. Interessiert Sie, welche Prämien höher als 2000 DM sind und welche niedriger, dann setzen Sie diesen Wert auf 2000 und sehen danach auf einen Blick, dass alle Haftpflicht- und Teilkaskoprämien kleiner sind als 2000, dafür jedoch alle Vollkaskoprämien höher (Abbildung 24.41).

**Einheiten
anzeigen**

Unter *Einheiten anzeigen* ist normalerweise *Keine* vorselektiert. Wählen Sie eine andere Einheit wie *Tausende*, wird auf der Größenachse »12000,00« als »12,00« dargestellt, »11000,00« als »11,00« etc.

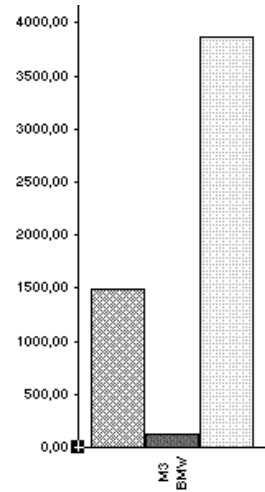


Abbildung 24.40:
Teilungsstriche

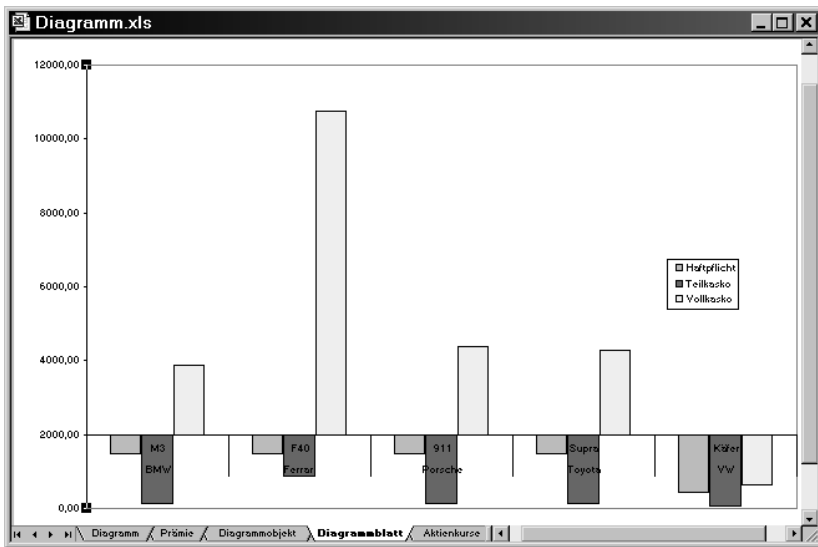


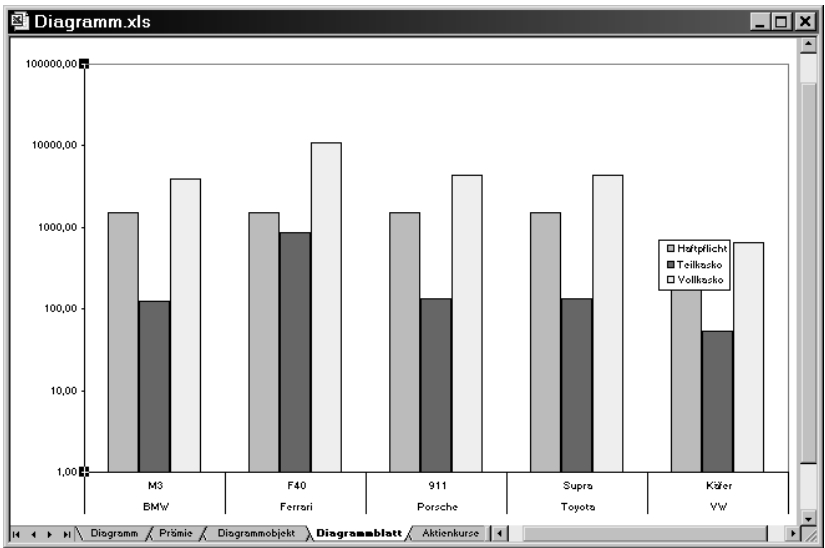
Abbildung 24.41:
Schnittpunkt von
Größen- und
Rubrikenachse
festlegen

Dann sollten Sie jedoch auch *Beschriftung im Diagramm anzeigen* aktivieren, da Excel in diesem Fall den zugehörigen Text, beispielsweise *Tausende*, als Beschriftung der Größenachse einblendet.

Normalerweise ist die Größenachse linear unterteilt und die Höhe einer Säule wächst also linear mit der Größe des zugehörigen Wertes. Bei extrem unterschiedlichen Werten ist diese Skalierung jedoch nicht sehr aussagekräftig, wie wir am Beispiel der Ferrari F40-Vollkaskoprämie sahen. Sinnvoller ist in diesem Fall eine logarithmische Skalierung (Abbildung 24.42).

**Logarithmische
Skalierung**

Abbildung 24.42:
Logarithmische
Skalierung



Der Extremwert 100746 ist nicht mehr abgeschnitten, sondern wird korrekt dargestellt. Dennoch ist dank der logarithmischen Skalierung ein vernünftiger Vergleich aller Werte möglich.

Im gleichen Moment, in dem Sie *Logarithmische Skalierung* aktivieren, passt Excel alle Skalierungsoptionen an diese Skalierungsart an (Abbildung 24.43).

Abbildung 24.43:
Automatisch
vorgenommene
Anpassungen



Als Minimum verwendet Excel nun den Wert 1 und als Maximum die kleinste Zehnerpotenz, die größer als der höchste Wert ist. Also 100000, da in unserem Fall der größte Wert 10746 ist und die nächstkleinere Zehnerpotenz 10000 somit kleiner wäre.

Der Wert für das Hauptintervall muss eine – ganzzahlige – Zehnerpotenz sein, also 1, 10, 100, 1000 oder ähnlich.

Die Vorgabe 10 bedeutet, dass jeder weitere Hauptstrich einen Wert darstellt, der um den Faktor 10 höher ist als der Wert des letzten Teilstrichs. Würden Sie die 10 in 100 ändern, würde Excel nur noch drei Teilstriche anzeigen, nämlich jene, die den Werten 100 (Minimum), 10000 und 100000 (Maximum) entsprechen.

Der Wert für das Hilfsintervall muss ein Zehntel oder die Hälfte des Hauptintervalls betragen, bei einem Hauptintervall von 100 zum Beispiel 10, mit dem Resultat, dass ein Haupt- in zehn Hilfsintervalle unterteilt ist.

Größen in umgekehrter Reihenfolge bewirkt eine Spiegelung der Rubrikenachse und eine Umkehrung der Größenachsenskalierung (Abbildung 24.44).

*Größen in
umgekehrter
Reihenfolge*

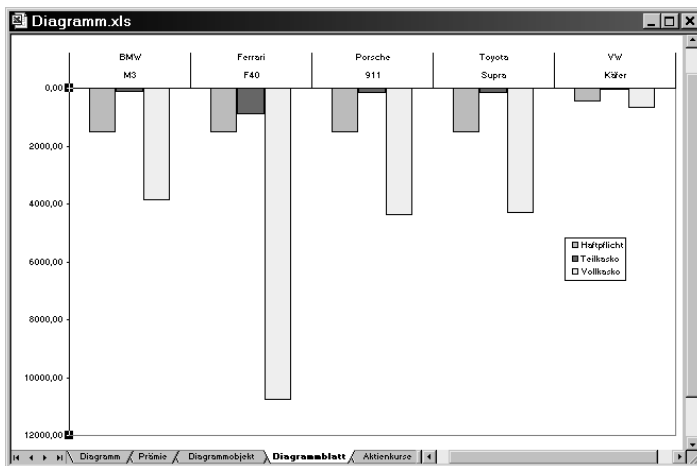
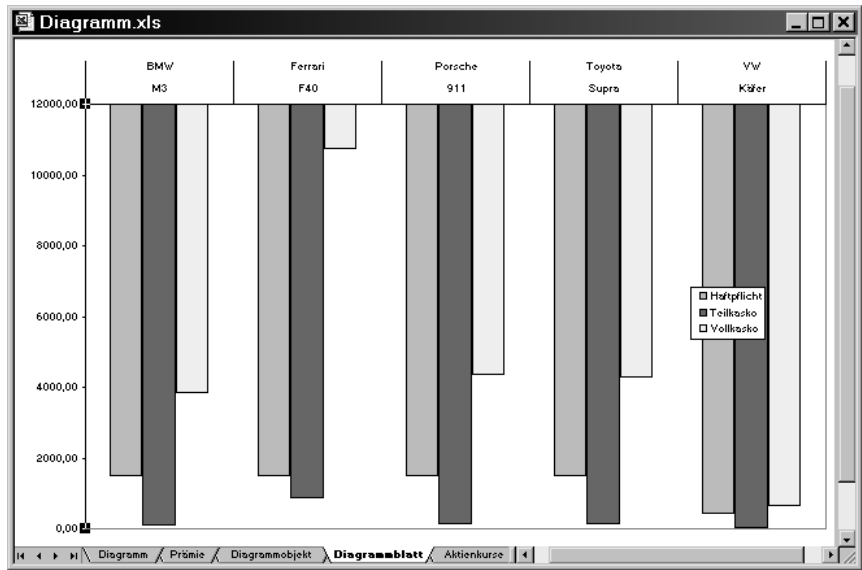


Abbildung 24.44:
Umkehrung der
Größenachse

Eine ähnliche Darstellung ergibt sich, wenn Sie *Größen in umgekehrter Reihenfolge* wieder deaktivieren und stattdessen *Rubrikenachse (X) schneidet bei Höchstwert* aktivieren (Abbildung 24.45).

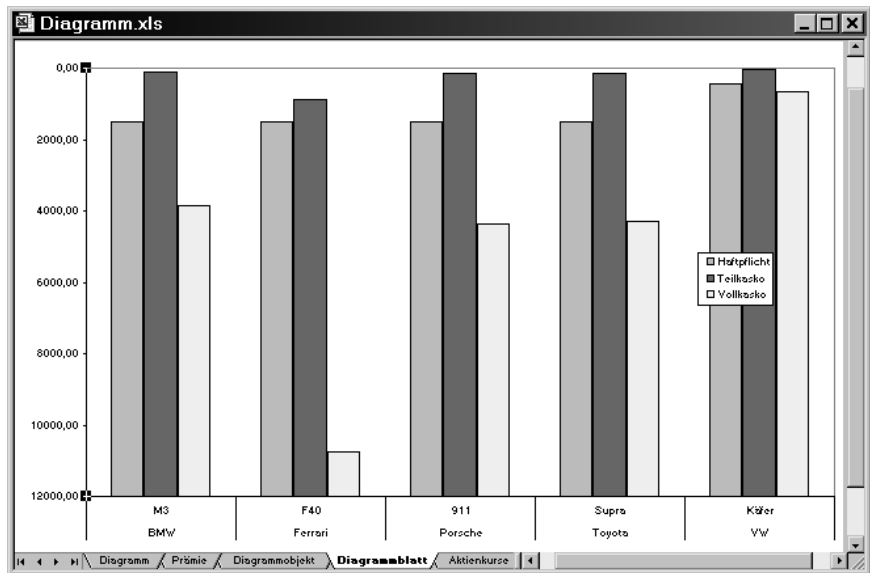
Die Größenachse verläuft wie üblich von unten nach oben vom kleinsten zum größten Wert. Die Rubrikenachse ist jedoch erneut gespiegelt, da sich ihr Schnittpunkt mit der Größenachse diesmal ja beim höchsten Wert befinden soll, mit dem Ergebnis, dass den höchsten Werten die kürzesten Säulen entsprechen und umgekehrt.

Abbildung 24.45:
Schnittpunkte mit
der Rubrikenachse
bei Höchstwert



Aktivieren Sie *Größen in umgekehrter Reihenfolge* und gleichzeitig(!) *Rubrikenachse (X) schneidet bei Höchstwert*, so wird das Diagramm wieder vom Kopf auf den Fuß gestellt, die Invertierung der Säulenlängen jedoch beibehalten (Abbildung 24.46).

Abbildung 24.46:
Beide Optionen
gleichzeitig



24.9.5 Skalierung der Rubrikenachse

Selektieren Sie die Rubrikenachse, so enthält das Register *Skalierung* folgende Optionen (Abbildung 24.47).

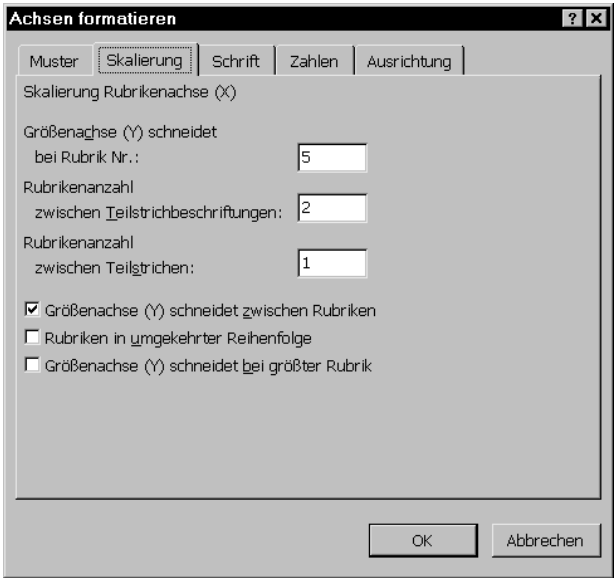


Abbildung 24.47:
Skalierung der
Rubrikenachse

Normalerweise enthalten alle drei Eingabefelder den Wert 1. Ändern Sie die beiden ersten Werte wie in der Abbildung in 5 beziehungsweise in 2, so erhalten Sie folgendes Diagramm (Abbildung 24.48).

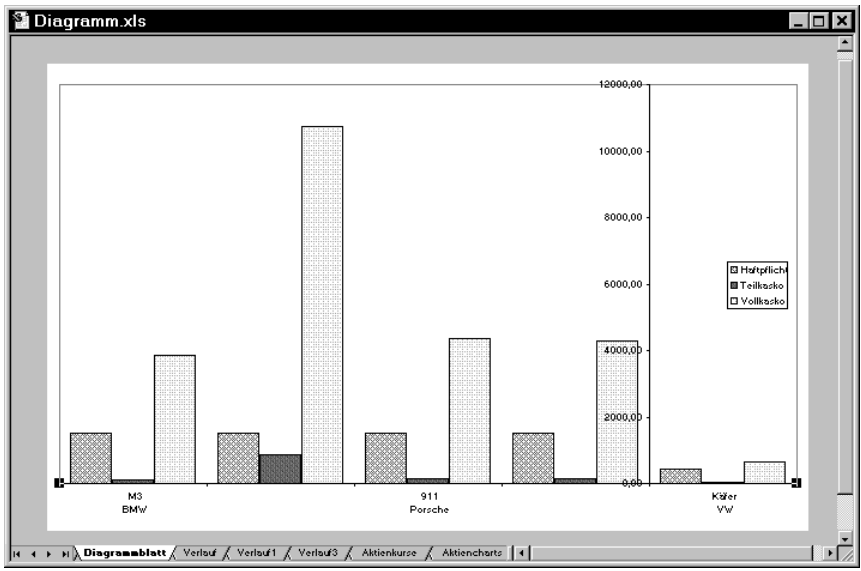


Abbildung 24.48:
Achsenschnitt-
punkt verschieben,
beschriftete
Rubrikenanzahl
bestimmen

**Achsen-
schnittpunkt**

Größenachse... bestimmt, wo sich der Schnittpunkt zwischen der Rubriken- und der Größenachse befindet, vor welcher Rubrik.

Normalerweise (Standardwert 1) befindet er sich vor der ersten Rubrik, also am linken Ende der Rubrikenachse. Mit dem Wert 5 wird er entsprechend vor die fünfte Rubrik »VW Käfer« verschoben (was nicht als Ausgrenzung dieses charaktervollen Fahrzeugs verstanden werden soll).

**Rubriken-
beschriftung**

Die zweite Option entscheidet darüber, ob alle oder nur einige der Rubriken beschriftet werden. Der Standardwert 1 bedeutet, dass jede Rubrik beschriftet wird. Der hier verwendete Wert 2 bedeutet, dass – siehe Abbildung – nur jede zweite Rubrik beschriftet wird; bei 3 wäre es stattdessen nur jede dritte etc.

**Rubrikenteil-
striche**

Normalerweise befindet sich auf der Rubrikenachse pro Rubrik genau ein Teilstrich (Standardwert 1). Ähnlich wie bei der vorhergehenden Option bewirkt der Wert 2 für *Rubrikenanzahl zwischen Teilstrichbeschriftungen*, dass nur bei jeder zweiten Rubrik ein Teilstrich vorhanden sein wird, mit einer 3 nur bei jeder dritten Rubrik und so weiter.

Die Option *Größenachse (Y) schneidet zwischen Rubriken* ist aktiviert. Was dazu führt, dass der Schnittpunkt der beiden Achsen zwischen zwei Rubriken verläuft.

Würden Sie diese Option deaktivieren, verlief die Größenachse anschließend mitten durch die Rubrik »VW Käfer« und zwar durch die mittlere Säule »Teilkasko« dieser Rubrik.

Die Aufgabe der Option *Rubriken in umgekehrter Reihenfolge*: Die Reihenfolge der Rubriken von links nach rechts wird umgekehrt, sodass sich beispielsweise »VW Käfer« nun am linken statt am rechten Ende der Rubrikenachse befindet. Die Größenachse wird dabei gespiegelt und befindet sich anschließend am rechten Ende der Rubrikenachse statt am linken.

Bei *Größenachse (Y) schneidet bei größter Rubrik* geht es keineswegs um die Rubrik mit den höchsten Säulen oder Balken, sondern um die Rubrik mit dem höchsten Index.

Dazu müssen Sie wissen, dass die Rubrik ganz links für Excel den Index 1 besitzt, die Rubrik rechts daneben den Index 2 etc. Normalerweise schneidet die Größen- die Rubrikenachse bei der Rubrik mit dem kleinsten Index, also am linken Ende der Rubrikenachse. Wird *Größenachse (Y) schneidet bei größter Rubrik* aktiviert, liegt der Schnittpunkt dagegen bei der Rubrik mit dem höchsten Index, also am rechten Ende der Rubrikenachse.

Aktivieren Sie *Rubriken in umgekehrter Reihenfolge* und gleichzeitig (!) *Größenachse (Y) schneidet bei größter Rubrik*, wird somit die Rubrikenreihenfolge umgekehrt, die Größenachse bleibt jedoch weiter am linken Diagrammrand und wird nicht gespiegelt (Abbildung 24.49).

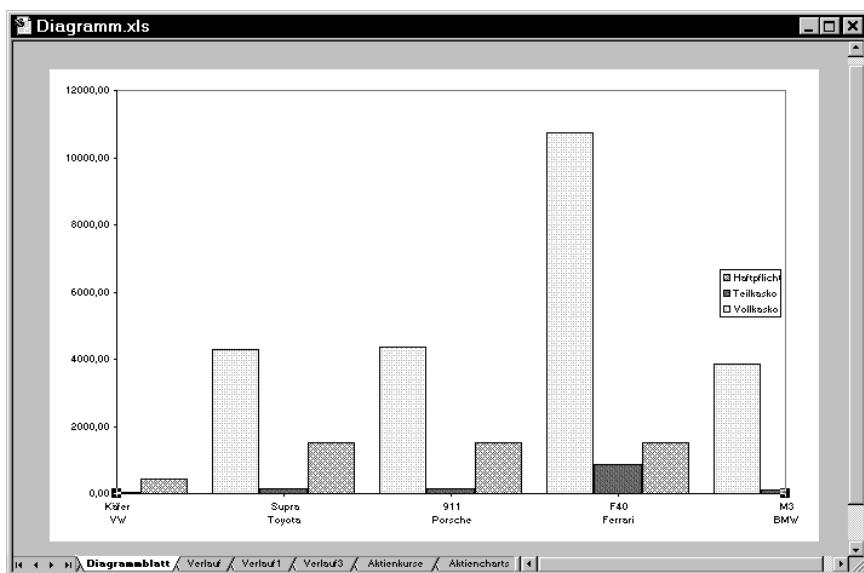
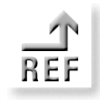


Abbildung 24.49:
Rubrikenreihenfolge umkehren und Größenachse dabei nicht spiegeln

Teil 5 Excel und das Internet

Kapitel 25: Hyperlinks	699
Kapitel 26: Excel-Tabellen im Internet/Intranet	711
Kapitel 27: Webseiten	721
Kapitel 28: Webabfragen	735
Kapitel 29: Webserver einrichten	745
Kapitel 30: Sonstiges	759



Nicht alles, was in irgendeiner Weise mit dem Internet zu tun hat, finden Sie im vorliegenden Kapitel, da manche entfernt mit dem Internet zusammenhängende Themen in anderen Kapiteln besser untergebracht sind. Beispielsweise könnte man theoretisch auch das Versenden und Weiterleiten von Excel-Mappen per E-Mail zum Internet zählen. Stattdessen wird dieses Thema jedoch in Kapitel 8, »Arbeitsmappen und Vorlagen«, besprochen, in das es sich weit harmonischer einfügt (siehe Kapitel 8.1.1, »Mappen versenden und weiterleiten«).

Im folgenden Teil erläutere ich zunächst Hyperlinks, das heißt Verweise zu Dokumenten, die sich praktisch an einem beliebigen Ort befinden können, außer auf Ihrer Festplatte beispielsweise auch auf der eines Kollegen, dessen Rechner mit Ihrem im firmeninternen Intranet verbunden ist.

Ich zeige Ihnen zunächst, wie Hyperlinks erstellt und benutzt werden. Anschließend öffnet das Anklicken des Hyperlinks automatisch das betreffende Dokument, beispielsweise eine andere Excel-Tabelle im Internet.

Wie Sie sehen werden, können Hyperlinks nicht nur auf Excel-Tabellen verweisen, sondern auf praktisch beliebige Dokumenttypen. Zum Beispiel auf Word-Dokumente: dann wird durch Klicken auf den Hyperlink Word aufgerufen und das betreffende Dokument darin angezeigt.

Hyperlinks können auch auf FTP- und HTTP-Adressen verweisen. In beiden Fällen handelt es sich um die Adresse einer Datei oder eines Dokuments, das sich im Internet befindet. Zu deren Anzeige wird nach Aktivierung des Hyperlinks Ihr Webbrowser aufgerufen, der daraufhin die Verbindung zum betreffenden Server herstellt und die Seite anzeigt.

Das Umherspringen zwischen verschiedenen Dokumenten durch Anklicken von Hyperlinks steuern Sie mit Hilfe der Web-Leiste, die praktisch all jene Symbole enthält (Vor- und Zurückblättern, Springen zur Startseite (Homepage) und so weiter), die Sie bereits von Ihrem Webbrowser her kennen.

Formeln in Excel-Tabellen können Verknüpfungen zu im Internet auf FTP-Servern gespeicherten Tabellen enthalten und Sie können sogar selbst Excel-Mappen im Internet speichern und öffnen.

Danach zeige ich Ihnen, wie Sie in Webseiten enthaltene Tabellen in Ihre Excel-Tabellen importieren und umgekehrt Ihre Excel-Tabellen als Webseiten speichern, um Sie anschließend im Internet zu publizieren; oder aber im firmeninternen Intranet, beispielsweise um auch Kollegen, die nicht über Excel verfügen, das Anschauen Ihrer Tabellen und Analysen zu ermöglichen.

Sie können aus einer Excel-Tabelle sogar ein Webformular erstellen, das es Ihren Kunden ermöglicht, eine Artikelbestellung per Internet aufzugeben, indem im zugehörigen Webformular alle dazu notwendigen Angaben erfasst werden.

Mit Webabfragen ist es sogar möglich, von Excel aus Daten zu importieren, die sich irgendwo im Internet in einer Webseite befinden. Wie Sie sehen werden, kann Excel alle in einer solchen Webseite enthaltenen Tabellen importieren bzw. – wenn die Webseite keine Tabellen enthält – alternativ dazu einfach den gesamten Seiteninhalt.

Zum Schluss zeige ich Ihnen, wie Sie selbst einen einfachen Webserver einrichten und was dabei zu beachten ist.

25 Hyperlinks

Sie kennen Hyperlinks bereits aus der Excel-Hilfe. Ein Hyperlink ist ein Verweis auf ein beliebiges Dokument auf Ihrer Festplatte oder einem sonstigen Speichermedium, auf das Sie Zugriff besitzen. Klicken Sie auf einen Hyperlink, erfolgt ein Sprung zum betreffenden Dokument, so als hätten Sie es geöffnet und den Cursor darin zu jener Stelle bewegt, auf die der Hyperlink verweist.

25.1 Hyperlinkfelder

Nehmen wir an, Sie arbeiten ständig mit einer bestimmten Arbeitsmappe und darin wiederum mit einer bestimmten Tabelle. Ruft Ihr höchst ungeduldiger Chef an, benötigen Sie sofortigen Zugriff auf eine ebenfalls in dieser Mappe vorhandene Tabelle »Adressen«, um ihm ohne Verzögerung darin enthaltene Daten durchgeben zu können.

Dazu fügen Sie Ihrer ständig benutzten Tabelle einfach einen Verweis auf die Tabelle »Adressen« hinzu, um durch Anklicken des Verweises sofort zu dieser springen zu können (Abbildung 25.1).

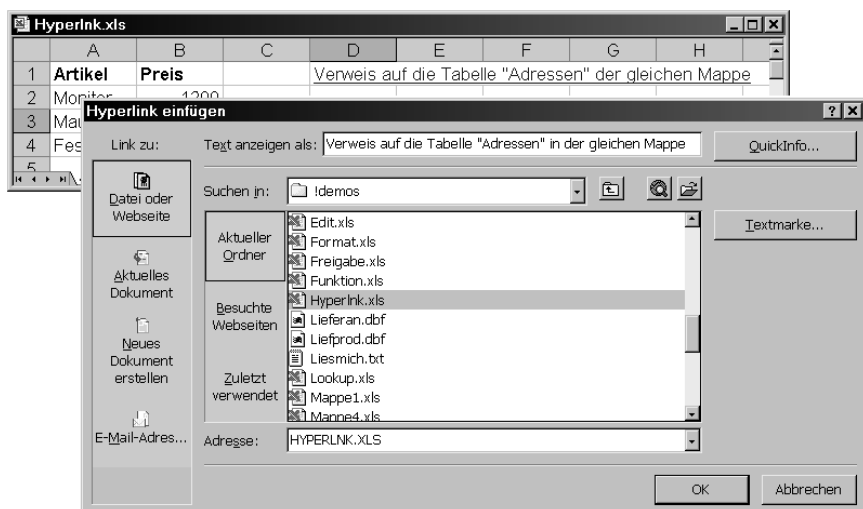


Abbildung 25.1:
Hyperlink.xls,
Tabelle »Artikel«

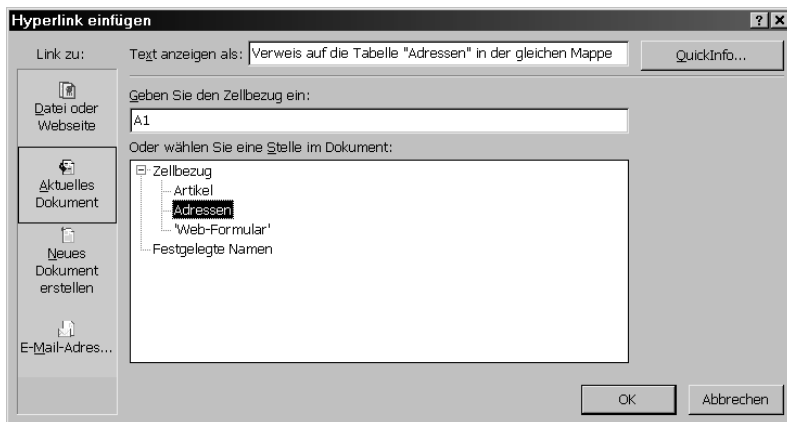
**Hyperlink
einfügen**

Um den Verweis auf die Tabelle einzufügen, gab ich zunächst in D1 den Text »Verweis auf die Tabelle »Adressen« der gleichen Mappe« ein. Um der Zelle einen Hyperlink hinzuzufügen, wählen Sie im Zellkontextmenü HYPERLINK... oder in der Menüleiste EINFÜGEN|HYPERLINK... .

Ist im linken Dialogfeldteil *Datei oder Webseite* selektiert, so können Sie einen Hyperlink einzufügen, der auf eine beliebige Datei oder Webseite verweist.

Um stattdessen einen Verweis auf eine Zelle der momentan geöffneten Mappe als Ziel des Hyperlinks festzulegen, aktivieren Sie im linken Dialogfeldteil *Aktuelles Dokument* (Abbildung 25.2).

Abbildung 25.2:
Hyperlink einfügen

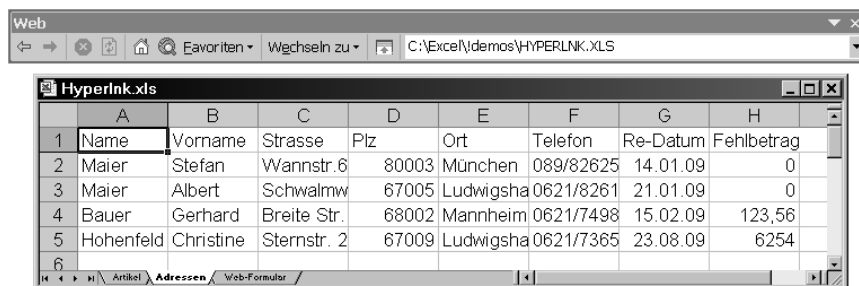


Der einzige Unterschied: Die Schaltfläche *Textmarke...* öffnet ein eigenes Dialogfeld, in dem jedoch die gleichen Informationen einzugeben sind:

- ➔ *Text anzeigen als:* Ein beliebiger beschreibender Text, wobei Excel den Inhalt der aktuellen Zelle vorgibt
- ➔ *Geben Sie den Zellbezug ein...:* die Zielzelle
- ➔ *Oder wählen Sie eine Stelle im Dokument:* In diesem Feld werden alle im aktuellen Dokument vorhandenen Namen aufgelistet, unter *Zellbezug* die Tabellennamen »Artikel«, »Adressen« und »Webformular« der aktuellen Mappe, unter *Festgelegte Namen* alle benannten Bereiche.

Wählen Sie eine Tabelle wie »Adressen« aus, wird als Zellbezug automatisch A1 vorgegeben, also die linke obere Ecke der Tabelle. Soll der Sprung zu einer anderen Zelle in der Tabelle führen, ändern Sie diese Vorgabe entsprechend.

Nach dem Schließen mit OK genügt ab jetzt das Anklicken des Hyperlinks, um jederzeit zu Zelle A1 der Tabelle »Adressen« zu springen (Abbildung 25.3).



The screenshot shows a web browser window with the address bar displaying 'C:\Excel\demos\HYPERLINK.XLS'. The browser displays a table from an Excel spreadsheet named 'Hyperlink.xls'. The table has 8 columns: Name, Vorname, Strasse, Plz, Ort, Telefon, Re-Datum, and Fehlbetrag. The data rows are as follows:

	A	B	C	D	E	F	G	H
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetrag
2	Maier	Stefan	Wannstr.6	80003	München	089/82625	14.01.09	0
3	Maier	Albert	Schwalmw	67005	Ludwigsha	0621/8261	21.01.09	0
4	Bauer	Gerhard	Breite Str.	68002	Mannheim	0621/7498	15.02.09	123,56
5	Hohenfeld	Christine	Sternstr. 2	67009	Ludwigsha	0621/7365	23.08.09	6254
6								

Abbildung 25.3:

Sprung zum
Hyperlink-Ziel

Dazu müssen Sie die Zelle D1 von »Artikel« anklicken, in der sich der Hyperlink befindet. Anklicken der benachbarten und nur zur Anzeige des überlängten Textes mitbenutzten Zellen ist wirkungslos!



Hyperlinks werden farblich und unterstrichen hervorgehoben. Aktivieren Sie erneut die Tabelle »Artikel«, so stellen Sie fest, dass sich die Farbe des soeben darin angeklickten Hyperlinks geändert hat. Dadurch wird Ihnen mitgeteilt, dass Sie den Hyperlink bereits benutzten und daher wissen, was sich hinter diesem Verweis verbirgt.

Sie können auch ein Excel- oder ein eingebettetes Grafikobjekt als Hyperlink verwenden. Dazu selektieren Sie das Grafikobjekt und wählen danach wie bei Zellen EINFÜGEN|HYPERLINK.



Sie können auch Symbolen oder Befehlen Hyperlinks zuweisen (siehe Kapitel 38.1, »Menü- und Symbolleisten anpassen«).



Sie können einen Hyperlink auch mit Hilfe der Funktion *Hyperlink* einfügen, der zwei Argumente zu übergeben sind:

**Hyperlink-
Funktion**

`=Hyperlink(Hyperlink_Adresse;Freundlicher_Name)`

Hyperlink_Adresse ist der Name und Pfad einer beliebigen Datei auf Ihrer Festplatte oder der eines im LAN damit verbundenen Rechners, beispielsweise einer Excel-Tabelle:

`=Hyperlink("e:\excel\!demos\zinsen.xls";"Excel-Mappe <zinsen.xls>")`

Klicken auf den mit dieser Formel erzeugten Hyperlink so wird der zur Mappe ZINSEN.XLS springen, die sich im Verzeichnis EXCEL\!DEMOS von E: befindet.

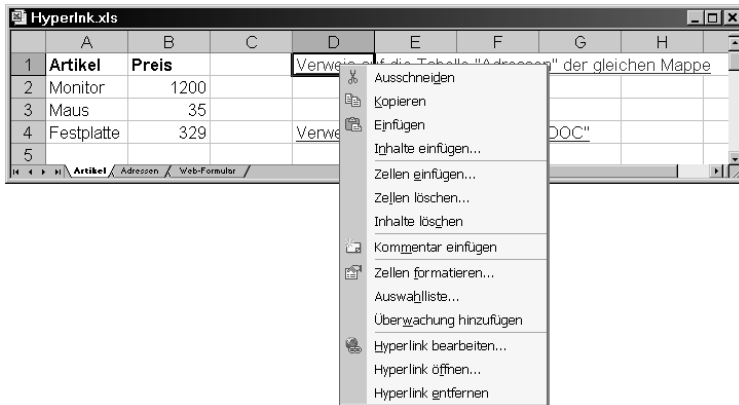
Stattdessen können Sie auch die Adresse einer Webseite wie *http://www.microsoft.com* angeben, die Microsoft-Homepage.

Freundlicher_Name ist der Hinweistext, der in der Zelle angezeigt werden soll, die die *Hyperlink*-Funktion enthält.

25.2 Hyperlinks bearbeiten

Nach dem Einfügen eines Hyperlinks in eine Zelle enthält das Zellkontextmenü alle zum nachträglichen Bearbeiten des Hyperlinks benötigten Befehle (Abbildung 25.4).

Abbildung 25.4:
Hyperlink
bearbeiten



- ➔ HYPERLINK BEARBEITEN... öffnet das Dialogfeld zur Angabe der Zieladresse, die Sie nun ändern können.
- ➔ HYPERLINK ÖFFNEN... besitzt die gleiche Funktion wie das Anklicken des Hyperlinks, springt also zur dadurch definierten Adresse.
- ➔ HYPERLINK ENTFERNEN löscht den Hyperlink (der Inhalt der Zelle bleibt natürlich erhalten).

25.3 Die Web-Symboleiste

Die Tabelle »Adressen« kann selbst wieder Hyperlinks zu beliebigen anderen Excel-Tabellen oder sonstigen Dokumenten enthalten. Die Umherspringerei von Hyperlink zu Hyperlink kontrollieren Sie mit Hilfe der automatisch eingeblendeten Webleiste (siehe Abbildung 25.3).

Die darin enthaltenen Symbole kennen Sie vermutlich bereits von Ihrem Webbrowser her:



Zur zuletzt eingesehenen Seite zurückblättern, egal ob das eine Webseite oder eine lokale Excel-Mappe auf Ihrer Festplatte ist



Wieder vorwärts blättern

Den aktuellen Sprung abbrechen (nur im Web mit seinen langen »Sprungzeiten« interessant, nicht auf der lokalen Platte)



Die gegenwärtig angezeigte Seite aktualisieren



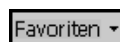
Zur Startseite springen



Das Web anhand von Suchbegriffen nach Seiten mit bestimmten Informationen durchsuchen



Die Unterbefehle dieses Symbols ermöglichen Ihnen den schnellen Zugriff auf häufig benutzte Dateien oder Webseiten über den Ordner FAVORITEN (siehe Kapitel 3.2, »Öffnen«):



- ➔ *Zu Favoriten hinzufügen...* fügt in den Ordner FAVORITEN eine Verknüpfung ein, die die Adresse des aktuellen Objekts beinhaltet. Dabei kann es sich um die aktuelle Arbeitsmappe handeln oder um eine momentan angezeigte Webseite.
- ➔ *Favoriten öffnen...* zeigt den Inhalt dieses Ordners an, also alle darin abgelegten Verknüpfungen; doppelklicken Sie auf eine davon, wird die betreffende Mappe geöffnet bzw. die Webseite angezeigt.
- ➔ Unter diesen beiden Befehlen wird jede im Ordner FAVORITEN abgelegte Adresse zusätzlich als eigener Unterbefehl angezeigt (analog zum Menü FENSTER, das ja ebenfalls für jedes geöffnete Fenster einen zusätzlichen Befehl enthält). Das Anklicken eines dieser Befehle genügt, um zur zugehörigen Adresse zu springen.

Dieses Menü enthält die besprochenen Symbole noch einmal als Befehle und darüber hinaus einige zusätzliche Befehle:



- ➔ *HYPERLINK ÖFFNEN...* öffnet ein Dialogfeld, in dem Sie eine Sprungadresse eingeben und per OK dorthin springen können.
- ➔ *STARTSEITE BESTIMMEN...* ermöglicht, die momentan angezeigte Seite, in unserem Fall also die aktuelle Tabelle der gerade geöffneten Excel-Mappe, zur Startseite Ihres Webbrowsers zu machen, die nach dessen Aufruf automatisch angezeigt wird.
- ➔ *SUCHSEITE BESTIMMEN...* ermöglicht, die momentan angezeigte Seite zur Suchseite Ihres Webbrowsers zu machen, die zur Suche im Internet verwendet wird.

Alle Symbolleisten außer der Webleiste ausblenden (bzw. danach wieder einblenden)





Eine Sprungadresse eingeben oder eine der bereits vorhandenen (zuletzt benutzten) auswählen

25.4 Links zu beliebigen Dokumenten

Das Dialogfeld zum Einfügen eines Hyperlinks ermöglicht die Angabe beliebiger Dokumente und Orte als Sprungziel.

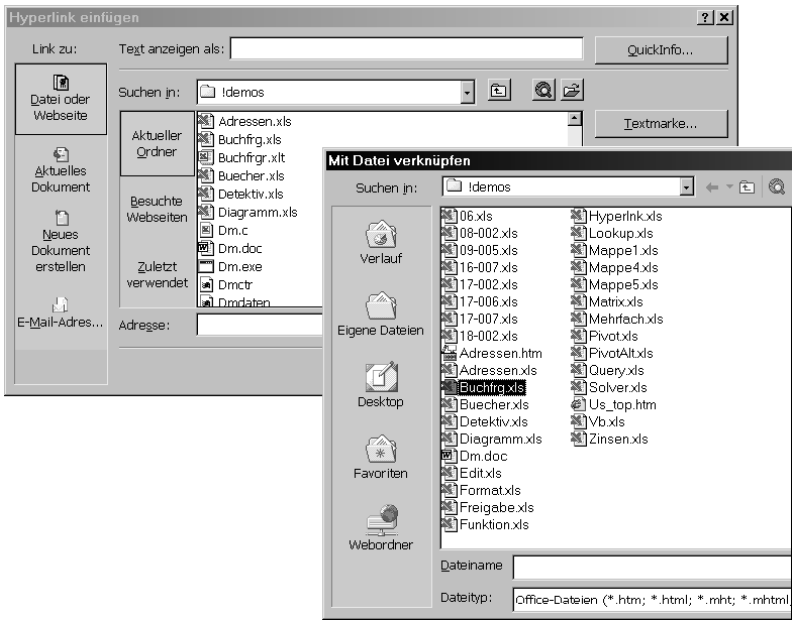
Ist im linken Register *Aktuelles Dokument* aktiviert, können Sie wie erläutert eine Zelle oder einen benannten Bereich innerhalb der aktuellen Arbeitsmappe angeben.

25.4.1 Dateien

Excel-Mappen

Die Schaltfläche *Datei oder Webseite* ermöglicht stattdessen die Angabe eines beliebigen Dokuments (Abbildung 25.5).

Abbildung 25.5:
Zielfdatei auswählen



Handelt es sich um eine Datei auf der Festplatte Ihres oder eines damit per Netzwerk verbundenen Rechners, so aktivieren Sie die Schaltfläche *Aktueller Ordner*. Das Listenfeld ähnelt nun dem gewohnten *Dateiauswahl*-Dialogfeld und Sie können die ausgewählte Datei interaktiv selektieren.

Alternativ dazu können Sie auch auf das Symbol zum Öffnen eines Ordners klicken. Dann erscheint das gewohnte *Dateiauswahl*-Dialogfeld als eigenständiges Fenster.

Im Beispiel wählte ich als Ziel die Arbeitsmappe BUCHFRG.XLS aus, worauf nach dem Schließen mit OK der Pfad zu dieser Datei (falls nicht im aktuellen Verzeichnis enthalten) und der Dateiname unter *Adresse* eingefügt werden. (Sie können diese Angaben dort natürlich auch manuell eingeben, statt die Datei interaktiv zu selektieren.

Handelt es sich um eine Excel-Mappe, können Sie anschließend mit *Textmarke...* ein Dialogfeld öffnen, in dem Sie die Zieladresse näher bestimmen und wie erläutert die fragliche Tabelle und Zelle jener Mappe angeben können (Abbildung 25.6).

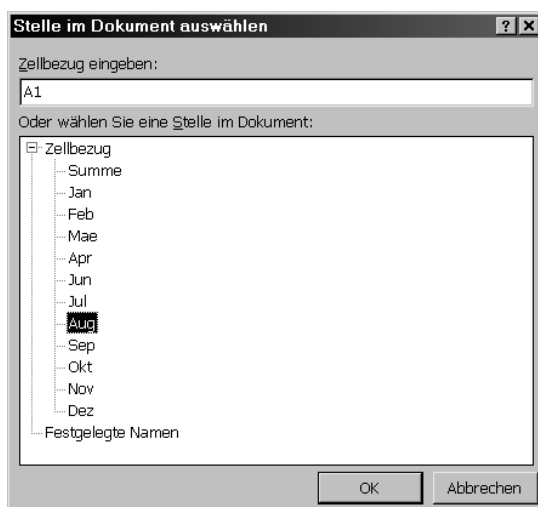


Abbildung 25.6:
Zielzelle angeben

Hyperlinks können auf beliebige Dokumente verweisen, außer auf Excel-Mappen beispielsweise auf Word-Dateien (Abbildung 25.7).

Andere Dateien

Eine genauere Zielbestimmung ist wie erläutert zwar nur in Excel-Dateien möglich, aber zumindest wird nach dem Klicken auf den Hyperlink Word aufgerufen und darin die Textdatei geöffnet (Abbildung 25.8).

Allgemein: Hyperlinks können auf beliebige Dokumente verweisen, vorausgesetzt, ein zugehöriges Anwendungsprogramm ist installiert, das in der Lage ist, Dateien des betreffenden Typs zu öffnen und anzuzeigen!



Wo sich die betreffende Datei befindet, ist gleichgültig: Sie kann sich

- ➔ auf Ihrer lokalen Festplatte befinden,
- ➔ auf der Festplatte eines Kollegen, dessen Rechner mit Ihrem im Firmennetzwerk verbunden ist
- ➔ oder im Internet auf einem FTP-Server (siehe Kapitel 26, »Excel-Tabellen im Internet/Intranet«).



Abbildung 25.7:
Hyperlink auf ein
Word-Dokument

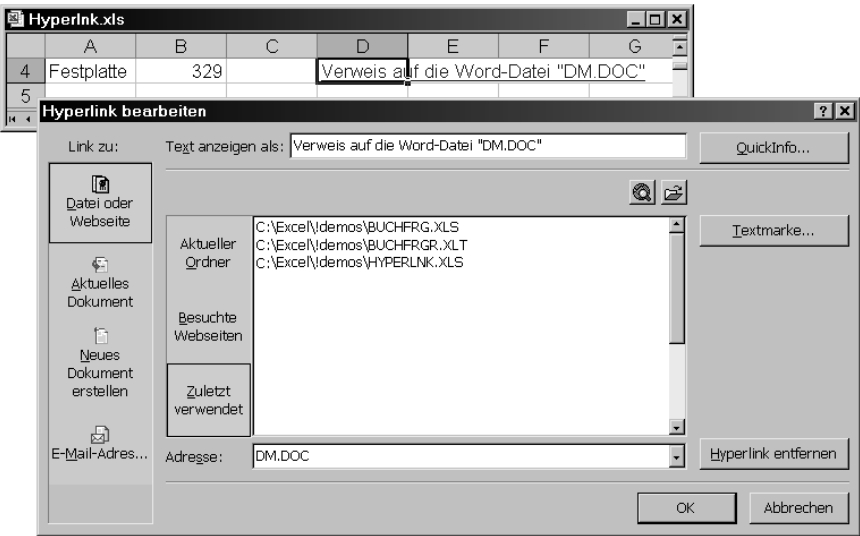
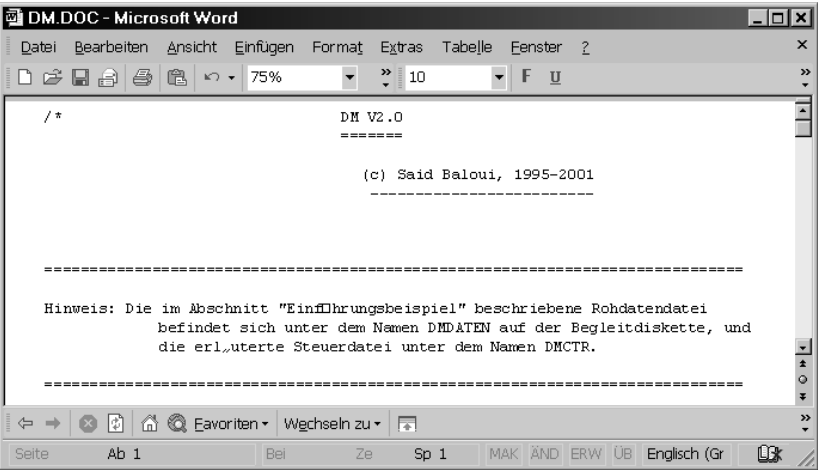


Abbildung 25.8:
Word-Dokument
anzeigen



Allgemein: Haben Sie Zugriff auf die betreffende Datei, können Sie sie also im Dateiauswahldialogfeld anzeigen, dann können Sie auch einen Hyperlink erzeugen, der auf diese Datei verweist!

Zuletzt verwendet

Wollen Sie auf eine erst kürzlich geöffnete Datei verweisen, klicken Sie auf die Schaltfläche *Zuletzt verwendet*: Im Listefeld werden die zuletzt geöffneten Dateien angezeigt und Sie wählen die gewünschte per Anklicken aus.

Excel fügt daraufhin in *Adresse* Pfad und Name der Datei ein und Sie geben nur noch im oberen Textfeld einen passenden Kommentar ein.

25.4.2 Webseiten

Hyperlinks können statt auf Dateien auch auf Webseiten verweisen.

Sie können die Adresse der Sie interessierenden Webseite im Feld *Adresse* direkt eintippen, beispielsweise »c:\test.htm« wenn Sie sich für die Webseite TEST.HTM im Stammverzeichnis Ihrer Festplatte interessieren, oder »http://www.microsoft.com«, wenn Sie einen Hyperlink zur Microsoft-Homepage erstellen wollen.

Besuchten Sie die Webseite kürzlich, so klicken Sie auf *Besuchte Webseiten*. Das Listenfeld enthält nun alle zuletzt von Ihnen besuchten Webseiten und Sie suchen die gewünschte aus.

Sie können die Adresse auch im Listenfeld *Adresse* auswählen, das ebenfalls eine Liste der zuletzt besuchten Webseiten enthält, die in exakt der gleichen Form dargestellt wird wie in der Adressleiste Ihres Webbrowsers.

Klicken Sie stattdessen auf das abgebildete Symbol, so wird ihr Webbrowser gestartet und Sie können die Webseite, auf die der Hyperlink verweisen soll, interaktiv suchen (Abbildung 25.9).



Abbildung 25.9:
Web durchsuchen

Hier wird gerade eine Webseite dargestellt, die sich auf dem Server »haupt« befindet (einem Webserver in einem Intranet, einem Firmennetzwerk).

Kehren Sie anschließend zu Excel zurück, ohne den Browser zu schließen, so wird darin die betreffende Adresse angezeigt.

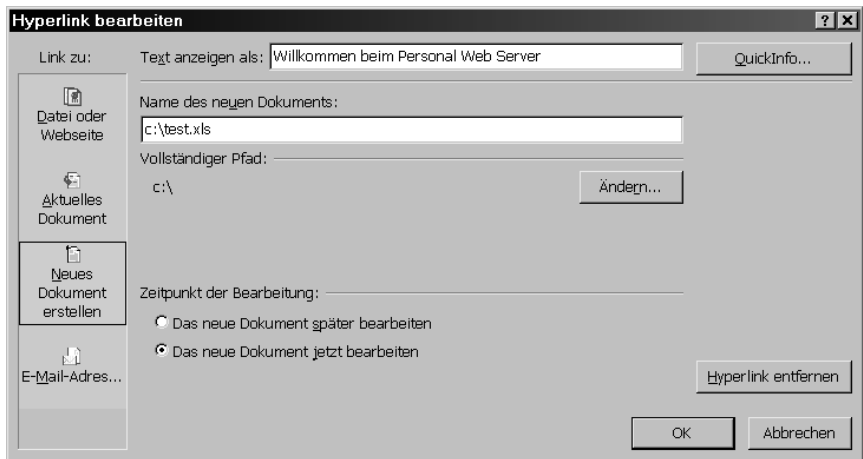
Nach Auswahl einer Webseite und OK fügt Excel den Hyperlink ein. Anschließend genügt ein Klick darauf, um die betreffende Webseite im

Webbrowser anzuzeigen, entweder sofort, falls Sie lokal verfügbar ist, sich beispielsweise auf Ihrer Festplatte befindet, oder aber mit ein wenig Verzögerung, wenn erst eine Verbindung zum Internet aufgebaut werden muss.

25.4.3 Neue Dokumente

Aktivieren Sie im linken Dialogfeldteil *Neues Dokument erstellen*, so können Sie einen Hyperlink zu einem noch gar nicht vorhandenen Dokument erstellen (Abbildung 25.10).

Abbildung 25.10:
Hyperlink zu neuem
Dokument



Entweder geben Sie den kompletten Namen des zu erstellenden Dokuments inklusive Laufwerk und Pfad manuell ein oder aber Sie aktivieren *Ändern...*, um Laufwerk und Pfad wie gewohnt interaktiv im *Dateiauswahl*-Dialogfeld anzugeben und nur noch den Dateinamen selbst einzutippen.

Bei dem Dokument kann es sich um einen beliebigen Typ handeln: TEST.XLS würde eine neue Excel-Mappe anlegen, TEST.DOC ein neues Word-Dokument, TEST.GIF eine neue Grafikdatei etc. – vorausgesetzt, ein zum betreffenden Dokumenttyp passendes Programm ist installiert.

25.4.4 E-Mail-Adressen

Äußerst praktisch ist die Schaltfläche *E-Mail-Adresse* im linken Teil des Dialogfelds (Abbildung 25.11).

Klicken Sie darauf, so können Sie anschließend unter *E-Mail-Adresse* eine E-Mail-Adresse eintippen bzw. im Listefeld darunter eine der zuletzt verwendeten Adressen aussuchen. OK erzeugt einen Hyperlink zur betreffenden E-Mail-Adresse.

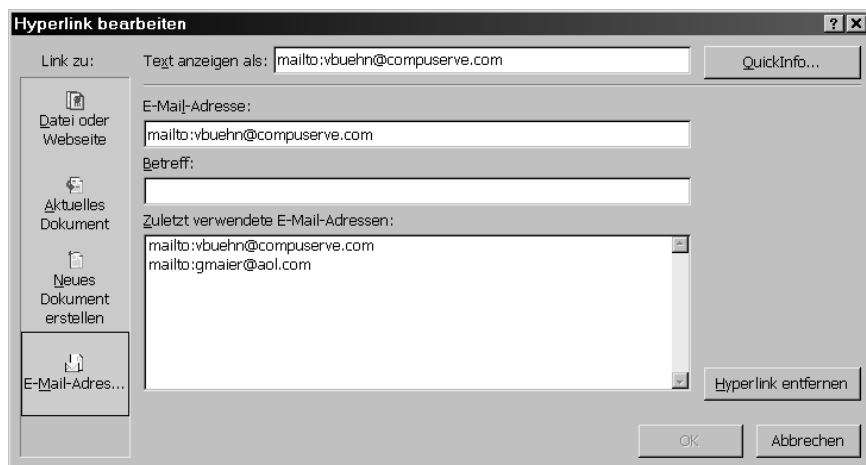


Abbildung 25.11:
Eingabe/Auswahl
einer E-Mail-
Adresse

Auf diese Weise können Sie in Ihre Excel-Tabellen Verweise auf die E-Mail-Adressen von Personen einfügen, die in irgendeiner Beziehung zur betreffenden Tabelle stehen, beispielsweise zu jenem Herrn, der die betreffende Tabelle erstellte oder jemandem, der sie regelmäßig aktualisiert.

Taucht in der Tabelle ein Problem auf, kontaktieren Sie die zuständige Person per E-Mail, indem Sie auf den betreffenden Hyperlink klicken (Abbildung 25.12).

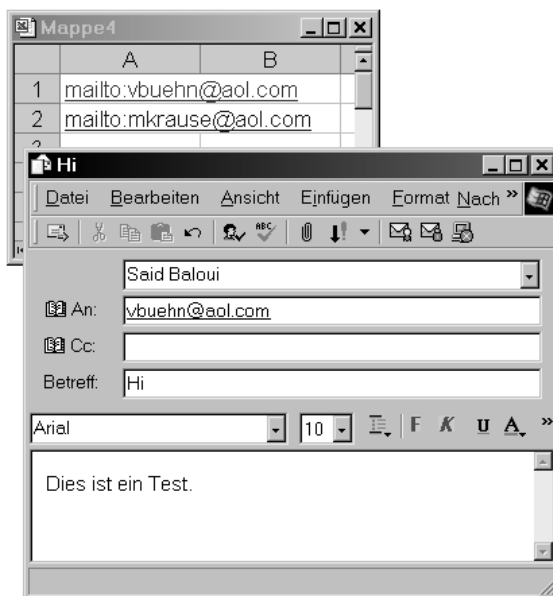


Abbildung 25.12:
E-Mail schreiben

Ihr E-Mail-Programm wird aktiviert und die betreffende Adresse darin vorgegeben. Enthält Ihr E-Mail-Adressbuch bereits einen entsprechenden Eintrag zu dieser Adresse, wird statt der Rohform »vbuehn@compuserve.com« gleich der Name des Adressaten vorgegeben, beispielsweise »Van der Buehn«.

26 Excel-Tabellen im Internet/ Intranet

In Kapitel 3 deutete ich bereits an, dass Sie Excel-Tabellen auch im Internet speichern bzw. dort abgelegte Tabellen öffnen können. Dazu müssen mehrere Voraussetzungen erfüllt sein:

- ➔ Oder Sie haben Zugriff auf einen FTP-Server. Im Gegensatz zu WWW-Servern, auf denen sich Webseiten befinden, werden auf einem FTP-Server Dateien gespeichert, zum Beispiel eben Excel-Arbeitsmappen. Zugriff bedeutet, dass es Ihnen möglich ist, per lokal an Ihrem Rechner angeschlossenen Modem oder über das Firmennetzwerk eine Verbindung zum betreffenden FTP-Server herzustellen.
- ➔ Darüber hinaus müssen auch die Anmeldezugriffsvoraussetzungen erfüllt sein. Der FTP-Server muss Ihnen also den lesenden (Öffnen von Excel-Mappen) bzw. schreibenden (Speichern von Excel-Mappen) Zugriff gestatten. Möglicherweise ist das automatisch der Fall. Bei weniger freizügigen Servern, die nicht einfach für jedermann zugänglich sind, müssen Sie sich zunächst einen Anmeldenamen und ein Anmeldekennwort besorgen.

Entweder haben Sie Zugriff auf einen Server, der Webordner unterstützt. Dann können Sie wie in Kapitel 26.2, »Webordner«, erläutert Ihre Dokumente auf diesem Server speichern und wieder öffnen.



Wollen Sie das Speichern im Internet/Intranet ausprobieren, obwohl Sie momentan keinen Zugriff auf einen derartigen Server besitzen, können Sie mit Hilfe des Kapitels 29, »Webserver einrichten«, selbst einen kleinen Webserver einrichten, der Webordner (der PWS 4.0) bzw. FTP (der PWS 3.0) unterstützt.



26.1 FTP-Server

Sind diese Voraussetzungen erfüllt, benötigen Sie nur noch die Adresse des FTP-Servers im Internet oder Firmennetzwerk, um darauf abgelegte Excel-Tabellen zu öffnen bzw. um dort eigene Excel-Tabellen zu speichern. Im Gegensatz zu Adressen von WWW-Servern, die prinzipiell mit »http://« beginnen, beginnt die Adresse eines FTP-Servers mit »ftp://«.

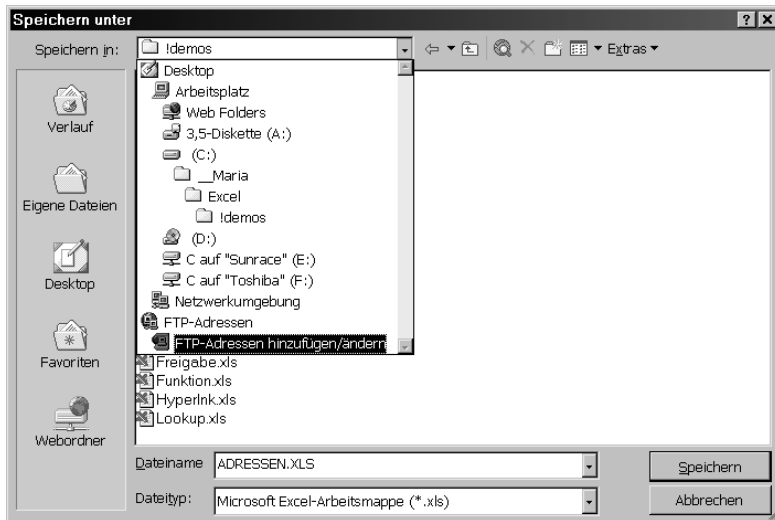
Dabei handelt es sich entweder um einen im Domänen-Namen-System (DNS) registrierten Namen wie *ftp://microsoft.com* oder – beim Zugriff auf einen FTP-Server, der sich im lokalen Firmennetzwerk befindet – um den Namen dieses Rechners im Windows-Netzwerk, beispielsweise *ftp://haupt*.

In jedem Fall handelt es sich dabei um ein Verzeichnis auf der Festplatte des betreffenden Rechners, das Sie genauso behandeln können wie Verzeichnisse auf Ihrer eigenen Festplatte.

*Im Internet
speichern*

Angenommen, Sie wollen auf *ftp://haupt* die Excel-Mappe ADRESSEN.XLS speichern. Dazu wählen Sie DATEI|SPEICHERN UNTER.... und selektieren die betreffende Internetadresse, die sich unterhalb von *FTP-Adressen* befindet. Haben Sie noch keine FTP-Adressen definiert, enthält das Listenfeld darunter nur den Eintrag *FTP-Adressen hinzufügen/ändern* (Abbildung 26.1).

Abbildung 26.1:
FTP-Adressen



Klicken Sie darauf, erscheint ein Dialogfeld, in dem Sie die FTP-Adresse definieren, im Beispiel also »ftp://haupt« eingeben, und Zusatzinformationen wie ein eventuell beim Zugriff darauf anzugebendes Kennwort festlegen (Abbildung 26.2).

Sie können in diesem Dialogfeld gleich beliebige weitere Adressen anderer FTP-Server definieren, indem Sie es nicht mit OK schließen, sondern die angegebene Adresse mit *Hinzufügen* in das Listenfeld einfügen und danach die nächste Adresse definieren.

Um eine auf diese Weise definierte Adresse später zu ändern, klicken Sie erneut auf den Eintrag *FTP-Speicherort hinzufügen/ändern*, um das Dialogfeld wieder zu öffnen, selektieren im Listenfeld *FTP-Sites* die betreffende FTP-Adresse und löschen sie mit *Entfernen* oder editieren sie mit *Ändern*.



Abbildung 26.2:
FTP-Adresse
definieren

Alle auf diese Weise definierten FTP-Adressen werden unterhalb von *FTP-Adressen* angezeigt, sodass Sie diese Adressen zum Speichern von Dateien nun auf die gleiche Weise wie beliebige Ordner auf Ihrer Festplatte benutzen können (Abbildung 26.3).

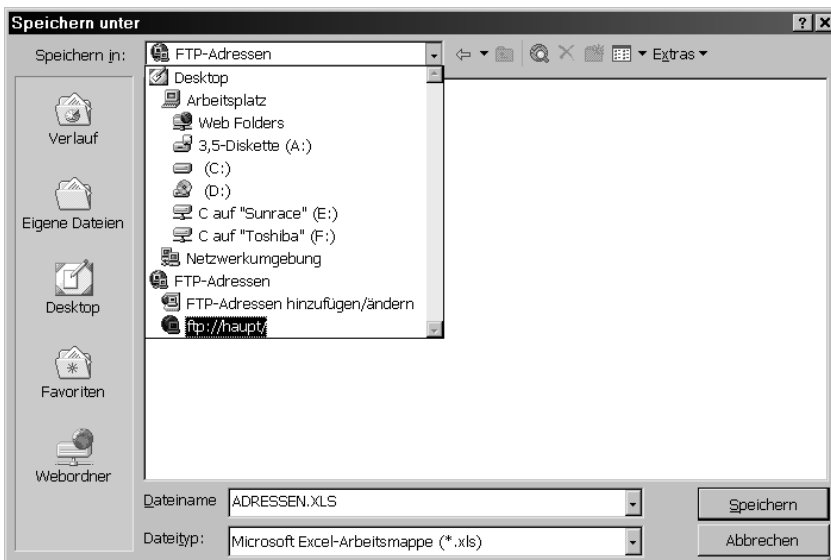


Abbildung 26.3:
Definierte FTP-
Adresse

In der Praxis sieht das so aus: Sie wählen nun als Speicherort Ihrer Excel-Mappe ADRESSEN.XLS den zuvor definierten FTP-Server *ftp://haupt*, dessen Inhalt angezeigt wird.

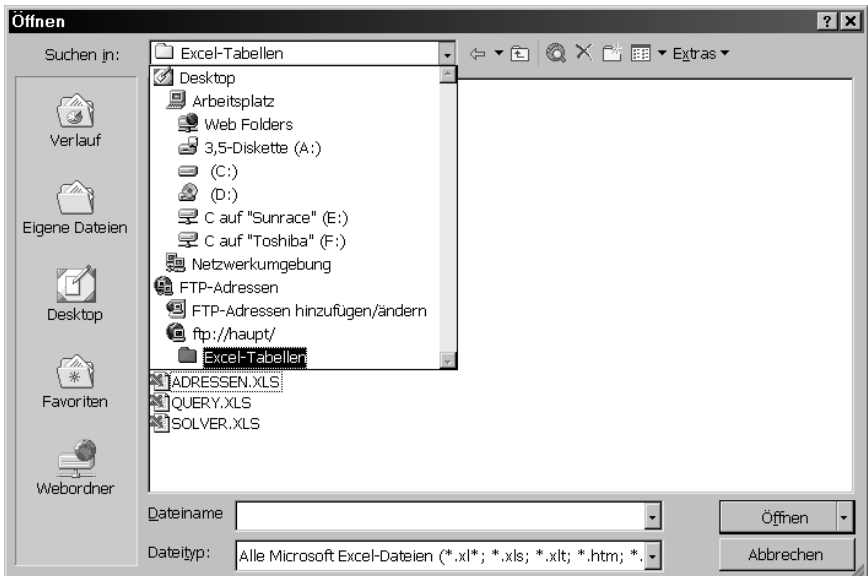
Dabei handelt es sich um ein ganz normales Verzeichnis jenes FTP-Servers, das wie üblich Dateien und auch selbst wieder Unterverzeichnisse enthalten kann, beispielsweise die Unterverzeichnisse ACCESS-DATENBANKEN, EXCEL-TABELLEN und WORD-DOKUMENTE.

Sie wählen sich nun zum Verzeichnis vor, in dem Sie die Excel-Datei speichern wollen, beispielsweise EXCEL-TABELLEN, und klicken wie gewohnt auf *Speichern*, um ADRESSEN.XLS in diesem Verzeichnis des FTP-Servers zu speichern.

Im Internet öffnen

Um diese oder eine andere auf dem FTP-Server gespeicherte Datei zu öffnen, wählen Sie DATEI|ÖFFNEN.... und selektieren die Internetadresse *ftp://haupt*. Im Verzeichnis EXCEL-TABELLEN befindet sich die zuvor dort gespeicherte Datei ADRESSEN.XLS (Abbildung 26.4).

Abbildung 26.4:
Gespeicherte Datei



26.2 Webordner

Webordner setzen voraus, dass die Webserver, auf die Sie zugreifen, dieses spezielle Feature mithilfe der Frontpage Server Extensions auch tatsächlich unterstützen und dass Sie die zum Speichern/Öffnen von Dateien erforderlichen Berechtigungen besitzen. Ist das der Fall, vereinfachen Webordner den Zugriff auf jene Server.

Um einen Webordner vom Windows Explorer aus zu erstellen, öffnen Sie den Ordner *Webordner* und doppelklicken auf den darin enthaltenen Eintrag *Webordner hinzufügen*.

*Webordner
erstellen*

Anschließend meldet sich ein Assistent, mit dessen Hilfe Sie einen neuen Webordner erstellen oder eine Verknüpfung zu einem bereits existierenden Webordner hinzufügen können. Um einen neuen Webordner zu erstellen, wählen Sie die entsprechende Option und müssen im folgenden Schritt die URL-Adresse (URL = Uniform Resource Locator) des Servers angeben und dem neuen Ordner einen Namen geben.

Sie können Webordner jedoch auch direkt in Excel erstellen, indem Sie im Aufgabenbereich auf Webordner hinzufügen... klicken. Danach wählen Sie die Option Neue Webordner erstellen zum Erstellen eines neuen Webordners und geben im folgenden Schritt gleich beide benötigten Informationen ein: unter Ordnerspeicherort die URL-Adresse des Webordners, auf den Sie Zugriff besitzen, und unter Ordner den Namen des neuen Ordners, der darin erstellt werden soll (Abbildung 26.5).

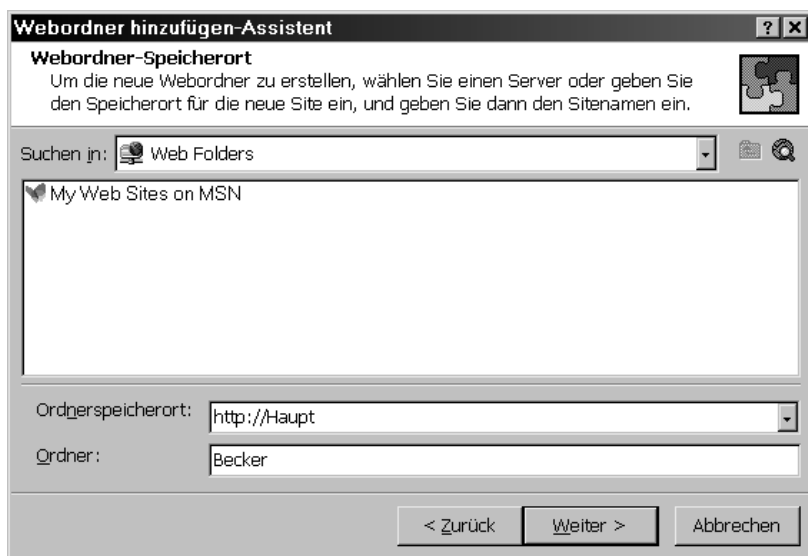


Abbildung 26.5:
Webordner
hinzufügen

Nehmen wir an, der Systemadministrator Ihrer Firma teilt Ihnen mit, dass Sie im firmeneigenen Intranet Zugriff auf einen Webserver namens »Haupt« haben, der Webordner unterstützt und dass die URL-Adresse dieses Web-servers *http://Haupt* lautet (bei einem Server im Internet würde sie stattdessen *http://microsoft.com* oder ähnlich lauten).

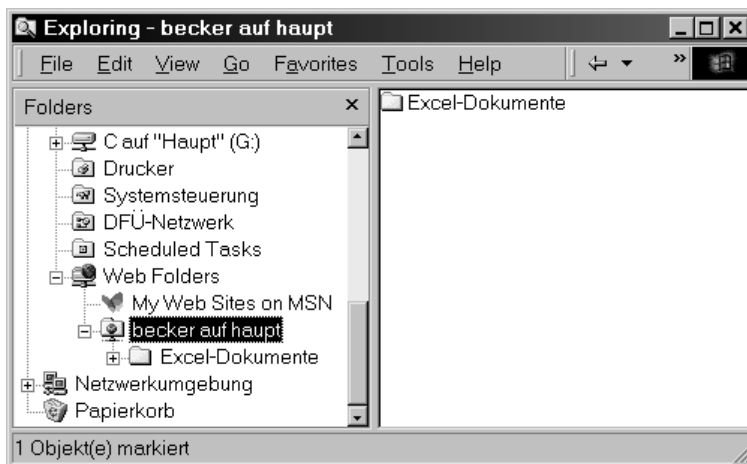


Unter dieser Adresse verbirgt sich ein bestimmtes Verzeichnis dieses Rechners, das so genannte Basisverzeichnis (siehe Kapitel 29, »Webserver einrichten«).

In diesem Verzeichnis wollen Sie nun einen Ordner namens »Becker« erstellen, um darin verschiedene Excel-Dokumente zu speichern. Dazu geben Sie als *Ordnerspeicherort* »http://Haupt« ein und als *Ordner* »Becker«.

Anschließend befindet sich im Explorer im Ordner WEBORDNER außer dem Eintrag *Webordner hinzufügen* zusätzlich auch ein Ordner mit dem von Ihnen eingegebenen Namen, beispielsweise »Becker« oder »images« (Abbildung 26.6).

Abbildung 26.6:
Hinzugefügte
Webordner



Um im Ordner BECKER, IMAGES oder wie auch immer Sie ihn nannten, Unterverzeichnisse wie EXCEL-DOKUMENTE oder WORD-DOKUMENTE zu erstellen, gehen Sie genauso vor, geben jedoch als *Ordnerspeicherort* die URL-Adresse »http://Haupt/Becker« an (das Verzeichnis, in dem die neuen Unterverzeichnisse erstellt werden sollen) und als *Ordner* entsprechend »Excel-Dokumente« bzw. »Word-Dokumente«. Die auf diese Weise erstellten Ordner besitzen die URL *http://Haupt/Becker/Excel-Dokumente* bzw. *http://Haupt/Becker/Word-Dokumente*.

In Excel nutzen

Zur Nutzung eines Webordners in Excel: Um beispielsweise eine Excel-Tabelle im Webordner EXCEL-DOKUMENTE zu speichern, aktivieren Sie im linken Teil des *Dateiauswahl*-Dialogfelds *Webordner*.

Daraufhin wird wie immer der Inhalt dieses Ordners angezeigt, der unter anderem die soeben erstellten Webordner enthält. Sie wählen EXCEL-DOKUMENTE aus, um diesen Ordner zu öffnen und können Ihr Dokument anschließend darin speichern (Abbildung 26.7).

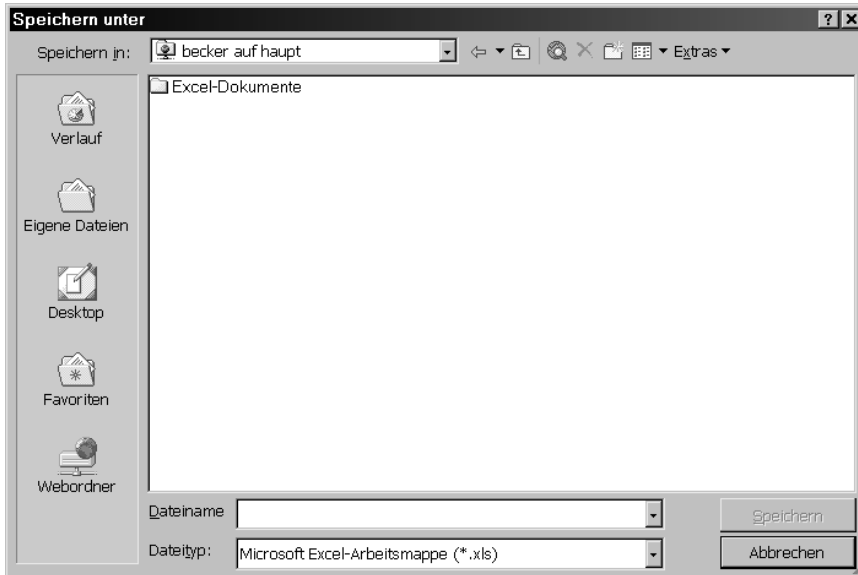


Abbildung 26.7:
In Webordner
speichern

Um es später wieder zu öffnen, wählen Sie DATEI|ÖFFNEN..., aktivieren erneut *Webordner*, öffnen wieder den Webordner BECKER, darin das Unterverzeichnis EXCEL-DOKUMENTE und darin wiederum das zuvor darin gespeicherte Dokument.

Benutzen Sie im Webordner BECKER den Unterordner EXCEL-DOKUMENTE sehr häufig, so können Sie zum schnelleren Zugriff darauf eine Verknüpfung erstellen. Dazu wählen Sie im ersten Schritt des Assistenten die Option zum Erstellen einer Verknüpfung und geben im zweiten Schritt die URL des gewählten Ordners an, im Beispiel also »http://Haupt/Becker/Excel-Dokumente«, und den Namen, den die Verknüpfung zu diesem Ordner erhalten soll, z.B. einfach »Excel-Dokumente«.

*Verknüpfung
erstellen*

Anschließend erscheint diese Verknüpfung im Listenfeld *Webordner* und Anklicken genügt, um diesen Webordner direkt zu öffnen, ohne sich erst in der Baumhierarchie vom übergeordneten Ordner BECKER aus dorthin vor-tasten zu müssen.

All diese Webordner können auch im Windows Explorer fast wie lokale Ordner Ihres Rechners benutzt werden.

*Im Explorer
benutzen*

Fast, da lokale Dateien, die Sie in diese Ordner schieben, prinzipiell kopiert und nicht verschoben werden. Darin enthaltene Dateien können Sie jedoch wie gewohnt öffnen, umbenennen und löschen oder in lokale Ordner auf Ihrer Festplatte kopieren.

Woran Sie jedoch immer denken müssen, egal ob bei derartigen Aktionen im Windows Explorer oder beim Speichern/Öffnen von Dokumenten in Excel: Webordner befinden sich nun einmal nicht auf Ihrer Festplatte, sondern auf einem Webserver. Das bedeutet, dass jede solche Aktion voraussetzt, dass eine Verbindung zu diesem Webserver besteht bzw. nun hergestellt wird.

Selbst wenn diese Verbindung keine langsame Modem-, sondern eine schnelle Netzwerkverbindung im Internet ist: erwarten Sie nicht, dass das Öffnen einer Datei auch nur annähernd so schnell geschieht wie das normalerweise in Ihrem Netzwerk der Fall ist!

Schließlich erfolgen alle Zugriffe auf Dateien in diesem Webordner über die auf dem Webserver installierte Webserver-Software – und die bremst teilweise gewaltig. Vor allem bei Verwendung eines Spielzeug-Webserver wie dem auf der Windows 98-CD enthaltenen Personal Web Server, der sich zwar wunderbar für den gesamten Prozess der Erstellung von Webseiten eignet, nicht aber dafür, diese Webseiten letztlich auch tatsächlich öffentlich verfügbar zu machen.

26.3 Verknüpfungen

Sie können Verknüpfungen mit Excel-Dateien erstellen, die sich im Internet befinden, um in einer Zelle Ihrer Tabelle einen Wert anzeigen zu lassen, der sich in einer Tabelle am anderen Ende der Welt befindet. Dazu geben Sie in der betreffenden Zelle eine Formel folgenden Formats ein:

=Adresse([Dateiname]Blattname!Zelle|Bereich|Name

- ➔ *Adresse* ist die Internetadresse, das heißt der Name eines FTP-Servers wie *ftp://ftp.microsoft.com*, auf den Sie Zugriff haben und auf dem sich die Sie interessierenden Excel-Dateien befinden.
- ➔ *Dateiname* ist der Name der Excel-Mappe (z.B. TEST.XLS)
- ➔ *Blattname* ist der Name des ausgewählten Blatts dieser Mappe (z.B. »Blatt1«)

Zelle|Bereich|Name ist entweder eine Zellreferenz wie C10, ein Bereich wie A1:B5 oder ein Name wie »Test« (siehe Kapitel 9, »Namen«).

Eine nach diesem Schema aufgebaute Adresse wäre beispielsweise *ftp://ftp.microsoft.com/[test.xls]!A1:B5*.



Öffnen Sie eine Mappe mit derartigen Verknüpfungen in das Internet, so fragt Excel Sie, ob die darin enthaltenen Verknüpfungen aktualisiert werden sollen und stellt gegebenenfalls die Verbindung zum FTP-Server im Internet her, um die aktuellen Daten abzufragen und in Ihre Tabelle einzutragen.

Sie können diese Verknüpfungen in einer bereits geöffneten Mappe jederzeit aktualisieren, indem Sie **BEARBEITEN|VERKNÜPFUNGEN...** wählen, die zu aktualisierende(n) Verknüpfung(en) markieren und auf *Jetzt aktualisieren* klicken.

27 Webseiten

27.1 Importieren

Webseiten sind Dateien mit der Endung HTM, manchmal auch HTML, die üblicherweise in Ihrem Webbrowser angezeigt werden, die Sie jedoch auch in Excel importieren und dort weiterbearbeiten können.

- ➔ Um Daten zu importieren, die nicht aktualisiert werden müssen, befördern Sie sie einfach mit dem alten Hausmittel »1.) Interessierende Daten im Webbrowser markieren 2.) BEARBEITEN|KOPIEREN 3.) BEARBEITEN|EINFÜGEN« (bzw. per Drag&Drop) über die Zwischenablage in die Tabelle ein.
- ➔ Um Webseiten zu importieren, die auf Ihrer Festplatte gespeichert sind, öffnen Sie sie einfach in Excel. Nehmen wir als Beispiel eine Webseite, die über die US-Aktien mit der besten Performance der letzten zehn Jahren informiert (Abbildung 27.1).

Um Daten aus Webseiten zu importieren, die regelmäßig aktualisiert werden sollen (Börsenkurse etc.), benötigen Sie Webabfragen (siehe Kapitel 28, »Webabfragen«).

Im Grunde ist eine solche Webseite nichts als eine Textdatei, die Befehle in der Sprache Hypertext Markup Language enthält, kurz HTML. Diese Befehle enthalten nicht nur den dargestellten Text, sondern bestimmen auch das Layout der Seite, zum Beispiel wo genau sich eine Tabelle befindet oder die Schriftart/-größe der verschiedenen Texte.

Was passiert, wenn Sie eine HTML-Datei öffnen, hängt vom betreffenden Programm ab. Öffnen Sie die Datei mit einem Texteditor, wird einfach ihr Rohinhalt angezeigt, also die darin enthaltenen Befehle. Sie können sich diese Befehle anschauen und – Kenntnis der Sprache HTML vorausgesetzt – editieren, also die Webseite selbst verändern (Abbildung 27.2).

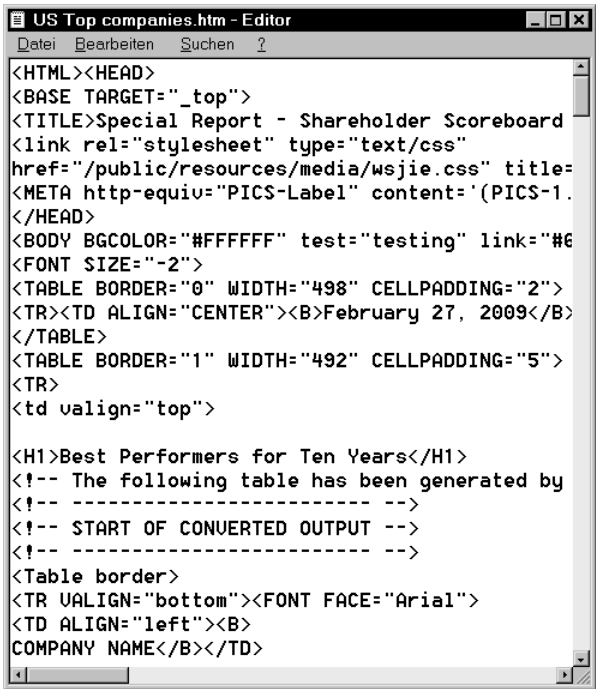
Webseite öffnen

Öffnen Sie die Datei stattdessen mit einer modernen Textverarbeitung wie Word, die die Sprache HTML versteht, so werden diese Befehle von Word wie von Ihrem Webbrowser interpretiert und ausgeführt und als Resultat erscheint statt der Befehle selbst die daraus resultierende Webseite.

Abbildung 27.1:
US_Top.htm



Abbildung 27.2:
HTML-Quelltext



Auch Excel versteht HTML, zeigt jedoch nach dem Öffnen der Datei nicht die Webseite selbst an, sondern importiert deren Inhalt, also alle Texte, Zahlen und Tabellen, die auch bei der Anzeige der Webseite im Webbrowser erscheinen (Abbildung 27.3).

The screenshot shows an Excel window titled 'US_TOP.HTM'. The spreadsheet contains a table with the following data:

	A	B
1	February 27, 2009	
2	Best Performers for Ten Years	
3		
4	COMPANY NAME	RETURN
5	CONCORD EFS INC	60.6%
6	ORACLE CORPORATION	53.5%
7	MICROSOFT CORP	51.0%
8	CLEAR CHANNEL COMM INC	49.4%
9	CONSECO INC	47.3%
10	NIKE INC CL B	46.9%
11	WORLDCOM INC	46.3%
12	TELLABS INC	44.3%
13	INTEL CORP	43.9%
14	HARLEY-DAVIDSON INC	43.3%
15	MICRON TECHNOLOGY INC	41.0%

Abbildung 27.3:
Import in Excel

Die resultierende Excel-Tabelle ist nahezu identisch mit der ursprünglichen Webseite.

Sie können die importierte Webseite nun bearbeiten und als Excel-Tabelle speichern – oder aber erneut als Webseite speichern und wieder im Internet/Intranet veröffentlichen.

27.2 Als Webseite speichern

Excel kann Tabellen auch auf Webseite speichern. Der große Vorteil: Excel selbst wird zur Betrachtung einer solchen Seite nicht benötigt. Jeder Benutzer, der über einen Internetbrowser verfügt, kann sich diese Seite ansehen.

Der Browser interpretiert nach dem Öffnen der Datei die darin enthaltenen Befehle und zeigt die resultierende Webseite an. Enthält die Seite Hyperlinks zu anderen Seiten, kann der Anwender wie gezeigt durch Klicken darauf zu den betreffenden Webseiten springen, die beispielsweise ebenfalls durch den Export von Excel-Tabellen als Webseiten entstanden sind.

Excel-Tabellen als Webseiten zu speichern, kann auch ohne Internetzugang nützlich sein, zum Beispiel im LAN (Local Area Network, im Gegensatz zum WAN, dem Wide Area Network), dem unternehmenseigenen lokalen Netzwerk. Werden darin Internetanwendungen verwendet, so spricht man vom Intranet, was soviel wie unternehmenseigenes Internet bedeutet.

Ihre Kollegen können sich darin die in Webseiten gespeicherten Tabellen auch dann anschauen, wenn sie nicht über Excel verfügen, sondern nur über einen – beliebigen – Webbrowser. Zusätzlich zum firmeneigenen Intranet können Sie diese Seiten natürlich auch im Internet selbst zugänglich machen, beispielsweise auf Ihrer Homepage, und somit Millionen von Anwendern zur Verfügung stellen, egal ob diese über Excel verfügen oder nicht.

Nehmen wir als Beispiel die Excel-Datei ADRESSEN.XLS (Abbildung 27.4):

Abbildung 27.4:
Adressen.xls



	A	B	C	D	E	F	G
1	Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Dat
2	Maier	Stefan	Wannstr.6	80003	München	089/826254	14.0
3	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826125	21.0
4	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.0
5	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.0
6	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	30.1
7	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.1
8	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.0
9	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.0
10	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/836395	18.0
11	Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/726384	01.0
12	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.1
13	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.1

Um einen Eindruck zu bekommen, wie diese Tabelle als Webseite aussieht, wählen Sie DATEI|WEBSEITENVORSCHAU. Excel öffnet die Tabelle daraufhin als Webseite in Ihrem Webbrowser.

Sind Sie mit dem Resultat zufrieden und wollen Sie die Mappe tatsächlich als Webseite speichern, wählen Sie entweder DATEI|SPEICHERN UNTER... und selektieren als Dateityp »Webseite«, »XML-Kalkulationstabelle« oder »Web Archiv«. Oder Sie wählen gleich den Befehl DATEI|ALS WEBSEITE SPEICHERN..., der die gleiche Aufgabe besitzt (Abbildung 27.5).

Sie können einfach *Speichern* wählen (und dabei das vorgegebene Dateiformat »Webseite« unverändert lassen oder alternativ dazu »XML-Kalkulationstabelle« oder »Web Archiv« wählen; mehr dazu in Kürze) oder zuvor noch ein paar Optionen wunschgemäß einstellen:

- ➔ *Titel ändern...* öffnet das abgebildete Dialogfeld, in dem die Überschrift der Webseite definiert wird. Diese Überschrift wird nach dem Öffnen in der Titelleiste des Webbrowsers angezeigt (siehe Bild 27.6).
- ➔ Ist vor dem Öffnen des Dialogfelds ein Bereich wie A1:C5 markiert, so ist die Option *Neu veröffentlichen: Tabelle* durch *Auswahl: \$A\$1:\$C\$5* ersetzt. Aktivieren Sie diese Option, wird entsprechend nur dieser Bereich als Webseite veröffentlicht.

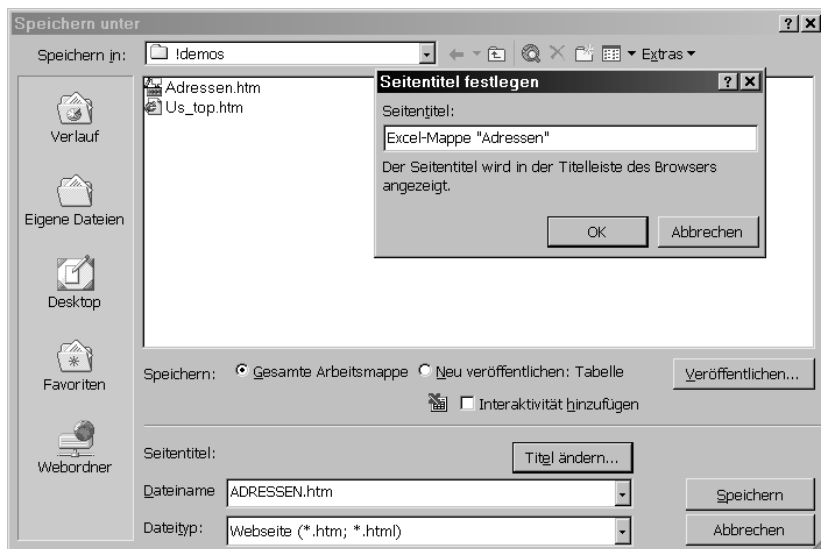


Abbildung 27.5:
Als Webseite speichern

Excel-Mappe "Adressen" - Microsoft Internet Explorer

Adresse C:\Excel\ldemos\ADRESSEN.htm

Name	Vorname	Strasse	Plz	Ort	Telefon	Re-Datum	Fehlbetra
Maier	Stefan	Wannstr.6	80003	München	089/826254	14.01.09	0,00
Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826125	21.01.09	0,00
Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749822	15.02.09	123,56
Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736541	23.08.09	6254,00
Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735322	30.11.09	573,70
Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262529	18.10.09	0,00
Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326738	21.09.09	875,64
Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/83626	03.02.09	273,90
Bühn	Walter	Hans-Pinsel Str.2	80042	München	089/836395	18.02.09	0,00
Irlbeck	Gustav	Maierbacherweg 21	80017	München	089/726384	01.02.09	0,00
Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/93651	10.12.09	3864,00
Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/83643	17.11.09	1925,00

Tabelle1 | Tabelle2 | Tabelle3 | Tabelle4 | Tabelle5 | Tabelle6 | Tabelle7 | Tabelle8 | Tabelle9

Arbeitsplatz

Abbildung 27.6:
Adressen.htm

Im HTML-Format
speichern

Veröffentlichen Sie wie vorgegeben die gesamte Mappe im vorgegebenen Format »Webseite«, so wird im angegebenen Verzeichnis außer der Datei ADRESSEN.HTM ein Unterverzeichnis ADRESSEN-DATEIEN erzeugt, das unter anderem für jede Tabelle der Mappe eine eigene HTML-Datei enthält.

Um die resultierende Webseite im Webbrowser zu öffnen, klicken Sie auf ADRESSEN.HTM (Abbildung 27.6).



Sie können stattdessen auch auf eine der HTML-Dateien im Verzeichnis ADRESSEN-DATEIEN klicken. Da jede dieser Dateien Verweise auf das im Verzeichnis darüber gespeicherte Hauptdokument ADRESSEN.HTM enthält, ist das Resultat das gleiche, als hätten Sie auf ADRESSEN.HTM selbst geklickt.

Das Resultat ist insofern recht beeindruckend, als tatsächlich nicht nur die gerade aktive Tabelle als Webseite gespeichert wurde, sondern jede Tabelle der Mappe. Wie in Excel selbst können Sie im unteren Register beliebige Blattnamen anklicken, um sich die betreffende Tabelle anzuschauen.

Noch beeindruckender ist es allerdings, ADRESSEN.HTM in Excel selbst zu öffnen (Abbildung 27.7).

Abbildung 27.7:
Nach dem Öffnen
in Excel

Der Name »Adressen.htm« in der Titelleiste beweist, dass es sich tatsächlich um die Webseite handelt. Dennoch ist nicht der geringste Unterschied zur Original-Excel-Tabelle vor dem Speichern als Webseite zu sehen.

Das betrifft nicht nur diese eine Tabelle, sondern die gesamte Mappe: durch das Speichern einer Excel-Mappe als Webseite geht nicht die geringste Information verloren. Alles bleibt erhalten, die in der Mappe enthaltenen Formeln, Formatierungen, eventuelle Gliederungen und so weiter.

Fazit: Wenn Sie wollen, brauchen Sie Excel-Mappen überhaupt nicht mehr im Format »Microsoft Excel-Arbeitsmappe« zu speichern, sondern können Sie ausschließlich als Webseiten speichern!

*Excel-Dokumente
immer als Web-
seite speichern*

Allerdings sollten Sie sich gut überlegen, ob Sie das vorgegebene Dateiformat »Webseite« unverändert lassen oder alternativ dazu »XML-Kalkulationstabelle« oder »Web Archiv« wählen. Die Nachteile des Formats »Webseite« sind:

- ➔ Statt einer einzigen Datei wie ADRESSEN.XLS wimmelt es auf Ihrer Festplatte in Kürze nur so von Dateien, außer, Ihre Mappen enthalten immer nur eine einzige Tabelle mit Daten und keinerlei Bildern, Clip-Arts etc.

? Ansonsten wird nämlich nicht nur eine HTML-Datei gespeichert, sondern im gleichen Verzeichnis ein zusätzlicher Ordner angelegt, der alle möglichen Hilfsdateien enthält: Formatvorlagendateien, pro Bild der Mappe eine eigene Datei und pro Blatt der Mappe jeweils eine eigene HTML-Datei.
- ➔ Diese Unzahl an Dateien benötigen ein Mehrfaches des Platzes, den die äquivalente Microsoft Excel-Arbeitsmappe belegt und verzögern das Speichern bzw. Öffnen gegenüber der äquivalenten Microsoft Excel-Arbeitsmappe.
- ➔ Jede der im Verzeichnis ADRESSEN-DATEIEN enthaltenen Dateien ist ebenso tabu wie ADRESSEN.HTM selbst! Löschen Sie eine dieser Dateien, können Sie auch gleich den Rest löschen, da er nun nutzlos ist.

Ich würde Ihnen daher raten, dieses Dateiformat nur für Excel-Mappen zu verwenden, die Sie tatsächlich veröffentlichen wollen und auf die auch Personen zugreifen können sollen, die nicht über Excel verfügen.

Das ist auch nur in dem Sonderfall sinnvoll, wenn jene Personen über Excel verfügen: wenn Sie zur Veröffentlichung Ihrer Excel-Tabellen im Internet einen Schmalspur-Webserver einsetzen, der weder FTP-Zugriff noch die Verwendung von Webordner erlaubt. Dann können interessierte Personen zwar auf Webseiten zugreifen, die sich auf Ihrem Server befinden, aber eben nicht auf dort gespeicherte XLS-Dateien, also auf Excel-Mappen.

Eine überlegenswerte Alternative besteht darin, als Dateiformat »XML-Kalkulationstabelle« oder »Web-Archiv« zu wählen.

In beiden Fällen wird die Mappe ebenfalls als Webseite gespeichert, allerdings wird nur *eine einzige* Datei erzeugt, in der sich auch alle Hilfsdateien (Grafikdateien etc.) befinden, sodass diese Art der Speicherung wesentlich kompakter ist.



*Als Web-Archiv
speichern*

Das Format »Web-Archiv« (Endung MHT) setzt zur Betrachtung der Webseite den Internet Explorer ab Version 4.0 voraus. Ist diese Bedingung bei allen Nutzern Ihrer Webseite erfüllt, spricht nichts gegen die Verwendung dieses Formats.

*Als XML-Kalkulations-Tabelle
speichern*

Abgesehen davon, steht auch noch das XML-Format zur Verfügung.

Beim XML-Format handelt es sich um einen allgemeinen Standard wie den HTML-Standard, wobei jedoch im Gegensatz zum HTML-Format weniger der optische Aufbau der Webseite beschrieben wird, sondern vielmehr die Strukturierung der darzustellenden Daten.

Um den optischen Aufbau zu definieren, kann zusätzlich zur erzeugten XML-Datei ein »XSL Stylesheet« verwendet werden, eine zusätzliche Datei mit der Endung XSL, praktisch eine Art Vorlage, die die optische Darstellung der in der XML-Datei beschriebenen Daten festlegt.

Da übrigens beides zusammen, XSL-Datei und XML-Datei, sowohl die darzustellenden Daten als auch die Art der Darstellung beschreiben, kann beides zusammen verwendet werden, um diese Informationen im älteren, aber weiter verbreiteten HTML-Format darzustellen, also als einfache Webseite, die jeder Browser versteht, sodass jederzeit »Abwärtskompatibilität« hergestellt werden kann. Das Speichern im modernen XML-Format ist somit nicht zwangsläufig eine Einbahnstraße.

Welches Format sollten Sie verwenden? Nun, XML ist wie HTML ein echter allgemeiner Standard, wesentlich flexibler und zukunftssträchtiger als Web-Archiv und – wenn es sein muss – können die dadurch beschriebenen Daten wie gesagt auch ins HTML-Format umgewandelt werden.

Um Excel-Mappen als Webseiten zu speichern, sollten Sie daher entweder den einfachsten Standard verwenden, der die geringsten Anforderungen auf Betrachterseite stellt, also HTML – oder gleich den jüngeren Standard XML, der jedoch geeignete moderne Webbrowser beim Betrachter voraussetzt.

27.3 Interaktive Webseiten

**!!
STOP**

Sie können auch Webseiten mit Interaktivität erzeugen. Dann kann der Anwender die Seite in seinem Webbrowser nicht nur betrachten, sondern auch verändern – vorausgesetzt, er verfügt über den Internet Explorer ab Version 4.0.

Die für die Interaktivität verwendeten Microsoft Office Web Components bieten alle Grundfunktionen von Excel: der Anwender kann im Webbrowser Zelleninhalte verändern, Summenformeln bilden, Pivot-Tabellen drehen und so weiter.

All das eröffnet die Schaltfläche *Veröffentlichen...* des zuvor gezeigten Dialogfelds (Abbildung 27.8).

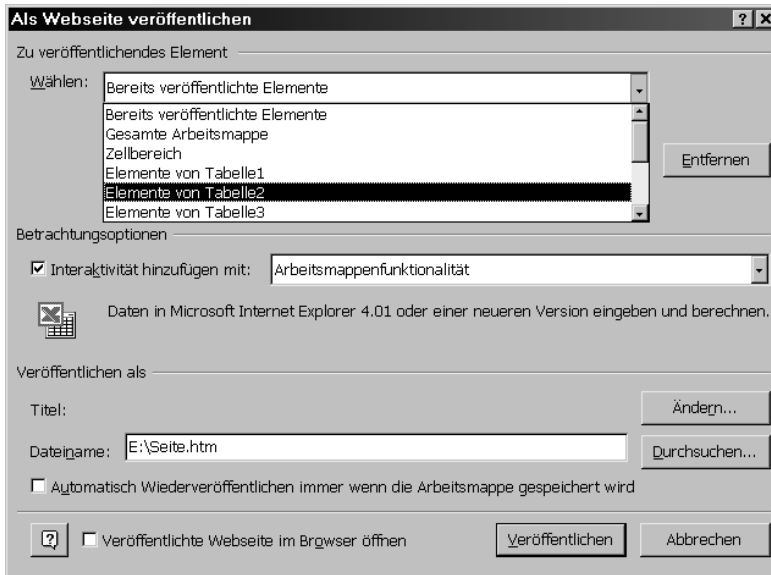


Abbildung 27.8:
Veröffentlichungs-
optionen

Zunächst einmal können Sie unter *Wählen* genauer spezifizieren, was als Webseite veröffentlicht werden soll:

- ➔ die zuvor veröffentlichten Elemente der Mappe (im Beispiel die gesamte Mappe). Excel merkt sich, was zuvor veröffentlicht wurde und zeigt diese Elemente bei Wahl dieser Option im Listenfeld an. Wollen Sie nicht erneut all diese Elemente veröffentlichen, können Sie einzelne selektieren und mit *Entfernen* aus dem Listenfeld löschen, um nur die übrig bleibenden Elemente zu veröffentlichen;
- ➔ ein bestimmter Zellenbereich (Auswahl dieser Option aktiviert darunter das bekannte Bereichs-Selektionsfeld, in dem Sie eine Bereichsangabe eintippen oder mit der Maus auf einen Bereich zeigen können)
- ➔ oder die Elemente einer bestimmten Tabelle der Mappe. Wählen Sie diese Option, erscheinen im Listenfeld darunter alle Elemente der betreffenden Tabelle: die Tabelle selbst mit all Ihren Daten, gegebenenfalls in dieser Tabelle enthaltene Pivot-Tabellen und eventuell darin enthaltene Diagramme werden alle einzeln aufgelistet. Sie können nun das zu veröffentlichende Element der Tabelle auswählen.

Die Schaltfläche *Ändern...* ermöglicht es, die Überschrift der erzeugten Webseite zu ändern. *Dateiname* verwenden Sie zur Eingabe des Namens, unter dem die Webseite gespeichert werden soll, und *Durchsuchen...* öffnet das Dateiauswahl-Dialogfeld zur Festlegung des Speicherorts.

Automatisch Wiederveröffentlichen... publiziert die Mappe jedesmal, wenn Sie sie speichern, erneut mit den aktuellen Einstellungen als Webseite, sodass jene immer aktuell ist, auch nach – gespeicherten – Änderungen der Excel-Mappe.

Aktivieren Sie *Veröffentlichte Webseite im Browser öffnen*, wird nach Erzeugung der Webseite automatisch Ihr Webbrowser aktiviert und die Webseite darin geöffnet.

Weitaus interessanter ist jedoch *Interaktivität hinzufügen mit*. Ist diese Option aktiviert, dann haben Sie im zugehörigen Listenfeld prinzipiell die Wahl zwischen *Arbeitsmappenfunktionalität*, *PivotTable-Funktionalität* oder *Diagrammfunktionalität*.

Welche dieser Optionen verfügbar sind, hängt davon ab, welches Objekt Sie unter *Wählen* bzw. im Listenfeld darunter zur Veröffentlichung auswählen:

- ➔ Veröffentlichen Sie eine Tabelle (normale oder Pivot-Tabelle), so können Sie zwischen *Arbeitsmappenfunktionalität* und *Pivot-Tabellenfunktionalität* wählen. Die erzeugte Webseite verhält sich dann wie eine normale Excel-Tabelle bzw. wie eine Pivot-Tabelle und kann genauso benutzt werden.
- ➔ Veröffentlichen Sie ein Diagrammobjekt, ist nur die Option *Diagrammfunktionalität* verfügbar. Dann verhält sich die Webseite wie ein Excel-Diagramm und kann ebenso manipuliert werden.

Nehmen wir als Beispiel die Veröffentlichung einer Tabelle mit der wohl am häufigsten verwendeten Option *Arbeitsmappenfunktionalität*, die folgende Webseite erzeugt (Abbildung 27.9)

Sollten Sie glauben, die Ähnlichkeit mit einer Excel-Tabelle sei nur optischer Natur, sind Sie gewaltig im Irrtum! Die hinter dem ganzen Zauber steckenden Microsoft Office Web Components verleihen dieser Webseite echte Excel-Funktionalität. Sie können

- ➔ Zellen per Anklicken selektieren,
- ➔ Zellinhalte eingeben, ändern und löschen,
- ➔ Formeln benutzen
- ➔ und wie ein Blick auf die Symbolleiste zeigt, ist sogar das Summieren von Werten auf Knopfdruck möglich, auf-/absteigende Sortierung der Tabelle nach dem Inhalt der aktuellen Spalte, das AutoFiltern etc.

Und das ist bei weitem noch nicht alles! Das zweite Symbol von rechts öffnet ein Dialogfeld, dessen Register Ihnen vertraut vorkommen dürften (Abbildung 27.10).

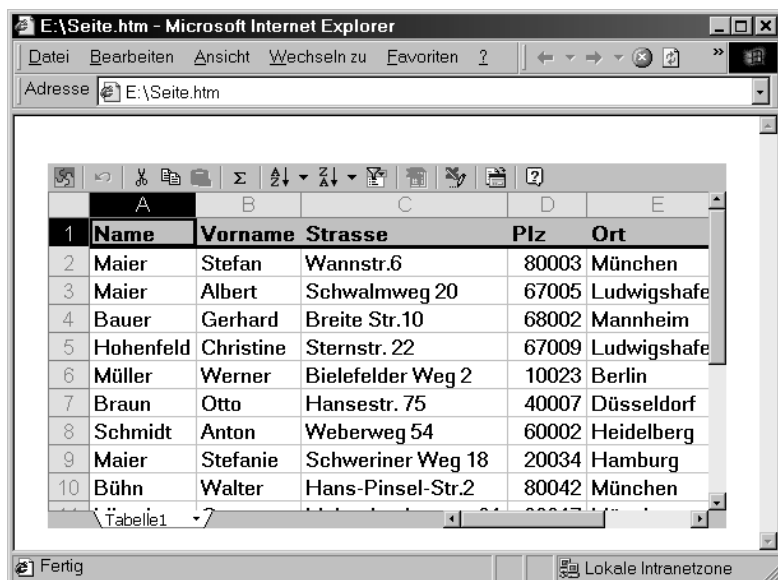


Abbildung 27.9:
Arbeitsmappen-
funktionalität

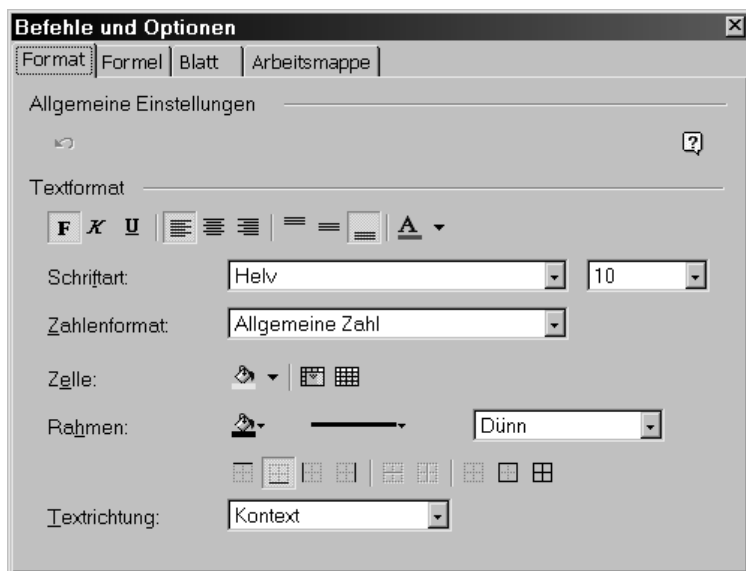


Abbildung 27.10:
Eigenschaftens-
toolbox

Weitere Worte erübrigen sich, da Sie all diese Befehle ja von Excel her selbst kennen. Nur ein kleines Beispiel für ihre Anwendung (Abbildung 27.11):

Für diese Abbildung blendete ich Gitternetzlinien, Spalten- und Zeilenköpfe ein, summierte die Werte in Spalte H, formatierte die Ergebniszelle kursiv und unterstrichen und blendete die Symbolleiste aus.

Abbildung 27.11:
Anwendung

	D	E	F	G	H
6	10023	Berlin	0372/735322	30.11.09	573,70
7	40007	Düsseldorf	0125/262529	18.10.09	0,00
8	60002	Heidelberg	0662/326738	21.09.09	875,64
9	20034	Hamburg	0326/83626	03.02.09	273,90
10	80042	München	089/836395	18.02.09	0,00
11	80017	München	089/726384	01.02.09	0,00
12	10065	Berlin	0372/93651	10.12.09	3864,00
13	68003	Mannheim	0621/83643	17.11.09	1925,00
14					13889,80
15					

Tabelle1

Fertig Lokale Intranetzone

Wie gesagt, analog dazu können Sie Tabellen mit *PivotTable-Funktionalität* veröffentlichen und die resultierende Pivot-Tabelle anschließend direkt im Webbrowser wie eine Excel Pivot-Tabelle benutzen (drehen etc.). Dazu

- ➔ wählen Sie unter *Wählen* (siehe Bild 27.8) zunächst die Tabelle Ihrer Mappe aus, die den interessierenden Datenbereich enthält,
- ➔ selektieren Sie im Listenfeld darunter als zu veröffentlichendes Element jene Tabelle bzw. eine eventuell darin enthaltene Pivot-Tabelle, die veröffentlicht werden soll
- ➔ und selektieren Sie als Interaktivitäts-Option diesmal *PivotTable-Funktionalität* (Abbildung 27.12).

Enthält die zu veröffentlichende Mappe ein Diagramm, so können Sie dieses mit *Diagrammfunktionalität* veröffentlichen, indem Sie

- ➔ unter *Wählen* die Tabelle mit dem Diagramm auswählen
- ➔ und im Listenfeld darunter als zu veröffentlichendes Element jener Tabelle das gewünschte Diagramm selektieren.
- ➔ Dadurch wird die – bei Diagrammen einzig mögliche – Interaktivitäts-Option *Diagrammfunktionalität* automatisch vorgegeben (Abbildung 27.13).

Abbildung 27.12:
Pivot-Tabellen-
funktionalität

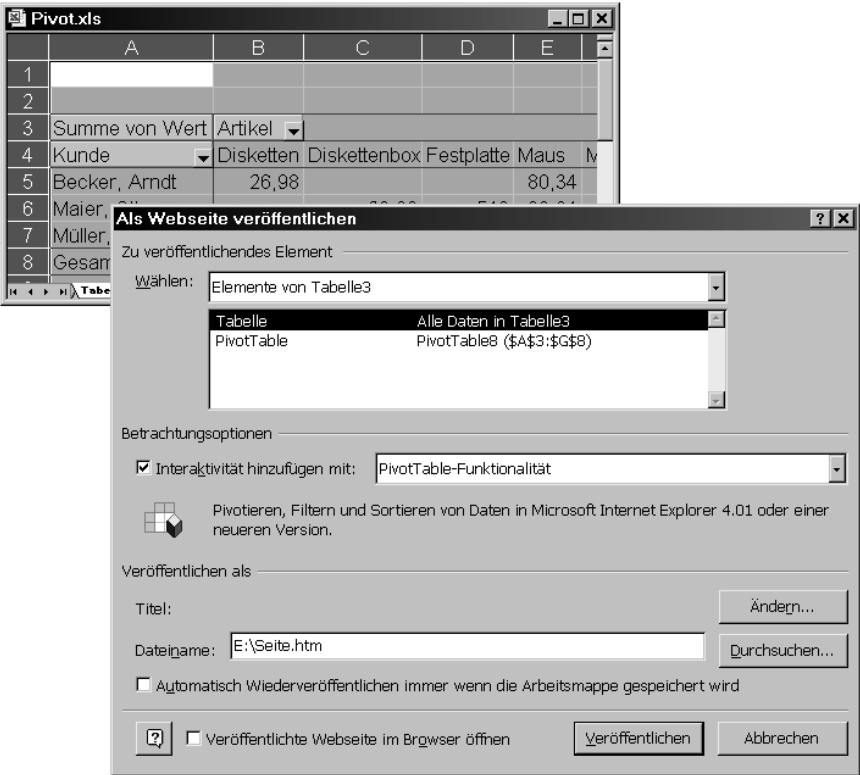
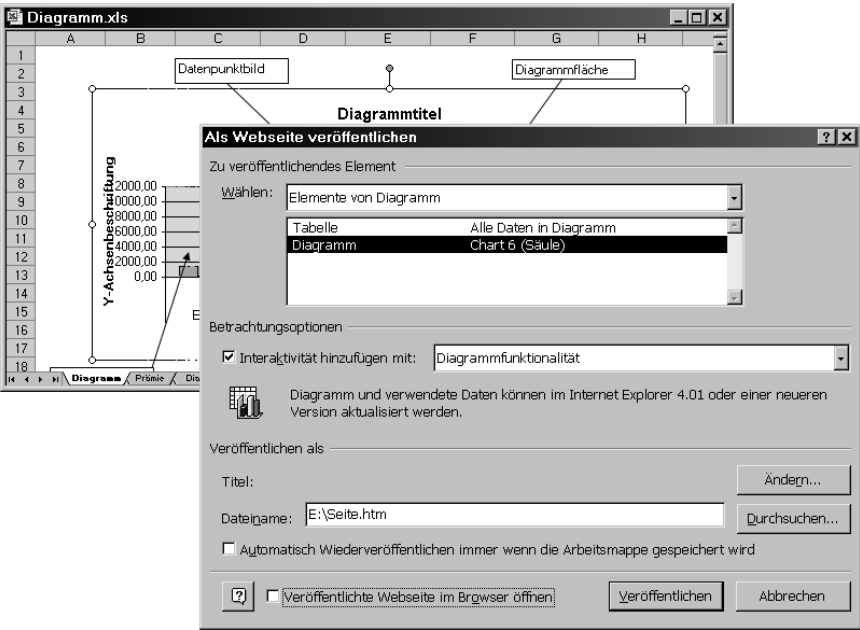


Abbildung 27.13:
Diagramm-
funktionalität



28 Webabfragen

Webabfragen importieren Daten aus einer Webseite in eine Excel-Tabelle und ermöglichen Ihnen, diese Daten jederzeit auf Knopfdruck zu aktualisieren.

28.1 Ausführen

Dazu wählen Sie DATEN|EXTERNE DATEN IMPORTIEREN|DATEN IMPORTIEREN..., um das bereits in Kapitel 14.2, »Externe Daten in Excel importieren«, erläuterte Dialogfeld zu öffnen, in dem unter anderem vorgefertigte Webabfragen (Endung IQY) aufgelistet sind. (Abbildung 28.1).

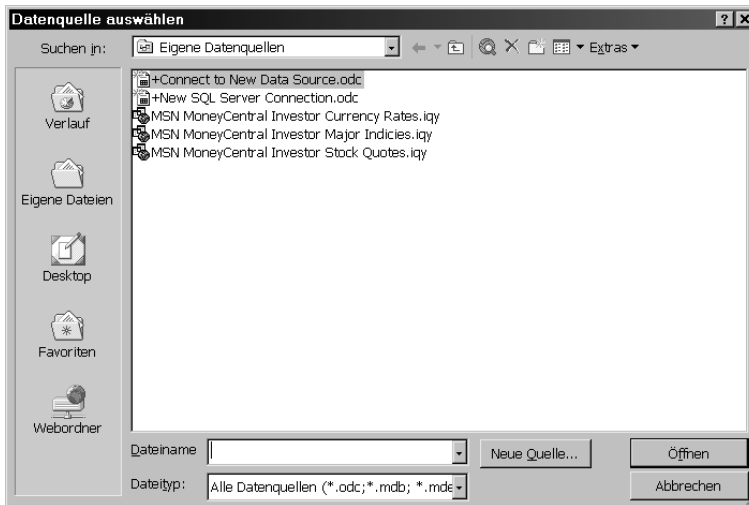


Abbildung 28.1:
Webabfragen
ausführen

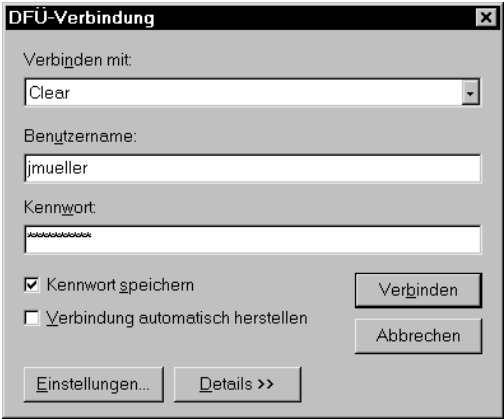
Öffnen Sie eine davon, beispielsweise !!!MSN MONEYCENTRAL INVESTOR MAJOR INDICIES.IQY, so erscheint das ebenfalls im Kapitel über Query erläuterte Dialogfeld, in dem Sie angeben, wohin die betreffenden Daten importiert werden sollen (Abbildung 28.2).

OK stellt die Verbindung zur abzufragenden Webseite her und startet dazu Ihren Webbrowser bzw. blendet zuvor gegebenenfalls das *Verbindungs-*Dialogfeld ein (Abbildung 28.3).

Abbildung 28.2:
Importbereich
festlegen



Abbildung 28.3:
Verbindung
herstellen



Im Beispiel wird eine Verbindung zu einer Webseite von »MSN« hergestellt, dem Microsoft Network, die Abfragen aller möglichen Kurse ermöglicht: Währungen, Indizes, Aktien etc.

Die ausgewählte Abfrage ruft die Kurse der wichtigsten Indizes ab und speichert sie im angegebenen Zielbereich (Abbildung 28.4).

Abbildung 28.4:
Importierte Tabelle

	A	B	C	D	E	F
1	Stock Quotes Provided by MSN MoneyCentral I					
2	Click here to visit MSN MoneyCentral Investor					
3				Last	Previous Close	High
4	Dow Jones Industrials Index	Chart	News	9785,4	9785,4	0
5	Dow Jones Composite Index	Chart	News	3026,4	3026,4	0
6	Dow Jones Transportation Index	Chart	News	2765,1	2765,1	0
7	Dow Jones Utilities Index	Chart	News	372,84	372,84	0
8	DAX (PERFORMANCEINDEX)	Chart	News	5792,5	5817,5	5842,6
9	ETSE 100	Chart	News	5550,7	5614	5614

Einige der mit Office ausgelieferten Abfragen sind so genannte Parameter-abfragen, beispielsweise MSN MONEYCENTRAL INVESTOR STOCK QUOTES. IQY. Beim Ausführen von Parameterabfragen wie dieser erscheint nach der Angabe des Importbereichs ein weiteres Dialogfeld (Abbildung 28.5).

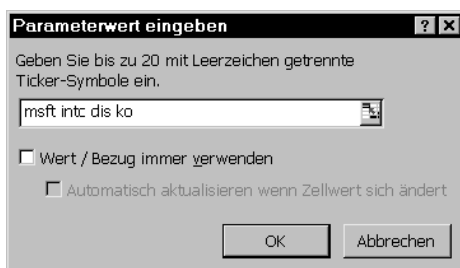


Abbildung 28.5:
Parameter

Es fordert Sie auf, die Werte für die Parameter anzugeben, die zur Ausführung der Webabfrage notwendig sind. Im Beispiel sind das die durch Leerzeichen getrennten Tickersymbole der Aktien, die Sie interessieren.

Die hier gewählten Symbole »msft«, »intc«, »dis« und »ko« stehen für Microsoft, Intel, Coca Cola und Walt Disney und die Abfrage importiert entsprechend diese Aktienkurse (Abbildung 28.6).

				Last	Previous Close	High	Low	Vol
4	Microsoft Corporation	Chart	News	55,6	55,56	0	0	79
5	Intel Corporation	Chart	News	26,9	26,94	0	0	249
6	Walt Disney Company	Chart	News	28,4	28,36	0	0	
7	Coca-Cola Company	Chart	News	46	45,95	0	0	

Abbildung 28.6:
Importierte
Aktienkurse

Bei jeder Aktualisierung einer Parameterabfrage werden die zu verwendenden Parameter im Dialogfeld Parameterwert eingeben erneut abgefragt. Möglicherweise wollen Sie jedoch immer wieder die gleichen Werte verwenden, beispielsweise die Tickersymbole »msft«, »intc«, »dis« und »ko«.



Dann sollten Sie nach der Eingabe dieser Werte das Kontrollkästchen Wert/Bezug immer verwenden dieses Dialogfelds aktivieren. Excel merkt sich nun die angegebenen Parameterwerte und Sie ersparen sich die erneute Eingabe bei folgenden Aktualisierungen.



Statt eines festen Werts wie »msft« oder »ko« können Sie eine Zellreferenz in diesem Dialogfeld angeben, die diese Zeichenkette enthält. Aktivieren Sie außer Wert/Bezug immer verwenden auch das Kontrollkästchen Automatisch aktualisieren wenn Zellwert sich ändert, so wird die Abfrage erneut ausgeführt, sobald sich der Wert der betreffenden Zelle ändert, beispielsweise weil darin inzwischen der Text »intc« eingegeben wurde.

28.2 Erstellen und Editieren

Trotz der kryptischen Erweiterung IQY (für »Internet Query«) sind Webabfragen schlichte Textdateien, die mit Notepad oder jedem anderen Editor erstellt und bearbeitet werden können. Der Inhalt ist meist recht harmlos und besteht im einfachsten Fall aus einer einzigen Zeile wie

`http://Name des Servers/Name der Webseite.`

Name des Servers ist der Name des Webserver, auf dem sich die Webseite befindet, auf die die Abfrage zugreifen soll, und *Name der Webseite* der Name der darauf gespeicherten Webseite.

Diese Zeile importiert alle HTML-Tabellen (in Webseiten eingebundene Tabellen), die die betreffende Webseite enthält.

Mit einem Editor

Befindet sie sich auf einem Webserver namens »MyCompany«, der sich im Internet befindet und mit der Domänen-Endung »com« registriert ist, erstellen Sie eine Textdatei, die folgende Zeile enthält:

`http://www.MyCompany.com/MeinHTML.htm`

Bei einem Webserver, der sich stattdessen im firmeneigenen Intranet befindet und den Namen »haupt« besitzt, lautet diese Zeile entsprechend

`http://haupt/MeinHTML.htm.`

Die Textdatei speichern Sie unter einem Namen wie WEBQUERY.IQY. Führen Sie diese Webabfrage aus, so importiert Excel alle in der Webseite enthaltenen Tabellen.



Häufig genügt die Angabe des Servernamens ohne zusätzliche Angabe eines Dateinamens, um die Startseite des betreffenden Servers zu importieren. Webserver definieren nämlich meist eine Default-Webseite, auf die automatisch zugegriffen wird, wenn keine spezifische Webseite angegeben wird.

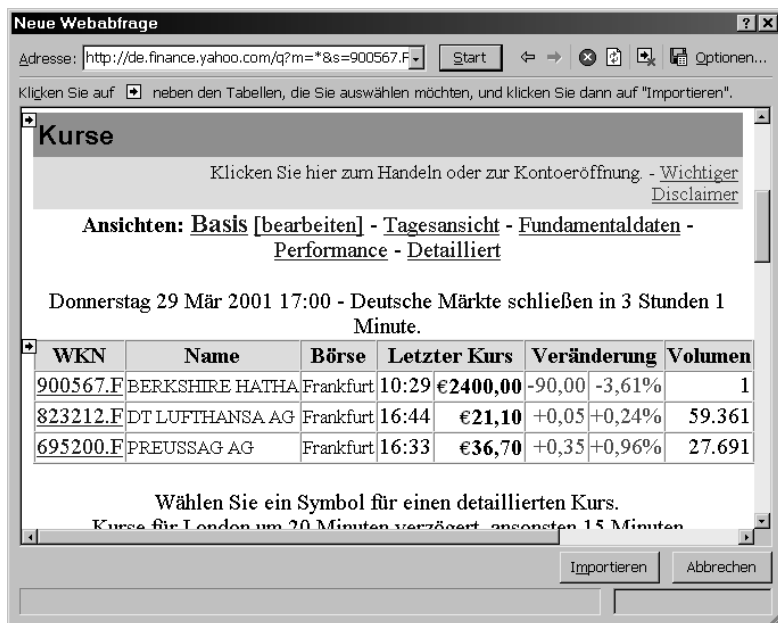
Viele Webseiten enthalten nicht nur Tabellen, sondern zusätzlich alle möglichen Texte, Bilder etc. Um dennoch gezielt nur darin enthaltene Tabellen zu importieren, erstellen Sie die IQY-Textdatei nicht selbst, sondern überlassen das Excel.

Dazu wählen Sie DATEN|EXTERNE DATEN IMPORTIEREN|NEUE WEBABFRAGE... (Abbildung 28.7).

Mit Excel



Abbildung 28.7:
Neue Webabfrage



Im oberen Eingabefeld geben Sie die Adresse der Webseite an bzw. wählen eine der zuletzt verwendeten Adressen oder steuern die Webseitenanzeige mit den Browser-Symbolen.

Excel zeigt entsprechend der Abbildung einen verschiebbaren Ausschnitt der betreffenden Webseite an und hebt mit kleinen Pfeilen am linken Rand die einzelnen Tabellen oder sagen wir besser »Abschnitte« – die manchmal nur schwer als Tabelle zu interpretieren sind – hervor, aus denen die Seite besteht.

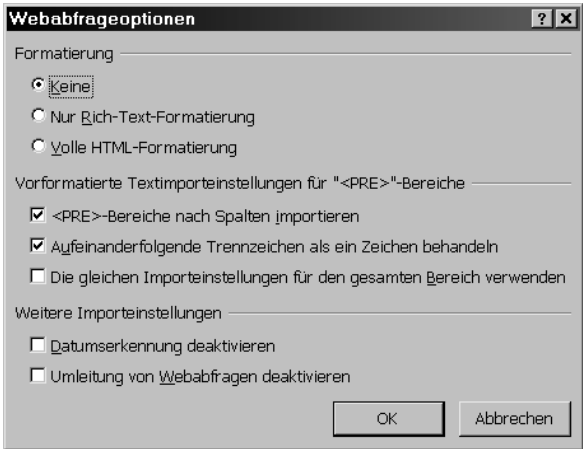
Durch Anklicken der betreffenden Pfeile selektieren Sie die zu importierenden Abschnitte, im Beispiel den untersten sichtbaren Abschnitt, die Tabelle mit den Aktienkursen.

Der Ausschnitt kann als Webbrowser benutzt werden. Wollen Sie doch eine Tabelle aus einer anderen Webseite importieren, tippen Sie im Adressfeld einfach deren Adresse ein und klicken auf Start bzw. benutzen zum Blättern die Web-Navigations-Symbole.



Importieren importiert die ausgewählten Abschnitte, wobei Sie zuvor noch mit *Optionen...* ein Dialogfeld mit verschiedenen Einstellungsmöglichkeiten öffnen können (Abbildung 28.8).

Abbildung 28.8:
Datenbereichs-
formatierung



Die wichtigste ist die Wahl der Formatierung. Mit der Voreinstellung *Keine* wird reiner Text bzw. Zahlen importiert. Wählen Sie dagegen *Volle HTML-Formatierung*, wird die Tabelle wie im Webbrowser angezeigt importiert (Abbildung 28.9).

Abbildung 28.9:
Formatierungs-
übernahme

The screenshot shows an Excel spreadsheet with the following data:

	A	B	C	D	E	F	G	H	I
	WKN	Name	Börse	Letzter Kurs	Veränderung	Volumen			Mehr Infos
2	900567.F	BERKSHIRE HATHA	Frankfurt	10:29	2400	-90	-3,61%	1	Chart
3	823212.F	DT LUFTHANSA AG	Frankfurt	16:47	21,1	0,05	0,24%	59.481	Chart, News, Profil, Analyse
4	695200.F	PREUSSAG AG	Frankfurt	16:53	36,8	0,4	1,10%	28.331	Chart, News, Profil, Analyse
5	766400.F	VOLKSWAGEN AG	Frankfurt	16:48	52,4	-0,6	-1,13%	35.627	Chart, News, Profil, Analyse
6	542500.F	DEGUSSA AG	Frankfurt	16:41	33,5	-0,1	-0,30%	12.860	Chart, News, Analyse
7	710000.F	DAIMLERCHRYSLER	Frankfurt	16:53	50,5	0,6	1,20%	84.984	Chart, News, Profil, Analyse
8	515100.F	BASF AG	Frankfurt	16:52	45,6	0,45	1,00%	77.606	Chart, News, Profil, Analyse



Ist eine Zelle des Datenbereichs einer Abfrage selektiert, so ermöglicht der nur dann verfügbare Befehl **DATEN|EXTERNE DATEN IMPORTIEREN|ABFRAGE BEARBEITEN...** bzw. das zugehörige Symbol der automatisch eingeblendeten Symbolleiste *Externe Daten*, eine Abfrage nachträglich zu editieren und öffnet dazu das zuvor gezeigte »Webseitenfenster« mit den hervorgerufenen Abschnitten.

Sie können darin andere Abschnitte auswählen als zuvor oder mit *Optionen...* andere Formatierungsoptionen wählen oder gar in der Adressleiste eine andere zu importierende Webseite auswählen.

28.3 Daten aktualisieren

Ist eine Zelle des Datenbereichs einer Abfrage selektiert, ermöglicht der Befehl DATEN|DATEN AKTUALISIEREN bzw. das zugehörige Symbol der *Externe Daten*-Symbolleiste die Aktualisierung der importierten Daten.



Die Abfrage wird dadurch erneut ausgeführt, also die Verbindung zur betreffenden Webseite wieder hergestellt und die nun aktuellen Daten in die Tabelle eingetragen.

Enthält Ihre Arbeitsmappe mehrere Webabfragen, können Sie durch das abgebildete Symbol mit einem Knopfdruck alle Abfragen aktualisieren.



Dauert Ihnen das dann zu lange oder haben Sie es sich einfach anders überlegt, ermöglicht dieses Symbol, die Aktualisierung abubrechen.



28.4 Abfrage-Eigenschaften und Parameter

Beim Öffnen einer bestehenden Webabfrage mussten Sie zunächst festlegen, wohin die Daten importiert werden sollen. Zusätzlich enthielt dieses Dialogfeld zwei von mir bisher übergangene Schaltflächen *Eigenschaften...* und – nur bei Parameterabfragen – *Parameter...*.

Die zugehörigen Dialogfelder können Sie auch im Anschluss an den Import öffnen, allerdings werden die vorgenommenen Änderungen erst nach dem Aktualisieren der Abfrage wirksam.

Das *Eigenschaften*-Dialogfeld öffnen Sie mit dem Befehl DATEN|EXTERNE DATEN IMPORTIEREN|DATENBEREICHSEIGENSCHAFTEN... – vorausgesetzt, eine Zelle des Datenbereichs ist selektiert (Abbildung 28.10).



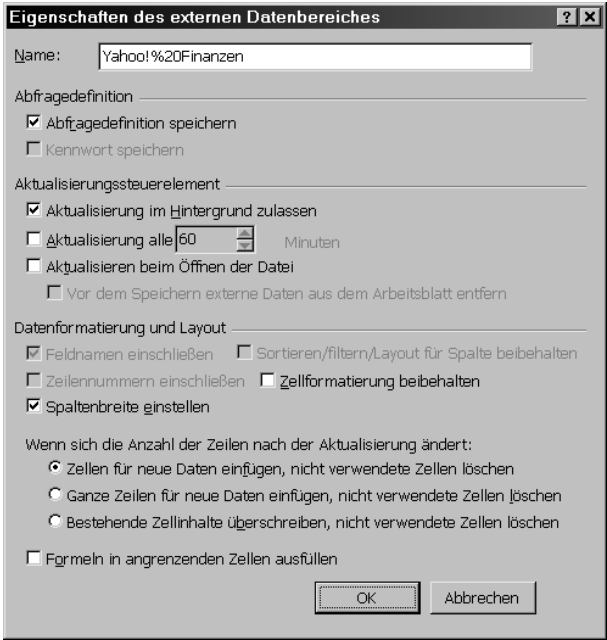
Es ermöglicht, verschiedene Eigenschaften der Abfrage festzulegen, vor allem ob nur Daten importiert werden oder »im« importierten Datenbereich auch die Abfragedefinition gespeichert wird, sodass Sie die importierten Daten später jederzeit wie erläutert aktualisieren können (ohne Speicherung der Abfragedefinition ist keinerlei nachträgliche Aktualisierung möglich).

*Abfrage-
Eigenschaften*

Weiterhin können Sie darin festlegen, ob die Abfrage automatisch ausgeführt werden soll, um die Daten beispielsweise in bestimmten Zeitabständen oder einmalig nach dem Öffnen der Arbeitsmappe zu aktualisieren.

Und Sie legen fest, was passieren soll, wenn sich die Größe des Datenbereichs nach dem Aktualisieren ändert und nun mehr Zeilen darin enthalten sind; ob dann vorhandene Zellen des angrenzenden Bereichs einfach überschrieben oder Zellen bzw. komplette Zeilen eingefügt werden sollen, um ein Überschreiben zu vermeiden.

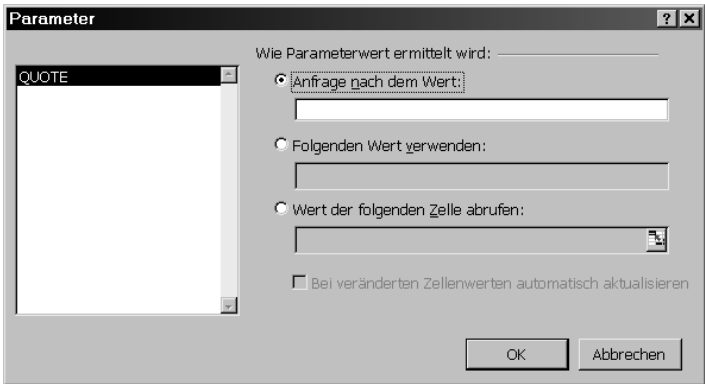
Abbildung 28.10:
Abfrage-
Eigenschaften



Abfrageparameter

Das *Parameter*-Dialogfeld ist nur verfügbar, wenn eine Zelle des Datenbereichs einer Parameterabfrage selektiert ist. Sie öffnen es mit DATEN|EXTERNE DATEN IMPORTIEREN|PARAMETER... (Abbildung 28.11).

Abbildung 28.11:
Abfrageparameter



Dieses Dialogfeld bietet kaum mehr Möglichkeiten als das bereits in Kapitel 28.1, »Ausführen«, besprochene Dialogfeld zur Parametereingabe. Wie mit den dort beschriebenen und vor Ausführung der Abfrage benutzten Kontrollkästchen legen Sie hier im Nachhinein fest, ob die Parameter jedesmal neu abgefragt werden sollen (Anfrage nach dem Wert) oder ob stattdessen feste Werte (geben Sie die Werte unter Folgenden Wert verwenden ein, im genannten Kapitel beispielsweise die Aktienkurs-Symbole »msft«, »intc«,

»ko« und »dis«) oder die Inhalte bestimmter Zellen (geben Sie unter Wert der folgenden Zelle abrufen den Zellbezug ein) als Parameter verwendet werden sollen.

28.5 Aktienkurse abfragen per Smart Tag

In den vorhergehenden Kapiteln verwendete ich als Beispiel die Abfrage von Aktienkursen. Diese Aufgabe kann auch auf einem anderen Weg gelöst werden – momentan allerdings nur für amerikanische Aktienkurse.

Angenommen, in Ihrem Webbrowser wird eine Webseite mit Kursen amerikanischer Aktien angezeigt. Sie markieren die Sie interessierenden Kurse, befördern Sie mit BEARBEITEN|KOPIEREN in die Zwischenablage und fügen sie anschließend mit BEARBEITEN|EINFÜGEN in die Excel-Tabelle ein.

Excel erkennt die im eingefügten Bereich enthaltenen amerikanischen Wertpapier-Symbole wie »MSFT« oder »KO« und blendet das Aktienkurs-Smart Tag ein (Abbildung 28.12).

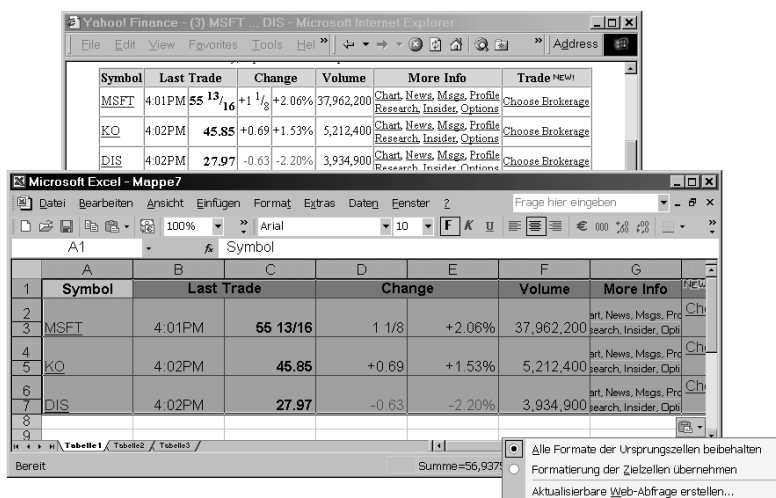


Abbildung 28.12:
Aktienkurs-
Smart Tag

Die beiden oberen Befehle ermöglichen nachträglich die Formatierung des eingefügten Bereichs festzulegen. Weit interessanter ist der Befehl AKTUALISIERBARE WEB-ABFRAGE ERSTELLEN..., der das Dialogfeld zum Importieren von Daten aus Webseiten öffnet (Abbildung 28.13).

Sie wählen darin wie erläutert die zu importierenden Abschnitte aus und Excel erstellt eine Webabfrage, die auf exakt die gleiche Webseite zugreift und die ausgewählte(n) Tabelle(n) importiert. Diese kann von Ihnen wie beschrieben jederzeit aktualisiert werden.

Abbildung 28.13:
Aktienkurse
importieren

Symbol	Last Trade	Change	Volume	More Info
MSFT	4:01PM 55 ¹³ / ₁₆	+1 ¹ / ₈ +2.06%	37,962,200	Chart , News , Msgs , Profile , Research , Insider , Options
KO	4:02PM 45.85	+0.69 +1.53%	5,212,400	Chart , News , Msgs , Profile , Research , Insider , Options
DIS	4:02PM 27.97	-0.63 -2.20%	3,934,900	Chart , News , Msgs , Profile , Research , Insider , Options

Add to My Portfolio - Set Alert Non-Tables Version - Dow

Select a Symbol for a detailed quote. Quotes delayed 15 minutes for Nasdaq, 20 minutes otherwise. Quote data provided by Reuters.

Importieren Abbrechen



Ein Tipp speziell zu internationalen Aktienkursen: Nach dem Import gibt es das Problem, dass eine Weiterverarbeitung dieser Kurse kaum möglich ist. Warum nicht? Schauen Sie sich die vorige Abbildung an: Das Zauberwort heißt »Dezimalpunkt« statt »Dezimalkomma«. »75.65« ist für Excel nun mal keine Zahl – im Gegensatz zu »75,65«. Also resultiert jede Formel, die sich auf eine Zelle mit einer solchen Zahl bezieht, in einem Fehlerwert.

Ein kleiner Trick löst das Problem: Bevor Sie die Kurse importieren, sei es per Drag&Drop oder per Webabfrage, schalten Sie auf den Dezimalpunkt als Dezimaltrennzeichen um. Entweder im Register *Zahlen* der *Windows-Ländereinstellung* (in der *Systemsteuerung*) oder aber direkt in Excel, im Register *International* von EXTRAS|OPTIONEN.

Importieren Sie nun einen Kurs wie »75.65«, so akzeptiert Excel das als numerischen Wert, als Zahl. Nach dem Import schalten Sie wieder auf das Dezimalkomma als Trennzeichen um und Excel ersetzt automatisch »75.65« durch »75,65« (was bei riesigen Kurstabellen übrigens recht lange dauern kann, also haben Sie Geduld).

29 Webserver einrichten

Ich will Ihnen noch ein paar Hilfestellungen dazu geben, wie Sie selbst Webseiten mithilfe eines eigenen Webserverns veröffentlichen, also anderen Anwendern zur Verfügung stellen können. Zum Einrichten eines echten Webserverns wird außer Excel zusätzliche Spezialsoftware benötigt. An Hardware ist theoretisch zwar nur ein einziger Rechner notwendig, im Idealfall besteht die Rechnerkonfiguration jedoch aus

- ➔ einem dedizierten Webserver, einem Rechner, der einzig und allein die Aufgabe besitzt, ständig für Zugriffe auf Ihre Webseiten aus dem Internet oder Intranet zur Verfügung zu stehen
- ➔ und einem damit verbundenen zweiten Rechner, auf dem Sie Ihre Webseiten erstellen und warten und auf dem die Resultate getestet werden.

Webserver-Software gibt es von Microsoft in zwei Varianten:

- ➔ Der Internet Information Server, der ausschließlich unter Windows NT läuft, aber nicht unter Windows 95/98
- ➔ Der Personal Web Server, der außer unter NT auch unter Windows 95/98 läuft

Aufgrund seiner niedrigeren Anforderungen verwende ich letzteren für meine Erläuterungen, da ich schlecht voraussetzen kann, dass jeder von Ihnen über Windows NT verfügt.

Den Personal Web Server gibt es jedoch in mehreren Versionen. Die ältere Version 3.0, die sich unter anderem auf der Frontpage 98-CD befindet, kann außer als Webserver auch als FTP-Server benutzt werden, ermöglicht also das Speichern/Öffnen von Dateien im Intranet/Internet gemäß dem allgemeinen und verbreiteten FTP-Standard.

Die Version 4.0 dagegen, die sich als Zugabe auf der Windows 98-CD befindet und zusätzlich von der Microsoft-Homepage downgeloadet werden kann, unterstützt unter Windows 95/98 kein FTP mehr, dafür jedoch den Microsoft-eigenen Standard zum Speichern/Öffnen von Dateien im Intranet/Internet, die Webordner.

Das ist ein wesentlich weniger verbreiteter Standard, da Microsoft jedoch in allen Office-Produkten seinen eigenen Webordner-Standard propagiert, werde ich für meine Erläuterungen den Personal Web Server 4.0 verwenden.

Wie gesagt, alternativ dazu können Sie auch den PWS 3.0 einsetzen oder – nur unter NT – gar den Internet Information Server, wodurch sich nur Details der folgenden Kapitel ändern, da die erläuterten Prinzipien für alle Webserver gleichermaßen gültig sind.

Das Grundprinzip:

- ➔ Auf Rechner A, dem Webserver-Rechner, läuft die Webserver-Software und befinden sich die publizierten Webseiten. Werden diese Webseiten im Internet publiziert, muss der Rechner entsprechend ständig per Modem mit dem Internet verbunden sein und steht entweder in Ihrer Firma oder aber bei einem speziellen Internet-Serviceprovider. Werden die Webseiten dagegen nur im firmeninternen Intranet publiziert, ist er entsprechend in das Firmennetzwerk eingebunden.
- ➔ Die Webseiten werden möglichst nicht auf A erstellt. Dieser Rechner sollte ausschließlich für Zugriffe von Anwendern zur Verfügung stehen, die sich die publizierten Webseiten anschauen wollen. Gleichzeitig darauf Webseiten zu erstellen oder mit sonstigen Programmen zu arbeiten, wäre katastrophal, da jederzeit die Möglichkeit des Rechnerabsturzes bestünde und die ständige Verfügbarkeit des Webserver dadurch gefährdet wäre!
- ➔ Stattdessen werden die Webseiten auf einem Rechner B erstellt, der mit A verbunden ist, üblicherweise wohl per Netzwerkverbindung.

29.1 Netzwerk konfigurieren

Der sinnvolle Einsatz eines Webserver setzt somit zumindest zwei Rechner voraus, die über ein Netzwerk miteinander verbunden sind. Welche Namen diese Rechner im Netzwerk besitzen, ist natürlich Ihre Sache, ich nenne sie im Folgenden »Haupt« und »Laptop«.

Warum? Ganz einfach: weil der Server in meinem Netzwerk nun mal »Haupt« heißt und ich einen Laptop besitze, den ich phantasieloserweise unter dem Namen »Laptop« in mein Netzwerk einband.

Das heißt, in den Netzwerkeigenschaften des Laptop ist im Register *Identification* unter *Computername* der Name »Laptop« eingetragen, unter dem dieser Rechner in meinem Netzwerk identifiziert wird (Abbildung 29.1).

Analog dazu ist unter *Computername* im Falle meines Webserver-Rechners »Haupt« eingetragen.

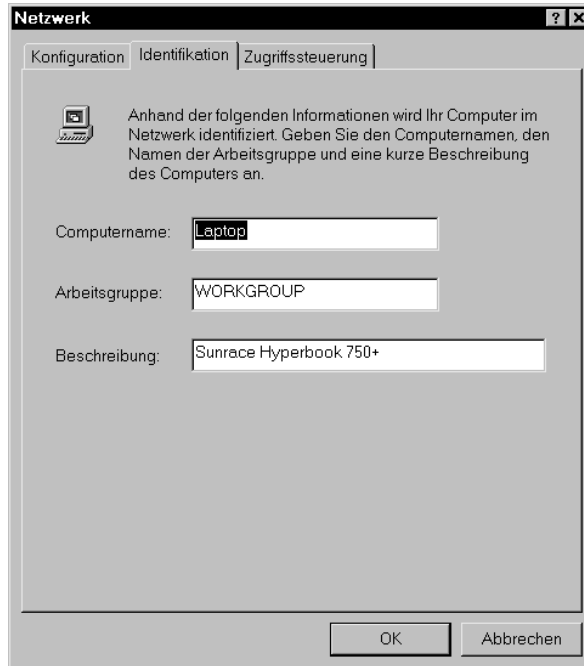


Abbildung 29.1:
Rechneridentifikation im Netzwerk

Mehr über Netzwerke zu sagen, wäre für Windows NT-Benutzer sicher höchst langweilig. Benutzen Sie NT, dann sind Sie im Zweifelsfall in Ihrer Firma DER Netzwerkexperte und ich kann Ihnen sowieso nichts Neues erzählen. Ist das der Fall, blättern Sie bitte einfach zum nächsten Abschnitt vor, zur Software-Konfiguration.

Bei Benutzern von Windows 95/98 liegt der Fall jedoch oft anders. Sie haben möglicherweise in Ihrem privaten Büro zu Hause oder in einem Kleinbetrieb ein Mini-Netzwerk aus mehreren Windows-Rechnern aufgebaut.

Windows 95 installierte dabei standardmäßig die Netzwerkprotokolle NetBEUI und IPX/SPX zur Kommunikation zwischen den Rechnern, was Sie sicherlich akzeptierten. Warum auch nicht? Mit diesen Protokollen funktioniert alles vollautomatisch und Sie können das Netzwerk umgehend und ohne Probleme benutzen.

In unserem Fall ist als Netzwerkprotokoll jedoch der Internet-Standard TCP/IP einzusetzen (Abbildung 29.2).

TCP/IP-Konfiguration unter Windows 95/98

Um es zu installieren, klicken Sie im Register *Konfiguration* der Windows-Netzwerkeigenschaften auf *Hinzufügen...* und wählen danach als hinzuzufügende Komponente »Protokoll« (Abbildung 29.3).

Abbildung 29.2:
Installierte Netzwerkprotokolle

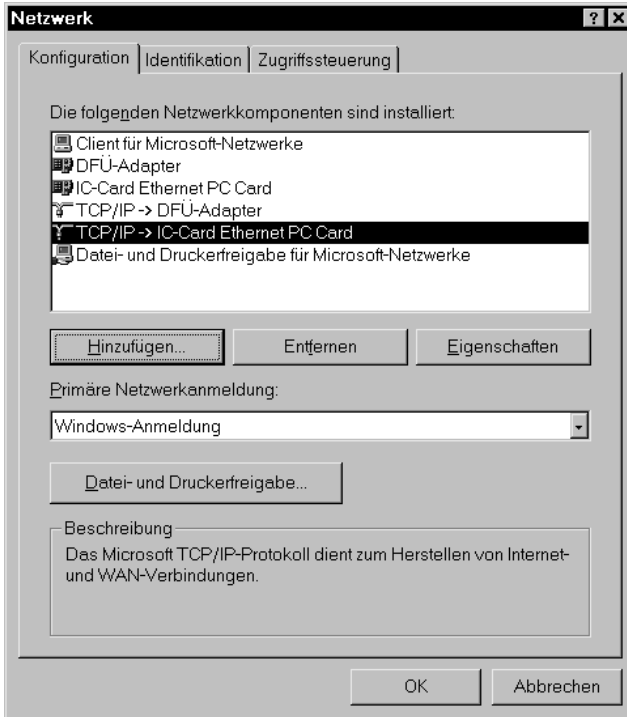
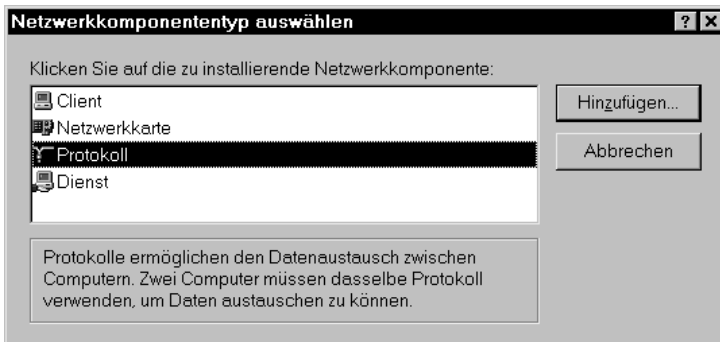


Abbildung 29.3:
Netzwerkprotokoll hinzufügen



Danach klicken Sie wieder auf *Hinzufügen...* und wählen im folgenden Dialogfeld als Hersteller »Microsoft« und als Netzwerkprotokoll »TCP/IP« (Abbildung 29.4).

Nach mehrfachem OK wird der Rechner neu gestartet und danach sollte im Register *Konfiguration* der Netzwerkeigenschaften das zusätzliche Protokoll TCP/IP für all Ihre Netzwerkadapter vorhanden sein.

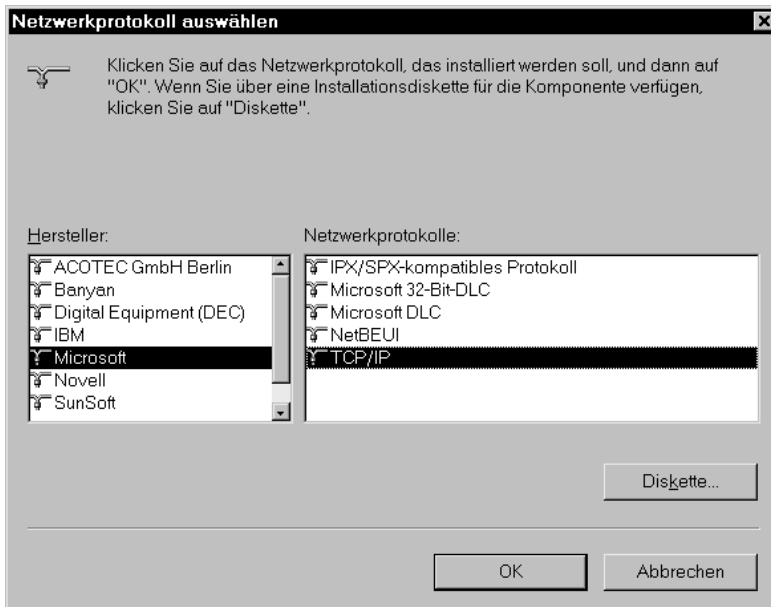


Abbildung 29.4:
TCP/IP hinzufügen

Anschließend sind die Eigenschaften dieses Protokolls einzustellen, genauer: die IP-Adresse. Eine IP-Adresse dient dazu, den betreffenden Rechner im Netzwerk eindeutig zu identifizieren.

NT-Server und Windows 98 können diese Adresse dynamisch zuweisen und Sie sind nun bereits fertig.

Bei Windows 95-Netzwerken ist die Sache komplizierter: Sie müssen jedem Rechner manuell eine eindeutige IP-Adresse zuweisen. Statt IP-Adressen lang und breit zu diskutieren, kürze ich das Ganze ab. Schauen Sie sich einfach die folgende Abbildung an (Abbildung 29.5).

Ein für kleine Netzwerke geeignetes Schema:

- ➔ Verwenden Sie als IP-Adresse auf dem ersten Rechner 192.168.0.1, auf dem zweiten Rechner 192.168.0.2 etc.
- ➔ Verwenden Sie bei jedem Rechner die gleiche Subnet-Mask 255.255.255.0.

Damit das Netzwerk funktioniert, ist ein weiterer Schritt notwendig: Im Windows-System müssen Sie eine Textdatei namens HOSTS (ohne Endung) erstellen, die die IP-Adressen aller in Ihrem Netzwerk miteinander verbundenen Rechner und die zugeordneten Rechnernamen enthält (Abbildung 29.6).

Abbildung 29.5:
IP-Adresse manuell
zuweisen

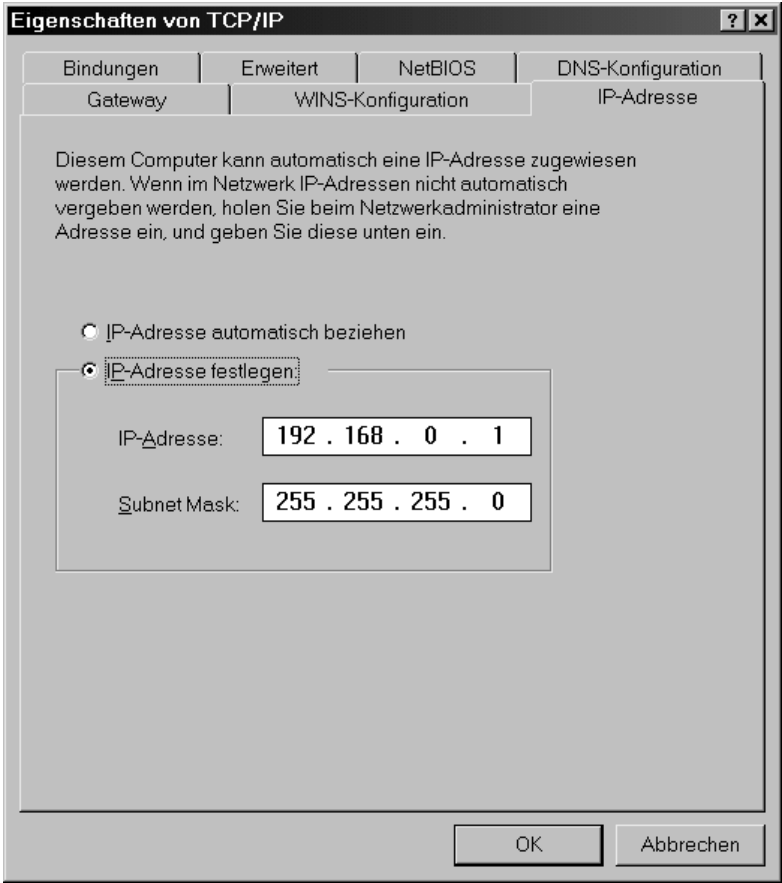
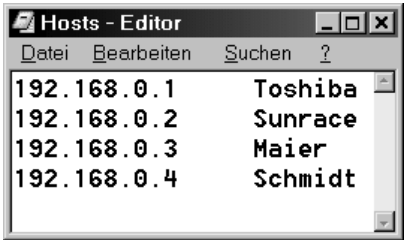


Abbildung 29.6:
Textdatei »Hosts«
im Verzeichnis
»Windows«



Diese Datei müssen Sie anschließend ins Verzeichnis WINDOWS aller anderen Rechner kopieren.

Nach dem Neustart von Windows auf allen Rechnern sollte das Netzwerk funktionieren. Wählen Sie im Windows-Explorer EXTRAS|SUCHEN|COMPUTER..., so sollten nach einem Klick auf *Starten* nach kurzer Zeit alle im Netzwerk miteinander verbundenen Rechner gefunden und ihre Namen angezeigt werden (Abbildung 29.7).

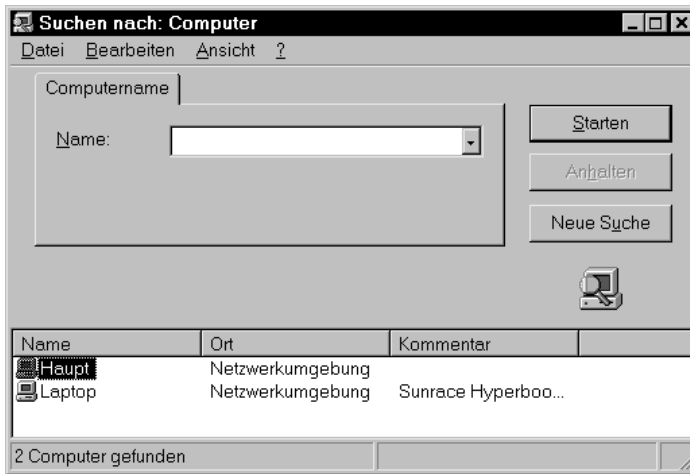


Abbildung 29.7:
Im Netzwerk
gefundene Rechner

Ist das nicht der Fall, benötigen Sie ein Buch, das im Detail auf Netzwerke eingeht – was ein Excel-Buch verständlicherweise nicht leisten kann.

Gehen wir davon aus, dass Ihr Windows-Netzwerk nun mit dem TCP/IP-Protokoll funktioniert. Dann ist der nächste Schritt die Installation und Konfiguration der Software.

29.2 Webserver konfigurieren

Wie gesagt, auf dem Server läuft entweder (unter Windows NT) Microsofts Internet Information Server oder aber (wahlweise unter Windows 95/98 oder NT) der Personal Web Server.

Da letzteres nicht zwangsläufig NT voraussetzt, erläutere ich die Installation und Konfiguration der Server-Software nun am Beispiel des Personal Web Servers – und setze dabei voraus, dass außer dem Personal Web Server zusätzlich auch ein Webbrowser wie zum Beispiel der Internet Explorer installiert ist.

Die Installation des Personal Web Servers ist trivial. Sie

*Personal Web
Server installieren*

- ➔ rufen SETUP.EXE im Verzeichnis ADD-ONS\PWS der Windows 98-CD auf oder
- ➔ starten das Setup-Programm nach dem Download des Personal Web Servers von der Microsoft-Homepage oder
- ➔ legen Ihre Frontpage-CD ein, worauf automatisch erkannt wird, dass noch kein Webserver installiert ist und Sie gefragt werden, ob der Personal Web Server nun installiert werden soll.

Nach beendeter Installation wird der Rechner neu gestartet – und danach können Sie auf Anhieb eigentlich keinen Unterschied zum vorhergehenden Zustand erkennen. Dennoch ist ab jetzt nach jedem Start des Servers im Hintergrund die Webserver-Software aktiv und kann benutzt werden.

Webserver- Eigenschaften

Schauen Sie sich die Taskleiste an, so werden Sie darin ein zusätzliches Symbol erkennen, das den im Hintergrund aktiven Webserver symbolisiert. Klicken Sie mit der rechten Maustaste darauf, können Sie

- ➔ mit *Dienst beenden* die Ausführung des Webservers beenden
- ➔ und mit *Dienst starten* wieder starten.

Vor allem jedoch können Sie mit *Eigenschaften...* (ebenso wie per Doppelklick auf das Symbol oder mit START|PROGRAMME|INTERNET EXPLORER|PERSONAL WEB SERVER|PERSONAL WEB-MANAGER) den Personal Web-Manager öffnen, der für die Verwaltung des Webservers zuständig ist (Abbildung 29.8).

Abbildung 29.8:
Webserver-
Eigenschaften



Zum Beispiel können Sie auch hier mit *Beenden* den Webserver beenden bzw. anschließend wieder mit *Starten* starten.

Im Gegensatz dazu sind viele Verwaltungs-Optionen des Webservers jedoch webbasiert. Klicken Sie beispielsweise auf den Link zur Homepage (hier: »<http://haupt>«), wird Ihr Webbrowser gestartet und zeigt das bei der Installation dort abgelegte Standarddokument an (Abbildung 29.9).



Abbildung 29.9:
Standarddokument

Beachten Sie, dass sich der Webserver »Haupt« im Intranet befindet, nicht im Internet! Je nach aktueller Einstellung fordert Sie der Internet Explorer nach dem Start zum Herstellen einer Internetverbindung auf (Abbildung 29.10).

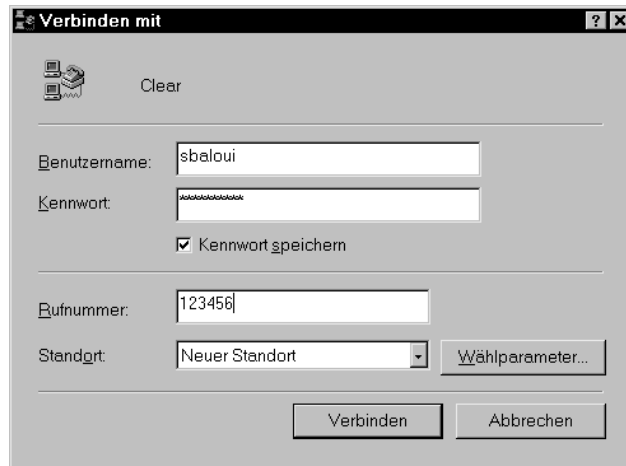


Abbildung 29.10:
Internetverbindung
herstellen

Das passiert, wenn im Internet Explorer im Register *Verbindung* (Befehl ANSICHT|INTERNETOPTIONEN...) die Option *Verbindung über ein Modem herstellen* aktiviert ist.

Aktivieren Sie stattdessen bitte die Option *Verbindung über ein lokales Netzwerk (LAN) herstellen*. Nach dem Schließen und erneuten Starten fordert Sie der Explorer zwar ebenfalls zum Herstellen einer Verbindung auf: Bestätigen Sie das, wird jedoch keine Modemverbindung ins Internet hergestellt, sondern Sie werden wie gewünscht mit Ihrem Intranet verbunden.

Resultat: Beim Aufbau der bisherigen Standard-Startseite, z.B. *http://home.microsoft.com/intl/de*, tritt eine Fehlermeldung auf, da mangels Internetverbindung nicht auf den Microsoft- oder einen sonstigen Server im Internet zugegriffen werden kann.

Zugriffe auf den Intranet-Server »*http://haupt*« sollten nun jedoch problemlos möglich sein, das heißt wenn Sie im Web-Manager erneut auf den Link zur Homepage klicken, sollte diese nun entsprechend Bild 29.9 erscheinen.

Basisverzeichnis

Das als Basisverzeichnis angegebene Verzeichnis, hier *C:\EIGENE DATEIEN\INETPUB\WWWROOT*, ist ab jetzt das wichtigste Verzeichnis dieses Rechners! Darin legen Sie nämlich die Webseiten ab, auf die andere Rechner per Internet- bzw. Netzwerkverbindung zugreifen sollen.

Dazu müssen Sie wissen, dass die Webserver-Software (egal, ob Personal Web Server oder Internet Information Server) ein beliebiges Verzeichnis der Serverfestplatte und ihrer Unterverzeichnisse für Zugriffe von anderen Rechnern aus zugänglich macht und *WWWROOT* das dazu verwendete Standardverzeichnis des Personal Web Servers ist.

**Webserver-
Installation testen**

Also sollten nun alle mit dem Server per Internet oder Intranet verbundenen Rechner auf Webseiten zugreifen können, die in diesem Verzeichnis abgelegt sind. Dazu ist auf jenen Rechnern nach dem Starten des Webbrowsers die Adresse dieses Basisverzeichnisses in folgender Form in der Adressleiste des Browsers anzugeben:

http://Servername

Diese Adresse ist aus Sicht anderer Rechner die Internet/Intranet-Adresse Ihres Webservers!

Ist Ihr Server beispielsweise im Domänen-Namen-System (DNS) als »*www.mycompany.de*« registriert, wäre also seine Internetadresse entsprechend

http://www.mycompany.de

in die Adressleiste des Browsers einzugeben.

Im lokalen Intranet geben Sie als Servername stattdessen den Windows-Netzwerk-Namen des Servers ein. Ist er wie im Beispiel unter dem Namen »Haupt« ins Netzwerk eingebunden, ist seine Internetadresse (eigentlich: Intranetadresse) entsprechend »*http://Haupt*« (siehe Bild 29.8).

Speichern Sie nun bitte irgendeine Webseite in diesem Verzeichnis. Starten Sie anschließend auf dem Server Ihren Webbrowser und tippen Sie in die Adressleiste ein:

`http://haupt/Dateiname`

Heißt die Webseite beispielsweise CHIPS.HTM, geben Sie entsprechend ein:

`http://haupt/Chips.htm`

Klappt der Zugriff, wird nun der Inhalt dieser im Basisverzeichnis gespeicherten Webseite im Webbrowser angezeigt (Abbildung 29.11).

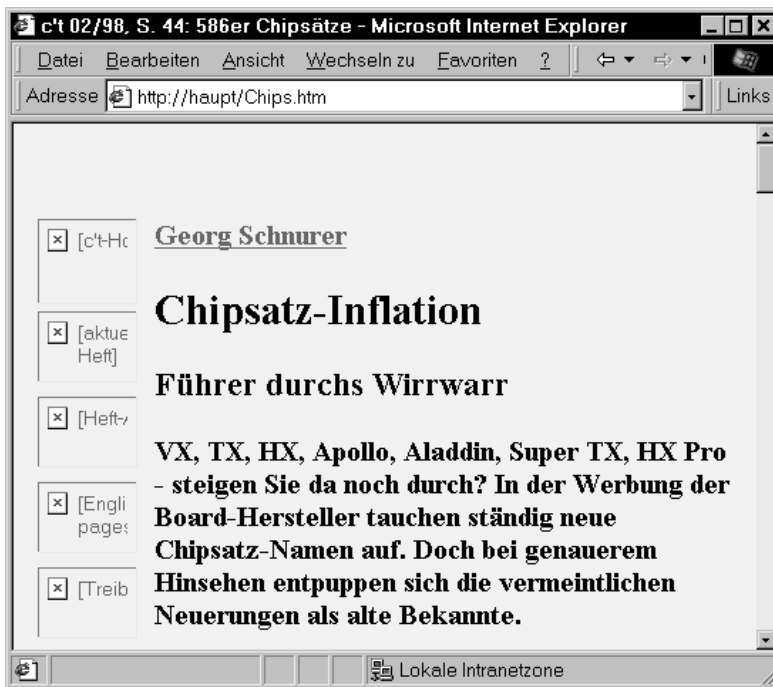


Abbildung 29.11:
Zugriff auf den
Webserver

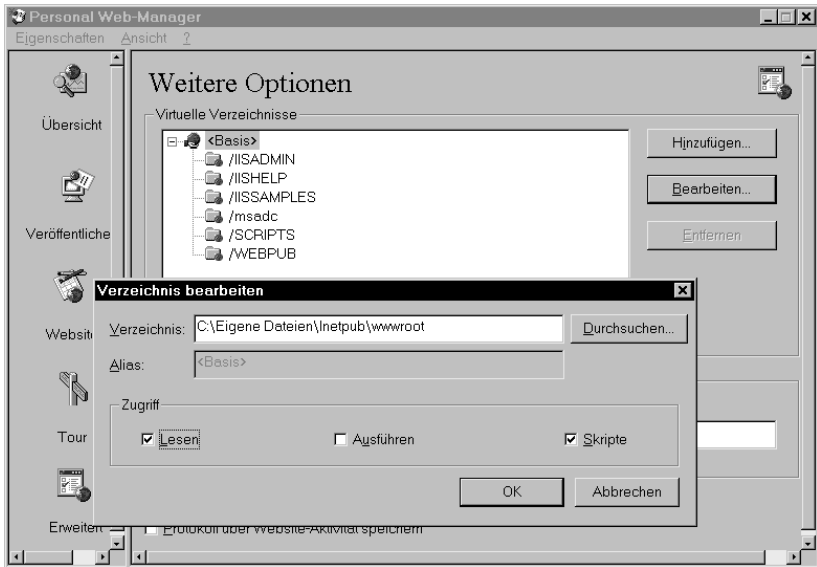
Im nächsten Schritt testen Sie, ob das gleiche auch von jedem anderen Rechner aus möglich ist, der im Intranet mit diesem Server verbunden ist. Wenn nicht, müssen Sie auch dort gegebenenfalls im Internet Explorer ANSICHT|INTERNETOPTIONEN... wählen und im Register *Verbindung* das Kontrollkästchen *Verbindung über ein lokales Netzwerk (LAN) herstellen* aktivieren.

Sind viele Webseiten zu verwalten, sollten Sie dazu im Verzeichnis WWW-ROOT zusätzliche virtuelle Verzeichnisse anlegen, die für darauf zugreifende Benutzer wie Unterverzeichnisse dieses Basisverzeichnisses erscheinen.

*Verzeichnis-
verwaltung*

Klicken Sie auf die Schaltfläche *Erweitert* in der linken Leiste des Web-Managers, werden alle momentan vorhandenen virtuellen Verzeichnisse angezeigt. Sie können ein beliebiges davon selektieren, beispielsweise das Basisverzeichnis »<Basis>«, und mit *Bearbeiten...* seine Eigenschaften ändern (Abbildung 29.12).

Abbildung 29.12:
Verzeichnis-
verwaltung



Das ermöglicht Ihnen beispielsweise, durch Ändern des unter *Verzeichnis* vorgegebenen Pfads (manuell oder mit *Durchsuchen...*) ein beliebiges anderes Verzeichnis zum neuen Basisverzeichnis zu machen.

Sinnvoller ist jedoch das Hinzufügen eigener virtueller Verzeichnisse, die Sie Ihren Anwendern unter einem geeigneten Alias-Namen zur Verfügung stellen, der den Zugriff erleichtert. Selektieren Sie »<Basis>« und klicken Sie auf *Hinzufügen...* (Abbildung 29.13).

Im zugehörigen Dialogfeld können Sie ein beliebiges Verzeichnis Ihrer Festplatte auswählen. Der zum Zugriff verwendete Alias-Name ist vollkommen unabhängig vom echten Namen des Verzeichnisses. Beispielsweise können Sie C:\EIGENE DATEIEN unter dem Alias-Namen MYWEBPAGES hinzufügen.

Dieses virtuelle Verzeichnis wird nun unterhalb des zuvor selektierten Basisverzeichnisses hinzugefügt und ist für den Benutzer unter der Adresse *http://haupt/MyWebPages* erreichbar.

Tippt er in seinem Browser »*http://haupt/MyWebPages/CanonLenses.htm*« ein, wird entsprechend die Webseite CANONLENSES.HTM angezeigt, die sich tatsächlich im realen Verzeichnis C:\EIGENE DATEIEN des Webserver befindet (Abbildung 29.14).



Abbildung 29.13:
Virtuelles
Verzeichnis
hinzufügen

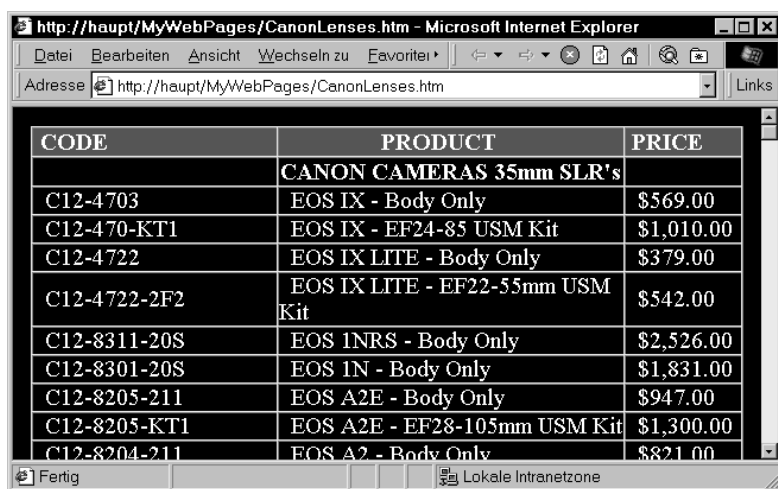


Abbildung 29.14:
Zugriff auf virtu-
elles Verzeichnis

Löschen Sie ein virtuelles Verzeichnis mit Entfernen, so wird es zwar aus der Liste der virtuellen Verzeichnisse entfernt – das reale Verzeichnis mit seinen Inhalten bleibt aber natürlich erhalten.

Die Verzeichniseigenschaften Lesen, Ausführen und SCRIPTE entscheiden darüber, welche Zugriffsmöglichkeiten der Anwender besitzt. Lesen ist zur Anzeige reiner Webseiten ausreichend, nicht jedoch beispielsweise zur Ausführung von Active Server Pages, mit denen beispielsweise dynamisch auf eine Datenbank zugegriffen wird.





Statt DEFAULT.HTM bzw. DEFAULT.ASP können Sie im Eingabefeld Standarddokument(e) (siehe Bild 29.13) eine beliebige andere Webseite als Standarddokument angeben, das angezeigt wird, wenn auf eines dieser Verzeichnisse ohne Angabe eines bestimmten Dateinamens zugegriffen wird (zum Beispiel mit `http://haupt` statt mit `http://haupt/Chips.htm`).

Sie können die dann erfolgende Standarddokument-Anzeige auch komplett unterbinden, indem Sie das Kontrollkästchen *Standarddokument aktivieren* deaktivieren.

29.3 Kritische Punkte

Wie Sie sehen, sind einige Voraussetzungen zu erfüllen, um erfolgreich Webseiten zu publizieren. Da man dabei leicht den Überblick verlieren kann, liste ich im Folgenden die kritischsten Punkte noch einmal auf:

- ➔ Auf dem Webserver-Rechner läuft entweder (unter Windows NT) Microsofts Internet Information Server oder aber (wahlweise unter Windows 95/98 oder NT) der Personal Web Server.
- ➔ Auf dem Rechner, der zur Erstellung der Webseiten verwendet wird (das kann notfalls der Webserver selbst sein) muss ein Webbrowser installiert sein.
- ➔ Beide Rechner müssen per Modem oder über Netzwerkkarten miteinander über das TCP/IP-Protokoll verbunden sein.
- ➔ Die Internetadresse des Webserver-Rechners ist der im Domänen-Namen-System (DNS) registrierte Name wie »`http://www.mycompany.de`« oder – Zugriff über lokales Netzwerk – der Name dieses Rechners im Windows-Netzwerk, beispielsweise »`http://Haupt`«.
- ➔ Ein Zugriff auf den Webserver über die Eingabe dieses Namens in der Browser-Adressleiste (des Servers selbst oder eines damit verbundenen Rechners) ist gleichbedeutend mit dem Zugriff auf das Basisverzeichnis (Standard: WWWROOT).
- ➔ Mit den Administrationstools der Webserver-Software kann das als Basisverzeichnis genutzte Verzeichnis geändert und es können ihm zusätzliche Verzeichnisse als virtuelle Verzeichnisse hinzugefügt werden, die von außen wie Unterverzeichnisse des Basisverzeichnisses erscheinen.

30 Sonstiges

30.1 Weboptionen

Das Register *Allgemein* des Befehls EXTRAS|OPTIONEN... enthält eine Schaltfläche *Weboptionen...*, die ein Dialogfeld mit verschiedensten Optionen öffnet, die beim Speichern einer Excel-Tabelle als Webseite interessant sind (Abbildung 30.1).

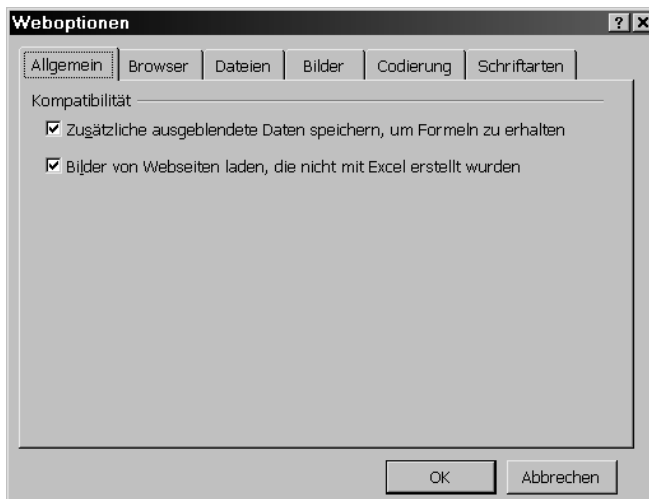


Abbildung 30.1:
Web-Optionen

Ich versuche Ihnen nun anhand einiger Beispiele einen Überblick über die verschiedenen Möglichkeiten zu geben:

➔ Das Register *Allgemein* enthält Optionen, die Microsoft selbst nicht anders einordnen konnte, zum Beispiel die Option zum Speichern aller versteckten Daten, die zum Erhalt von Formeln notwendig sind.

? Gemeint ist, dass bei aktivierter Option beim Speichern einer Excel-Mappe als Webseite alle externen Daten mitgespeichert werden, auf die die darin enthaltenen Formeln verweisen, sodass diese Formeln bei Änderungen der externen Daten aktualisiert werden können (bei deaktivierter Option wird dagegen nur das aktuelle Formelresultat in der Webseite gespeichert und eine Aktualisierung ist nicht möglich).

- ➔ Das Register *Browser* ist für die Webbrowser-Kompatibilität zuständig. Die verschiedenen Kontrollkästchen entscheiden über die Größe der als Webseite gespeicherten Tabelle. Jedes aktivierte Kästchen verringert die Größe, allerdings ist bei jedem Browser gewährleistet, dass er die dadurch aktivierte Option auch versteht und die Webseite noch einwandfrei darstellt (oder beispielsweise Grafiken nicht erscheinen).

Daher sollten Sie zunächst im Listenfeld den »Minimum-Browser« auswählen, die älteste Version, auf die Sie noch Rücksicht nehmen können. Excel aktiviert daraufhin die Kontrollkästchen jener Optionen, die der betreffende Browser einwandfrei verarbeitet und es ist Ihre Entscheidung, ob Sie dennoch weitere Optionen aktivieren wollen oder nicht.

- ➔ Das Register *Dateien* bestimmt, was mit den verschiedenen Dateien passiert, die beim Speichern einer Excel-Tabelle als Webseite erstellt werden, beispielsweise ob die verschiedenen Hilfsdateien (in der Tabelle enthaltene Grafiken etc. werden als eigene Dateien gespeichert) im gleichen Ordner wie die Webseite gespeichert werden oder in einem eigenen Unterverzeichnis (das in jenem Verzeichnis angelegt wird, in dem Sie die Webseite speichern).
- ➔ Im Register *Bilder* legen Sie unter anderem fest, welche Minimalauflösung der Monitor des Betrachters besitzen muss. Je niedriger Sie die Auflösung wählen, desto wahrscheinlicher ist es, dass die Seite in guter Qualität auch bei Anwendern mit älteren Rechnern/Monitoren dargestellt wird, desto kleiner werden jedoch auch Bilder.
- ➔ Im Register *Kodierung* wählen Sie die Sprachkodierung des Dokuments aus, ASCII, Unicode etc.
- ➔ Im Register *Schriftarten* legen Sie den Zeichensatz fest und bestimmen beispielsweise, welche Schriftart und Größe für die Darstellung normaler proportionaler Schrift verwendet wird und in welcher Schriftart und Größe nicht proportionale Schrift dargestellt wird.



Das gleiche Dialogfeld erscheint, wenn Sie DATEI|SPEICHERN UNTER... wählen, im Dateiauswahl-Dialogfeld das Extras-Listenfeld öffnen und WEBOPTIONEN... wählen.

30.2 Bilder komprimieren

Wenn Sie DATEI|SPEICHERN UNTER... wählen, im *Dateiauswahl*-Dialogfeld das *Extras*-Listenfeld öffnen und BILD KOMPRIMIEREN... wählen, können Sie detailliert festlegen, ob und wie Bilder komprimiert werden sollen; das ist eine sinnvolle Option beim Speichern von Tabellen als Webseiten im Internet, um große Webseiten mit entsprechend langwierigem Download zu vermeiden (Abbildung 30.2).

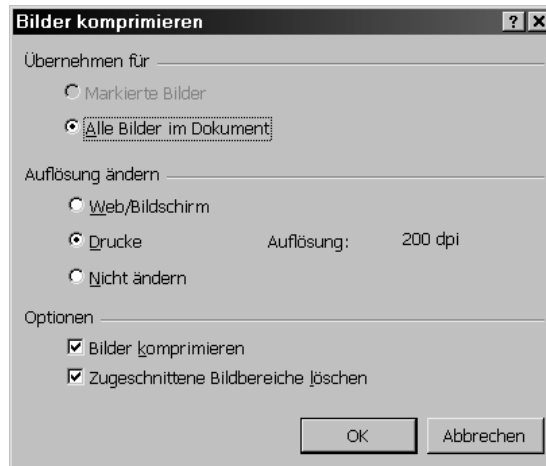


Abbildung 30.2:
Bildkomprimierung

Das gleiche Dialogfeld erscheint, wenn ein Bild selektiert ist und Sie in der Grafik-Symbolleiste auf das Symbol zum Komprimieren von Grafiken klicken.



Zunächst einmal können Sie mit dem entsprechenden Kontrollkästchen wählen, ob Bilder überhaupt komprimiert werden sollen. Das ist normalerweise eine sinnvolle Option, da dadurch die Dateigröße verringert wird – und zwar je nach Bildtyp gegebenenfalls sehr deutlich! Allerdings wird mit JPEG komprimiert, was bedeutet, dass die Bildqualität durch die Komprimierung vermindert wird. Falls Sie das nicht stört, lassen Sie die Komprimierung aktiviert.

Darüber hinaus können Sie wählen, ob nur zuvor selektierte Bilder oder alle in der Mappe enthaltenen Bilder komprimiert werden sollen. Sie können die Auflösung der Bilder verändern, wobei bei der Druckauflösung eine Auflösung von 200 dpi verwendet wird und Bilder damit weitaus mehr Speicherplatz benötigen als wenn Sie die Web-/Bildschirm-Option wählen, bei der Bilder nur mit 96 dpi gespeichert werden (mit schlechterer Qualität).

Und zuletzt können Sie den Speicherplatz von Bildern verringern, die Sie zuvor mit dem entsprechenden Symbol der *Grafik*-Symbolleiste beschnitten haben, indem Sie mit dem entsprechenden Symbol dafür sorgen, dass die abgeschnittenen Bildbereiche nicht mitgespeichert werden.

Teil 6 Makros und VBA-Programme

Kapitel 31: Makros	765
Kapitel 32: Visual Basic	773
Kapitel 33: Sprachelemente	817
Kapitel 34: Kontrollstrukturen	873
Kapitel 35: Objekte	903
Kapitel 36: Tabellen und Diagramme	919
Kapitel 37: Formulare und Steuerelemente	949
Kapitel 38: Praxisführer	985

Nachdem Sie nun mit so ziemlich allen Features von Excel vertraut sind, bleibt nur noch eines übrig: Ihnen zu zeigen, wie Sie sich die Arbeit vereinfachen und alle möglichen Tätigkeiten automatisieren und wie Sie Excel um zusätzliche Features erweitern können.

Die Automatisierung erfolgt mit Hilfe von Makros, das heißt der Aufzeichnung von Tastenbetätigungen und Mausektionen, die Sie anschließend jederzeit auf Befehl abspulen lassen können.

Excel um zusätzliche Features erweitern, das können Sie mit Visual Basic für Anwendungen (VBA), der in Excel enthaltenen Programmiersprache.

Sie ermöglicht Ihnen beispielsweise, eigene Funktionen zu definieren, wenn Ihnen die eingebauten Funktionen (*SUMME*, *MITTELWERT* etc.) nicht ausreichen.

Darüber hinaus können Sie mit VBA komplexe Berechnungen vereinfachen und mit Makros ebenfalls Abläufe automatisieren, integrierte Anwendungen erstellen, die selbstdefinierte Menüs und Dialogfelder besitzen und vieles mehr.

Ich werde zunächst den Umgang mit der VBA-Oberfläche erläutern, danach die Sprachelemente und Kontrollstrukturen, die VBA Ihnen zur Verfügung stellt, und Ihnen anschließend zeigen, wie mit dieser Sprache Excel-Objekte wie Tabellen und Diagramme erzeugt und verändert, eigene Dialogfelder erstellt und sogar beliebige Dateien manipuliert werden können.

31 Makros

Im folgenden Kapitel geht es um Makros, also um die Aufzeichnung von Tastenbetätigungen und Mausektionen, die Sie anschließend jederzeit ausführen, das heißt ihren Inhalt auf Befehl abspulen lassen können.

Ich zeige Ihnen, wie Sie derartige Aktionen mit Hilfe des Makro-Recorders aufzeichnen, Ihre Makros benennen und ausführen.

Vor allem aber erläutere ich, welche Optionen zum Speichern von Makros zur Verfügung stehen. Die Wahl der Speicheroption beeinflusst nämlich in großem Maße die Verfügbarkeit Ihrer Makros und legt beispielsweise fest, dass ein bestimmtes Makro immer verfügbar ist bzw. erst nach dem Öffnen der Mappe, in der es gespeichert ist.

Aufgezeichnete Makros können Sie auch nachträglich noch bearbeiten, um beispielsweise die zur Ausführung festgelegte Tastenkombination zu ändern oder aber den Inhalt des Makros selbst, den Makro-Code – letzteres setzt jedoch voraus, dass Sie über die erst in den folgenden Kapiteln erläuterten Visual Basic-Kenntnisse verfügen.

Ich zeige Ihnen, wie Sie Auto-Makros erzeugen, die automatisch ausgeführt werden, beispielsweise nach dem Öffnen einer Mappe, und wie Sie Makros mit Objekten verknüpfen, um ein Makro durch Anklicken eines in die Tabelle integrierten Grafikobjekts oder durch Klicken auf ein von Ihnen in einer Symbolleiste hinzugefügtes Symbol oder einen Menübefehl zu starten.

31.1 Der Makro-Recorder

Wie gesagt, Makros sind Aufzeichnungen von Tastenbetätigungen und Mausektionen. Sie können zum Beispiel die Tastenfolge aufzeichnen, die Sie benötigen, um eine Tabelle auszudrucken, und ihr den Namen »Drucken« geben.

Jeder Schritt wird aufgenommen, die Anwahl des Menüs DATEI, die Wahl des Kommandos DRUCKEN... und die Einstellungen im folgenden Dialogfeld. Alle diese normalerweise manuell durchzuführenden Schritte spult Excel automatisch ab, wenn Sie das Makro »Drucken« aufrufen.

31.1.1 Aufzeichnen und ausführen

Ein etwas theoretisches Beispiel: Nehmen wir an, Sie haben ein Tabellenblatt, das Sie täglich für Ihre Arbeit benötigen. Ab und zu müssen die im Bereich B1:E6 enthaltenen Daten als Diagramm dargestellt werden. Wird es nicht benötigt, wollen Sie jedoch auf das Diagramm verzichten, um das tägliche Öffnen und Speichern der Mappe zu beschleunigen.

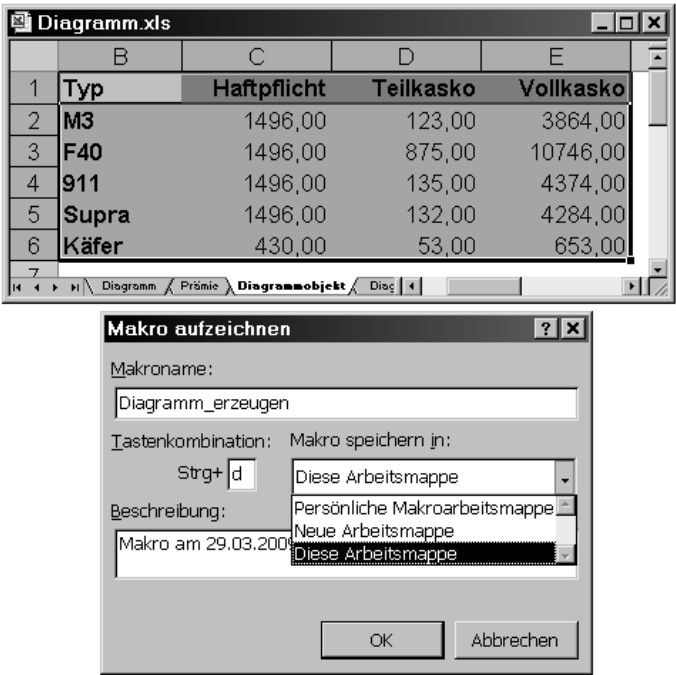
Also müssen Sie in der Lage sein, dieses Diagramm auf Knopfdruck zu löschen und bei Bedarf ohne Aufwand wieder erzeugen zu können.

Den Knopfdruck zum Löschen kennen Sie: die Taste `[Entf]`. Um das Diagramm auch auf Knopfdruck zu erzeugen, benötigen Sie jedoch ein Makro, in dem sie die dazu benötigten Schritte aufzeichnen.



Um diese immer wiederkehrenden Schritte als Makro aufzuzeichnen, selektieren Sie den Bereich B1:E6 und wählen EXTRAS|MAKRO|AUFZEICHNEN... oder klicken auf das entsprechende Symbol (Abbildung 31.1).

Abbildung 31.1:
Makroname und
Makrobeschreibung



Im Feld *Makroname* gibt Excel »Makro1«, »Makro2« etc. vor. Sie ersetzen die Vorgabe durch einen sinnvolleren Namen wie »Diagramm_erzeugen« (Leerzeichen sind in Makronamen nicht erlaubt) und geben unter *Beschreibung* eventuell eine Kurzbeschreibung ein.

Sie können ein Makro später jederzeit mit EXTRAS|MAKRO|MAKROS... ausführen lassen. Zusätzlich können Sie jedoch in diesem Dialogfeld unter *Tastenkombination* eine Taste definieren, die – zusammen mit der [Strg]-Taste gedrückt – das Makro ebenfalls ausführt. Geben Sie »d« ein, wird das Makro später ausgeführt, wenn Sie [Strg]+[D] drücken.

Ein Makro wird von Excel zusammen mit der Arbeitsmappe gespeichert. Nur wenn die Mappe geöffnet ist, sind die darin enthaltenen Makros verfügbar und können ausgeführt werden. *Makro speichern in* legt fest, in welcher Mappe ein Makro gespeichert wird:

➔ *Persönliche Makro-Arbeitsmappe* speichert das Makro in einem Modul einer Mappe namens PERSONL.XLS im Verzeichnis XLSTART. Sie wird bei jedem Start von Excel automatisch geöffnet, sodass die darin enthaltenen Makros jederzeit verfügbar sind.

Diese Mappe ist normalerweise ausgeblendet, kann jedoch mit FENSTER|EINBLENDEN... jederzeit ein- und mit FENSTER|AUSBLENDEN wieder ausgeblendet werden.

➔ *Diese Arbeitsmappe* speichert das Makro in der momentan aktiven Mappe.

➔ *Neue Arbeitsmappe* speichert es in einer neu angelegten Mappe.

OK startet die Aufzeichnung.

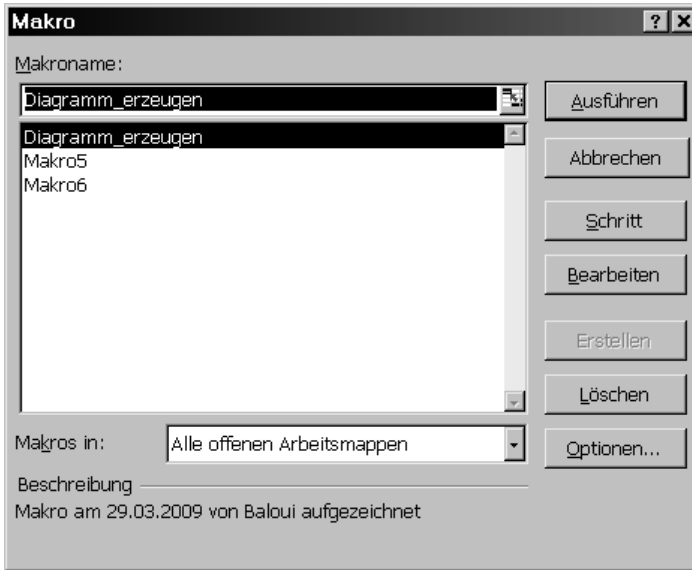
Ab jetzt registriert der Makro-Recorder jeden Ihrer Schritte:

1. Wählen Sie EINFÜGEN|DIAGRAMM... .
2. Klicken Sie im aufgerufenen Diagramm-Assistenten auf *Fertig stellen*, um das Diagramm mit den Standardeinstellungen zu erzeugen.
3. Beenden Sie die Aufzeichnung, indem Sie auf das bei Beginn der Aufzeichnung automatisch eingeblendete Symbol klicken oder EXTRAS|MAKRO|AUFZEICHNUNG BEENDEN wählen.
4. Um die aufgezeichneten Schritte abspulen zu lassen, das Diagramm also bei Bedarf auf Knopfdruck zu erzeugen, wählen Sie EXTRAS|MAKRO|MAKROS... oder klicken auf das zugehörige Symbol (Abbildung 31.2).
5. Das Listenfeld enthält die Namen aller bisher aufgezeichneten Makros. *Ausführen* führt ebenso wie ein Doppelklick auf einen Makronamen das selektierte Makro aus, ruft also den Diagramm-Assistenten auf und erzeugt ein auf dem Bereich B1:E6 basierendes Diagramm.

Einfacher können Sie das Makro über den zugehörigen Tastenschlüssel ausführen lassen, beispielsweise [Strg]+[D].



Abbildung 31.2:
Makro ausführen



31.1.2 Bearbeiten



Um Makros, die in der persönlichen Makro-Arbeitsmappe gespeichert sind, zu bearbeiten, müssen Sie diese zuvor mit FENSTER|EINBLENDEN... einblenden.

Die Bearbeitung von Makros erfolgt vorwiegend mithilfe des Makro-Dialogfelds, das wie erläutert mit dem Befehl EXTRAS|MAKRO|MAKROS... geöffnet wird (siehe Bild 31.2). Beispielsweise löscht die Schaltfläche *Löschen* das momentan selektierte Makro.



Schritt ermöglicht die schrittweise Ausführung des Makros, was das Auffinden von Fehlern erleichtert und von mir ausführlich in Kapitel 32.2.3, »Das Direktfenster«, und im Kapitel 32.2.4, »Fehlerverfolgung«, besprochen wird.

Optionen... öffnet eine vereinfachte Variante des in Bild 31.1 gezeigten Dialogfelds, in dem Sie die Beschreibung des Makros oder den zugeordneten Tastenschlüssel ändern können.

Makro-Code bearbeiten

Zusätzliche Möglichkeiten bietet die direkte Bearbeitung des Makro-Codes, die durch die Schaltfläche *Bearbeiten* aktiviert wird (Abbildung 31.3).

Dieses Fenster ist leider bedeutungslos für Sie, solange Sie nicht in Visual Basic programmieren können.

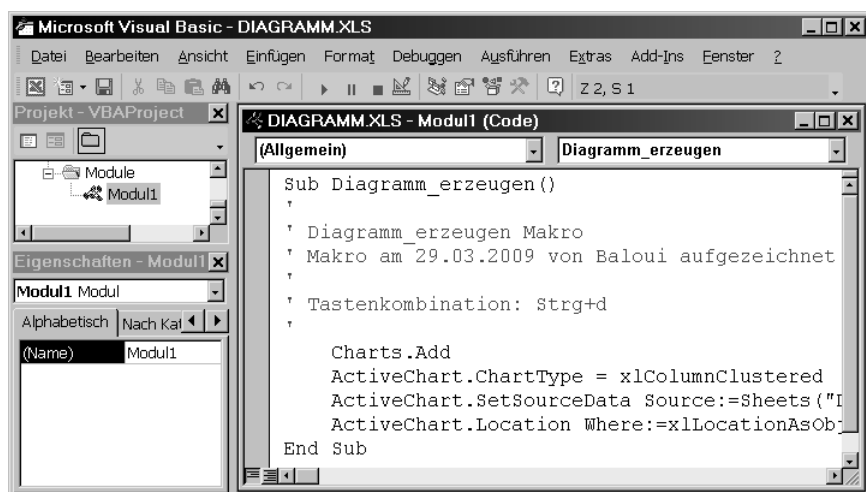


Abbildung 31.3:
Aufgezeichnetes
Makro

31.2 Auto-Makros

Eine sehr nützliche Sache sind die Auto-Makros, die beim Öffnen und/oder Schließen einer Datei oder beim Aktivieren/Deaktivieren eines Dateifeners automatisch ausgeführt werden. Dazu nennen Sie Ihr Makro einfach »Auto_Open«. Öffnen Sie die Mappe, die dieses »Auto_Open«-Makro enthält, so führt Excel es automatisch aus.

Wollen Sie eine Tabelle öffnen, die ein solches Makro enthält, ohne dass es automatisch ausgeführt wird, so halten Sie die -Taste gedrückt, während Sie im Dialogfeld des Befehls DATEI|ÖFFNEN... Öffnen aktivieren (bzw. während Sie im unteren Teil des DATEI-Menüs auf den betreffenden Dateinamen klicken).

Sollen nach dem Aufruf von Excel automatisch irgendwelche Aktionen ausgeführt werden, zeichnen Sie diese auf, geben dem Makro den Namen »Auto_Open« und speichern die betreffende Mappe im Verzeichnis XLSTART, da alle darin enthaltenen Dateien nach dem Start von Excel automatisch geöffnet werden.

Excel ermöglicht die Definition dreier weiterer »Auto-Makro-Typen«:

- ➔ Enthält eine Tabelle den Namen »Auto_Close«, wird das Makro, auf das er sich bezieht, immer dann automatisch ausgeführt, wenn die Tabelle geschlossen wird.
- ➔ Enthält die Tabelle den Namen »Auto_Activate«, wird das zugehörige Makro immer dann ausgeführt, wenn das Tabellenfenster aktiviert wird.

- ➔ Enthält die Tabelle den Namen »Auto_Deactivate«, wird das zugehörige Makro immer dann ausgeführt, wenn das Tabellenfenster deaktiviert wird.

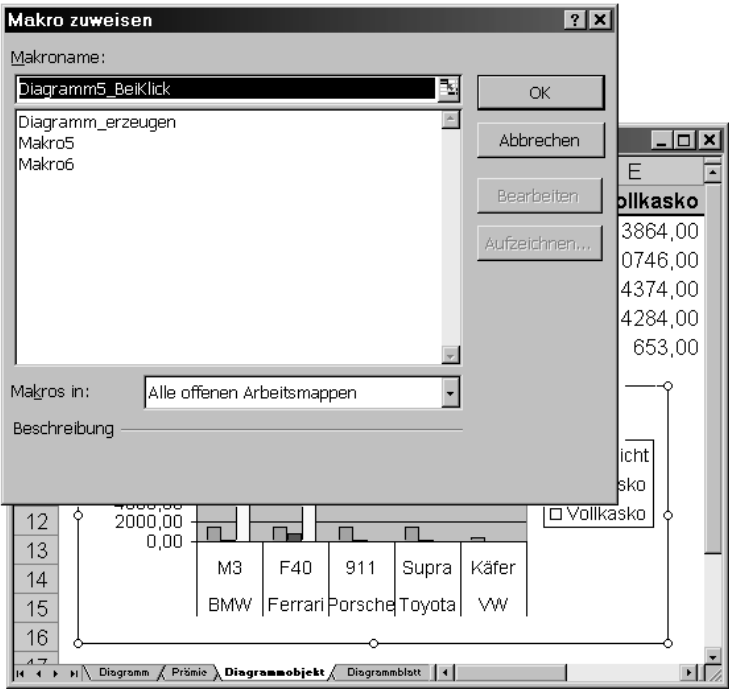
31.3 Makros mit Objekten verknüpfen

Makros können mit nahezu beliebigen Objekten verknüpft sein, mit Grafikobjekten wie Textboxen, Linien, Kreisen, mit Schaltknöpfen oder mit Symbolleisten-Symbolen (ohne Listenfeld). In all diesen Fällen geht es darum, dass nach Anklicken des betreffenden Objekts das zugehörige Makro ausgeführt werden soll.

Das Prinzip: Sie klicken mit der rechten Maustaste auf das Grafikobjekt und wählen im Kontextmenü **MAKRO ZUWEISEN...** . Ein Dialogfeld erscheint, in dem die Verknüpfung definiert und – falls es noch nicht existiert – das zugehörige Makro aufgezeichnet wird.

Um einem Symbol einer Symbolleiste ein Makro zuzuweisen, öffnen Sie zuvor das *Anpassen*-Dialogfeld. Der weitere Ablauf ist identisch: Sie klicken mit der rechten Maustaste auf das Symbol und wählen im Kontextmenü **MAKRO ZUWEISEN...** (Abbildung 31.4).

Abbildung 31.4:
Verknüpfung
definieren



Im Listenfeld sind alle in der aktuellen Mappe enthaltenen Makros aufgelistet. Sie können eines davon selektieren und es dem Objekt mit OK zuweisen.

Weisen Sie dem Objekt beispielsweise das im letzten Abschnitt aufgezeichnete Makro »Diagramm_erzeugen« zu, können Sie dieses Makro anschließend durch Anklicken des Objekts (auf dem Objekt wird der Cursor zu einer zeigenden Hand) ausführen lassen.

Existiert das mit dem Objekt zu verknüpfende Makro noch nicht, wählen Sie *Aufzeichnen...*: Das vom Befehl EXTRAS|MAKRO|AUFZEICHNEN... her bekannte Dialogfeld erscheint und ermöglicht Ihnen das Aufzeichnen des Makros. Nach dem Beenden der Aufzeichnung wird das Makro mit dem Objekt verknüpft.

Vergessen Sie nicht, die Symbolleiste an die Arbeitsmappe anzubinden, wenn Sie die Mappe an einen Kollegen weitergeben wollen (siehe Kapitel 38.1, »Menü- und Symbolleisten anpassen«). Sonst wird zwar die Mappe selbst weitergegeben, die zugehörige Symbolleiste fehlt jedoch!



32 Visual Basic

Um die Prozeduren in den Demos auf der Begleit-CD ausführen zu können, darf die Makrosicherheitsstufe nicht auf »Hoch« eingestellt sein. Wählen Sie EXTRAS|MAKRO|SICHERHEIT... und im folgenden Dialogfeld im Register Sicherheitsstufe entweder Niedrig oder – vorzuziehen – Mittel (dann werden Sie nach dem Öffnen einer Mappe gefragt, ob darin enthaltene Makros aktiviert werden sollen, Sie können das also für jede Mappe individuell entscheiden).



Das folgende Kapitel erläutert zunächst den grundlegenden Umgang mit Visual Basic für Anwendungen (VBA), der in Excel (und den restlichen Office-Produkten) enthaltenen Programmiersprache: das Erstellen von Modulen und Prozeduren, das Ausführen dieser Prozeduren, das Testen und die Behandlung darin auftretender Fehler.

Ich bespreche zunächst die VBA-Programmierungsumgebung, vor allem die verschiedenen Code-Ebenen (Prozeduren, Module, Projekte) und Prozedurtypen (Funktionsprozeduren, Ereignisprozeduren und Sub-Prozeduren).

Danach erfahren Sie im Detail, wie Sie Ihren Programmcode schreiben, speichern, editieren und ausführen und welche Hilfestellungen VBA dabei gibt (Quickinfo, Lesezeichen, Syntaxanalyse etc.).

Anschließend geht es vor allem um das Testen von Code, zunächst mithilfe der hamlosesten Möglichkeit, dem Direktfenster, das hervorragend dazu geeignet ist, einzelne Teile von Prozeduren auszuprobieren und fehlerhafte Ausdrücke darin zu entdecken.

Danach erläutere ich den integrierten Debugger, der für komplexere Fehleranalysen eingesetzt wird und entsprechend umfangreiche Möglichkeiten bietet, anfangen von der schrittweisen Ausführung von Prozeduren über das Einfügen von Haltepunkten bis hin zur ständigen Überwachung der Inhalte ausgewählter Variablen.

Zum Schluss zeige ich Ihnen, wie Sie Ihre VBA-Programme als Add-Ins weitergeben und auf diese Weise vor unbefugten Einblicken und Veränderungen schützen.

Danach können Sie zwar einigermaßen mit dem System selbst umgehen, praktisch mit der Oberfläche, aber noch nicht programmieren. Die dazu benötigten Sprachgrundlagen enthalten Kapitel 33, »Sprachelemente«, und Kapitel 34, »Kontrollstrukturen«.

Anschließend kennen Sie die Sprache selbst einigermaßen, müssen jedoch noch lernen, wie Sie diese Kenntnisse auf Excel-Objekte anwenden, vor allem auf Tabellen und Diagramme. Dafür sind Kapitel 35 und Kapitel 36 zuständig.

Diese beiden Kapitel sind die für Sie wichtigsten, falls Sie bereits über Basic-Kenntnisse verfügen. Beide stellen jedoch nur Grundlagenkapitel dar. Excel verfügt über zu viele Objekte, Methoden und Eigenschaften, um auch nur annähernd alle erläutern zu können. Zumindest gewinnen Sie jedoch einen Überblick und können anschließend selbstständig weiterlernen, was die Manipulation von Excel-Objekten angeht.

Das folgende Kapitel 37, »Formulare und Steuerelemente«, bildet eine Art krönenden Abschluss. Sie erfahren darin, wie Sie eigene Dialogfelder erzeugen, mit denen Sie umfangreicheren Anwendungen ein individuelles Aussehen geben können, sodass der Anwender kaum noch merkt, dass er tatsächlich mit Excel arbeitet.

Das Kapitel über Dateimanipulationen erläutert danach die verschiedenen Möglichkeiten des Dateizugriffs, den sequentiellen, den wahlfreien und den Direktzugriff.

Vielleicht wollen Sie jedoch erst einmal wissen, wozu Sie Visual Basic für Anwendungen (ab jetzt kurz VBA genannt) überhaupt brauchen:

- ➔ Definieren eigener Funktionen: Wenn Ihnen die eingebauten Funktionen (*SUMME*, *MITTELWERT* etc.) nicht ausreichen, können Sie sich mit VBA eigene Funktionen basteln und anschließend ebenso wie die eingebauten in beliebigen Ausdrücken verwenden.
- ➔ Add-Ins erstellen: Wenn Sie wollen, können Sie diese Funktionen als Add-Ins speichern. Geben Sie das Add-In an Benutzer weiter, können diese zwar die darin enthaltenen Funktionen benutzen, Ihre Programme jedoch weder einsehen noch ändern.
- ➔ Komplexe Berechnungen vereinfachen: Angenommen, in einer Tabelle werden wöchentlich die auf einer Baustelle geleisteten Arbeitsstunden erfasst. Abhängig von verschiedenen Bedingungen sollen nun die tatsächlich zu zahlenden Stunden berechnet werden, mit unterschiedlichen Zuschlägen für Überstunden, Wochenenden, Feiertage und so weiter. Die Berechnungen sind somit von sehr vielen Bedingungen abhängig.

Sie können zwar mit entsprechend vielen ineinander verschachtelten Aufrufen der *Wenn*-Funktion das Problem direkt in der Tabelle lösen, erhalten dadurch jedoch kilometerlange Formeln in Ihren Zellen, die kaum noch zu überschauen, geschweige denn anzupassen sind, sollte sich irgend etwas an den Berechnungsgrundlagen ändern.

VBA ermöglicht mit einer *Wenn-Dann-Sonst*-Blockstruktur eine weit aus übersichtlichere und elegantere Lösung des Problems.

- ➔ Automatisierte Abläufe: Sie können beispielsweise Aktienkurse automatisch aktualisieren, indem Ihr VBA-Programm den neuesten Aktienkurs von IBM aus einer Datei einliest, die Mappe mit der Aktienkurstabelle öffnet, diese Tabelle aktiviert, nach der Spalte mit der zugehörigen Überschrift IBM sucht, diese Spalte von oben nach unten durchläuft, bis die erste leere Zelle gefunden wird und den eingelesenen Kurs einträgt.
- ➔ Integrierte Anwendungen mit eigenen Menüs und Dialogfeldern, die Ihre Handschrift tragen und vor dem Anwender geschützt sind: Angenommen, Sie erstellen eine Buchführungs-Tabelle. Ist eine Ausgabe unter »Post« einzutragen, macht das der Anwender normalerweise direkt in der Tabelle, in der betreffenden Spalte. Dabei kann er sich vertun und die Ausgaben in die falsche Spalte eintragen. Er kann sogar mit Befehlen wie `FORMAT|ZELLEN...` die mühevoll aufgebaute Optik zerstören und so weiter.

VBA ermöglicht Ihnen, die Excel-Menüs durch eigene Menüs zu ersetzen, sodass dem Anwender gefährliche Befehle nicht mehr zur Verfügung stehen.

Zusätzlich entwerfen Sie eigene Dialogfelder, die von den betreffenden Befehlen geöffnet werden. Will der Anwender beispielsweise eine Ausgabe in der Spalte »Post« eintragen, wählt er den von Ihnen definierten Befehl `AUSGABEN|POST...` und trägt den Wert im zugehörigen Dialogfeld ein. OK führt dazu, dass nicht der Anwender, sondern *Ihr Programm* diesen Wert in die erste freie Zelle der Spalte »Post« einträgt, der Anwender also überhaupt nicht mit der eigentlichen Tabelle in Berührung kommt, sie daher also auch nicht ungewollt verändern kann.

Auf eines muss ich Sie ausdrücklich hinweisen: Speziell die Kapitel über die Excel-Objekte stellen eine Einführung dar. Eine vollständige Abhandlung würde ein eigenes Buch erfordern, das ebenso umfangreich wäre wie das vorliegende. Daher empfinde ich die Originalhandbücher zu Excel als unverzichtbar für jeden nicht nur gelegentlichen VBA-Programmierer!



Dennoch müssen Sie keineswegs ständig in diesen Handbüchern blättern, um benötigte Informationen zu finden. Die Online-Hilfe stellt Ihnen die gesamte Referenz zur Verfügung. Entweder suchen Sie im Register *Index*

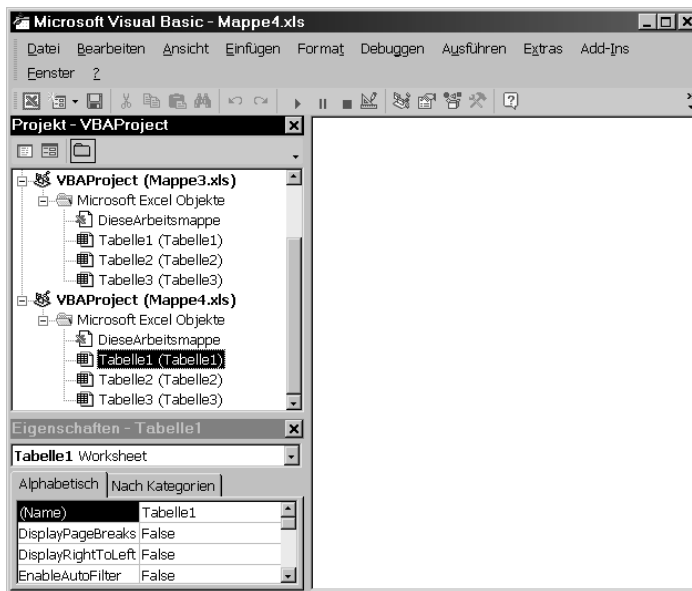
gezielt nach einem bestimmten Schlüsselwort oder aber Sie benutzen die im Register *Inhalt* enthaltenen Bücher, beispielsweise *Microsoft Excel Visual Basic Referenz*, das eine alphabetisch sortierte Übersicht aller »Methoden«, »Objekte«, »Eigenschaften« etc. enthält.

32.1 VBA-Projekte – ein Überblick

Die VBA-Programmierungsumgebung ist äußerst komplex. Im vorliegenden Kapitel will ich Ihnen daher zunächst einen Überblick darüber geben, mit welchen Objekten Sie es zu tun haben und wie Sie damit umgehen. Im folgenden Kapitel wird es dann konkret, das heißt, Sie erstellen Ihre ersten VBA-Programme.

Um ein VBA-Programm zu erstellen, benötigen Sie den Visual Basic-Editor, den Sie mit EXTRAS|MAKRO|VISUAL BASIC-EDITOR aufrufen (Abbildung 32.1).

Abbildung 32.1:
Visual Basic-Editor



Visual Basic ist im Grunde ein eigenständiges, von Excel unabhängiges Programm. Mit dem abgebildeten Excel-Symbol ganz links in der Symbolleiste können Sie jedoch jederzeit zu Excel und der darin gerade geöffneten Mappe zurückkehren bzw. durch erneute Wahl von EXTRAS|MAKRO|VISUAL BASIC-EDITOR wieder zur aktuellen Position in Visual Basic umschalten.

Noch einfacher ist das Umschalten mithilfe der Task-Leiste von Windows, in der nun sowohl »Microsoft Excel« als auch »Microsoft Visual Basic« angezeigt werden.

Projekte

Ein VBA-Programm besteht aus mehreren Ebenen. Die oberste ist die Projektebene. Ein Projekt umfasst alle Teile der zugehörigen Excel-Arbeitsmappe: alle darin enthaltenen Blätter und alle VBA-spezifischen Komponenten wie Formulare, Module und die in diesen Modulen enthaltenen Prozeduren.

Dazu müssen Sie wissen, dass VBA-Programme meist aus kurzen Programmstückchen bestehen, Prozeduren, die in Modulen enthalten sind. Ein Modul ist praktisch ein Behälter für eine beliebige Anzahl derartiger Programmstücke. Theoretisch benötigen Sie selbst für das umfangreichste denkbare Programm nur ein einziges Modul, das alle von Ihnen erstellten Prozeduren enthält.

In der Praxis verliert man dadurch jedoch leicht den Überblick und verwendet daher mehrere Module, um die erstellten Prozeduren nach Aufgabengebieten getrennt aufzubewahren.

Beispielsweise ein Modul namens »Datenmanipulation« für alle Prozeduren, die die Aufgabe besitzen, in Tabellen enthaltene Daten zu suchen oder zu verändern, oder ein Modul »Sortieren«, falls Sie eine Vielzahl an Prozeduren schreiben, die verschiedenste Sortieraufgaben besitzen, und sicherlich auch ein Modul »Allgemein« oder »Sonstiges« für Prozeduren, die Sie in keine andere Schublade einordnen können.

Das Abbildung 32.1 entstand, nachdem in Excel zwei Mappen angelegt und danach Visual Basic aufgerufen wurde. Entsprechend werden im Projekt-Explorer zwei Projekte angezeigt, eines namens »VBAProject (Mappe3)«, das zu »Mappe3« gehört, und ein weiteres zu »Mappe4« gehörendes Projekt namens »VBAProject (Mappe4)«.

Beide Projekte können vollkommen unabhängig voneinander bearbeitet werden. Selektieren Sie beispielsweise im Projektfenster irgendein Objekt des zu »Mappe3« gehörenden Projekts und wählen Sie danach EINFÜGEN|MODUL, so wird in diesem Projekt ein neues Modul angelegt und in einem eigenen Modulfenster angezeigt. Selektieren Sie dagegen zuvor ein Objekt von »Mappe4«, so wird entsprechend in diesem Projekt ein neues Modul erstellt.

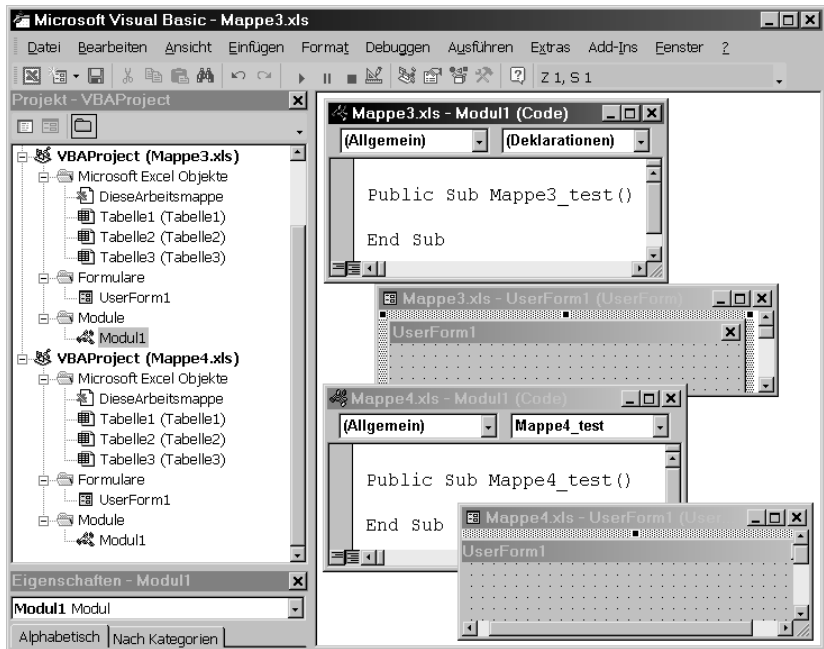
Um in ein Modul eine neue Prozedur einzufügen, doppelklicken Sie im Projekt-Explorer auf das betreffende Modul, um es zu öffnen, und wählen bei geöffnetem Modul nun EINFÜGEN|PROZEDUR...

Sie werden nach dem Namen gefragt, den die Prozedur erhalten soll, diese wird im Modulfenster angelegt und Sie können nun in dieser Prozedur Programmzeilen eintippen.

Analog dazu wird mit EINFÜGEN|USERFORM im zu »Mappe3« gehörenden Projekt ein Formular angelegt, wenn gerade ein Objekt dieses Projekts aktiv ist, bzw. im zu »Mappe4« gehörenden Projekt, wenn ein Objekt dieses Projekts aktiv ist.

Nehmen wir an, Sie erstellen für zwei Excel-Mappen namens MAPPE3.XLS und MAPPE4.XLS jeweils eine neues Formular und ein neues Modul und fügen in beide Module je eine Prozedur ein (Abbildung 32.2).

Abbildung 32.2:
Zwei Projekte,
Module und Proze-
duren



Dann besteht das erste Projekt aus

- ➔ den in MAPPE3.XLS enthaltenen normalen Excel-Objekten, das heißt aus der Arbeitsmappe selbst und den darin enthaltenen drei Tabellen
- ➔ und aus den zugehörigen VBA-Objekten: dem Formular »Userform1« und dem Modul »Modul1« und der darin enthaltenen Prozedur, die im Beispiel »Mappe3_test« heißt.

Analog dazu besteht das zweite Projekt aus

- ➔ den in MAPPE4.XLS enthaltenen Excel-Objekten, also aus der Arbeitsmappe selbst und den darin enthaltenen Tabellen
- ➔ und erneut aus den zugehörigen VBA-Objekten: dem gleichnamigen Formular »Userform1«, dem gleichnamigen Modul »Modul1« und der darin enthaltenen Prozedur, die im Beispiel »Mappe4_test« heißt.

Wie Sie sehen, können verschiedene Projekte gleichnamige Objekte enthalten. Innerhalb eines Projekts ist das jedoch nicht zulässig, kann es also keine zwei Module mit dem Namen »Modul1« geben! Und in einem Modul wiederum darf es keine gleichnamigen Prozeduren geben!

Um ein Objekt anzuzeigen, können Sie wie gewohnt das Menü FENSTER benutzen, das wie üblich separate Einträge für jedes geöffnete Fenster enthält; Anklicken aktiviert das betreffende Fenster.

Objekte anzeigen

Haben Sie ein Fenster geschlossen, müssen Sie stattdessen den Projekt-Explorer benutzen. Wie die vorherigen Abbildungen zeigten, funktioniert er prinzipiell wie der Windows-Explorer, listet also alle Objekte und Unterobjekte in Baumform auf. Doppelklicken Sie darin auf ein Objekt, so wird sein Fenster zum aktiven Fenster bzw. zuvor geöffnet, wenn es momentan geschlossen ist.

Das Löschen eines Objekts erfolgt mit DATEI|ENTFERNEN VON [OBJEKT-NAME]... . Dieser Befehl entfernt immer das momentan im Projekt-Explorer selektierte VBA-Objekt, beispielsweise ein Formular oder Modul (Excel-Objekte wie Tabellen können mit diesem Befehl nicht entfernt werden).

*Objekte entfernen,
ex- und impor-
tieren*

Vor dem endgültigen Entfernen werden Sie noch gefragt, ob Sie das Objekt exportieren wollen. Falls ja, wird es unter einem frei wählbaren Namen in einer separaten Datei gespeichert (mit spezifischen Endungen je nach Objekttyp, beispielsweise FRM für Formulare oder BAS für Module).

Das Gleiche können Sie mit DATEI|DATEI EXPORTIEREN... auch erreichen, ohne das Objekt aus dem aktuellen Projekt zu entfernen. In beiden Fällen kann ein exportiertes Objekt jederzeit mit DATEI|DATEI IMPORTIEREN... in das aktuelle Projekt importiert werden.

Wo werden die beiden Projekte aufbewahrt? Sie werden in der zugehörigen Excel-Datei aufbewahrt, also in MAPPE3.XLS bzw. in MAPPE4.XLS. Um die Speicherung müssen Sie sich innerhalb von VBA theoretisch überhaupt nicht kümmern. Die Projekte werden gespeichert, wenn Sie zu Excel zurückkehren und die zugehörigen Arbeitsmappen schließen.

*Projekte
speichern*

Excel fragt dann wie gewohnt, ob die Änderungen gespeichert werden sollen. Bejahen Sie diese Frage, werden in der Datei außer den Excel-Tabellen zusätzlich alle mit VBA erstellten Objekte gespeichert, Ihre Formulare, Module und Prozeduren.

Wollen Sie nicht bis dahin warten, können Sie auch zwischendurch zu Excel zurückkehren und Ihre Mappe wie gewohnt speichern. Allerdings ist dieser Programmwechsel überflüssig, da auch VBA den Befehl DATEI|SPEICHERN bietet. Sind mehrere Mappen und damit mehrere Projekte geöffnet, speichert dieser Befehl immer das aktive Projekt und lautet entsprechend MAPPE3.XLS SPEICHERN oder MAPPE4.XLS SPEICHERN.

Aktiv ist immer jenes Projekt, zu dem das Objekt gehört, mit dem Sie gerade arbeiten. In Abbildung 32.2 befindet sich der Cursor im Modulfenster »Mappe3.xls – Modul1«, also ist das Objekt »Modul1« aktiv und wird im Projektfenster hervorgehoben. Da dieses Modul zum Projekt »Mappe3.xls« gehört, ist gleichzeitig dieses Projekt aktiv und DATEI| MAPPE3.XLS SPEICHERN würde nun entsprechend dieses Projekt und somit MAPPE3.XLS speichern.



Sind Sie der Ansicht, ihn nicht zu benötigen, können Sie das Fenster des Projekt-Explorers schließen. Benötigen Sie ihn später doch, wählen Sie einfach ANSICHT|PROJEKT-EXPLORER bzw. klicken auf das zugehörige Symbol.



Unabhängig davon können Sie das unterhalb des Explorers sichtbare Eigenschaftsfenster jederzeit schließen und bei Bedarf mit ANSICHT|EIGENSCHAFTENFENSTER bzw. dem abgebildeten Symbol wieder einblenden.



Beide Fenster verhalten sich ähnlich wie Befehls- oder Symbolleisten: Doppelklicken Sie auf den Fenstertitel, werden sie zu unverankerten Fenstern. Verschieben Sie die Fenster und kommen Sie dabei an einen der Ränder des VBA-Fensters, will sich das betreffende Fenster dort verankern.

Jedes Objekt eines VBA-Projekts besitzt zumindest eine Eigenschaft, (*Name*), den Namen des Objekts, beispielsweise eines Moduls oder eines Formulars. Im Eigenschaftsfenster werden diese Objekteigenschaften angezeigt und können darin verändert werden.

In diesem Fenster werden immer die Eigenschaften des momentan im Projekt-Explorers selektierten Objekts angezeigt. Enthält es selbst mehrere Objekte, praktisch Unterobjekte, kann das Sie interessierende im geschlossenen Listenfeld selektiert werden.

Die linke Spalte des Eigenschaftsfensters zeigt alle Objekteigenschaften an und zwar abhängig vom gerade aktivierten Register entweder *Alphabetisch* oder *Nach Kategorien* sortiert. Die rechte enthält die aktuellen Einstellungen dieser Eigenschaften und ermöglicht gleichzeitig die Änderung dieser Einstellungen.

Beispielsweise definiert die Eigenschaft (*Name*) den Namen des betreffenden Objekts. Würden Sie in Abbildung 32.2 im Explorer das Modul »Modul1« selektieren, stünde im Eigenschaftsfenster als Eigenschaft (*Name*) dieses Objekts sein Name »Modul1«. Ändern Sie diesen Namen in »MyModul«, ändern Sie damit den Namen des Moduls.

Modultypen

Ihre Programme bestehen aus einzelnen Programmzeilen, die sich in Modulen befinden. Ein Projekt kann beliebig viele Module enthalten. Sie können zwei Arten von Modulen erzeugen: Der Befehl EINFÜGEN|KLASSENMODUL

erzeugt ein Klassenmodul. Darin enthaltene Prozeduren definieren neue Objekte und deren Eigenschaften und sind vorläufig uninteressant für uns.

Normale Module werden im Projekt-Explorer wie gezeigt als eigenständige Einträge angezeigt und können per Doppelklick auf den betreffenden Eintrag geöffnet werden, um die darin enthaltenen Programmzeilen zu bearbeiten.

Allerdings betrifft das nur Klassenmodule oder allgemeine Module, die Sie mit EINFÜGEN|KLASSENMODUL oder EINFÜGEN|MODUL erstellen. Zusätzlich gibt es verschiedene Objekte, zu denen bereits ein Klassenmodul gehört, das jedoch nicht im Projekt-Explorer angezeigt wird.

Das betrifft die Arbeitsmappe, alle darin enthaltenen Tabellen und alle von Ihnen erstellten Formulare. Zu jedem dieser Objekte gehört ein eigenes, an das betreffende Objekt gebundenes Klassenmodul, das so genannte Ereignisprozeduren enthält.

Um das zu einem solchen Objekt gehörende Klassenmodul anzuzeigen, doppelklicken Sie im Projekt-Explorer auf das betreffende Objekt, beispielsweise auf die Arbeitsmappe oder auf eine Tabelle wie »Tabelle1«. Daraufhin wird das zugehörige Klassenmodul in einem eigenen Fenster angezeigt (Abbildung 32.3).

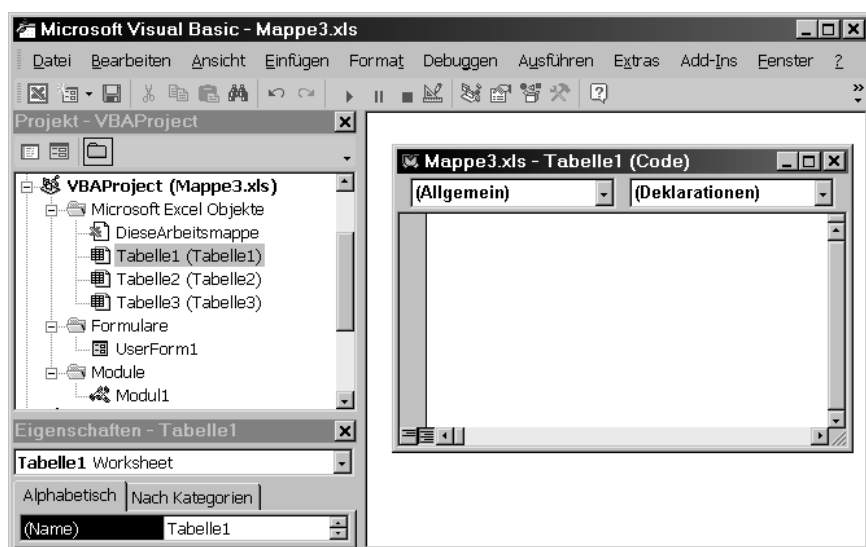


Abbildung 32.3:
Modul der Tabelle
»Tabelle1«

Allerdings gibt es eine Ausnahme: Doppelklicken Sie auf ein Formular, wird nicht das dahinter steckende Klassenmodul angezeigt, sondern das Formular selbst. Im Gegensatz dazu zeigt das abgebildete Symbol *immer* das Modul des gerade selektierten Objekts an.

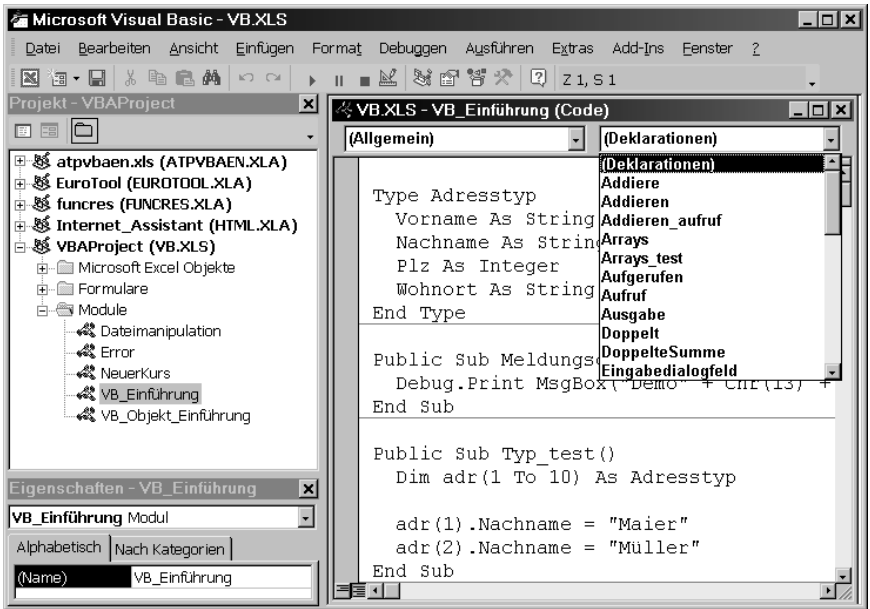




Dieses Symbol zeigt übrigens genau umgekehrt immer das Objekt selbst an und nicht den dahinter steckenden Code. Ist im Explorer die Tabelle »Tabelle3« selektiert, wird entsprechend zu Excel zurückgekehrt und darin »Tabelle3« angezeigt. Ist ein Formular selektiert, wird entsprechend dieses Formular angezeigt.

Um zu sehen, was ein Modul enthält, sollten Sie alle momentan geöffneten Mappen schließen, danach die Mappe VB.XLS der Begleit-CD öffnen, mit EXTRAS|MAKRO|VISUAL BASIC-EDITOR VBA aufrufen und per Doppelklick auf den Eintrag »VB_Einführung« im Explorer dieses Modul öffnen, das eine ganze Menge an Prozeduren enthält (Abbildung 32.4).

Abbildung 32.4:
Modulinhalt



*Prozedur-
Listenfeld*

Ganz oben, im Bereich bis zur ersten Trennlinie, befindet sich der Deklarationsabschnitt, in dem sich so genannte Deklarationen befinden.

Jedes Modul beginnt mit einem solchen Abschnitt, dem dann die einzelnen Prozeduren folgen, die jeweils durch Trennlinien voneinander unterschieden werden.

Mit dem Befehl EINFÜGEN|PROZEDUR... können Sie jederzeit weitere Prozeduren in das Modul einfügen, in dem sich gerade der Cursor befindet.

Sie können im Modul wie in einem Text umherblättern. Um zu einer bestimmten Prozedur zu gelangen, gibt es jedoch eine einfachere Möglichkeit: Das rechte Listenfeld des Fensters, das Prozedur-Listenfeld, enthält den

Eintrag (*Deklarationen*), der für den Deklarationsabschnitt steht, und darunter werden alle Prozeduren des Moduls aufgelistet.

Klicken Sie einen Eintrag an, wird automatisch zum Deklarationsabschnitt bzw. zur betreffenden Prozedur geblättert, die anschließend ganz oben im Fenster erscheint.

Besonders nützlich ist diese Auswahlmöglichkeit, wenn Sie auf das abgebildete Symbol klicken, das sich in der unteren linken Ecke jedes Modulfensters befindet. Es aktiviert die Prozeduransicht, in der immer nur eine einzige Prozedur (oder der Deklarationsabschnitt) angezeigt wird (Abbildung 32.5).

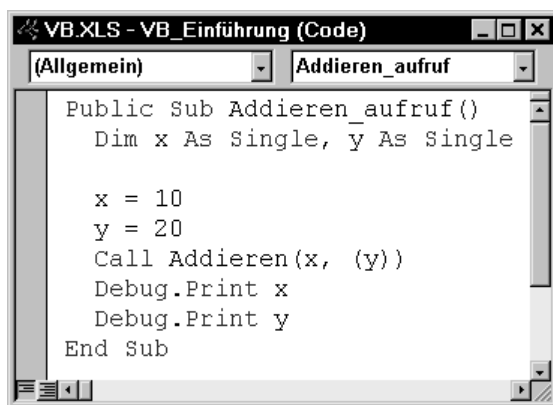


Abbildung 32.5:
Prozeduransicht

Mit diesem Symbol können Sie jederzeit wieder die volle Modulansicht einschalten.



Das linke Objekt-Listenfeld benötigen Sie nur in Klassenmodulen. Darin sind alle Prozeduren, die zu einem bestimmten Objekt gehören, zu Gruppen zusammengefasst. Um eine dieser Prozeduren zu editieren, müssen Sie zunächst im Objekt-Listenfeld statt »(Allgemein)« das Sie interessierende Objekt auswählen.

Objekt-Listenfeld

Anschließend werden im Ereignis-Listenfeld rechts daneben alle Ereignisse aufgelistet, die zu diesem Objekt gehören. Ist an eines dieser Ereignisse eine Prozedur gebunden, wird der Eintrag fett hervorgehoben. Anklicken öffnet die Prozedur, die nun editiert werden kann.

Ereignis-
Listenfeld

Klicken Sie einen nicht fett hervorgehobenen Eintrag an, wird stattdessen ein neuer Prozedurrumpf vorgegeben, der Ihnen das Erstellen einer an das betreffende Ereignis gebundenen Prozedur ermöglicht.

Prozedurtypen

Abgesehen von den so genannten Eigenschaftsprozeduren, die erst später besprochen werden, gibt es drei Typen von Prozeduren:

- ➔ Funktionsprozeduren sind selbstdefinierte Funktionen mit frei wählbarem Funktionsnamen, die wie die eingebauten Excel-Funktionen einen Funktionswert übergeben und genauso werden. Sie können beispielsweise eine selbstdefinierte Funktion namens *USt* schreiben und als Steuerelementinhalt eines Textfelds anschließend einen Ausdruck wie $=2 * USt$ verwenden, um diese Funktion aufzurufen.
- ➔ Ereignisprozeduren übergeben keinen Funktionswert, sondern führen einfach die von Ihnen definierten Anweisungen aus. Sie sind wie Makros an ein Ereignis eines Steuerelements gebunden. Tritt das Ereignis ein, werden sie aufgerufen und die darin enthaltenen Anweisungen ausgeführt.

Der Name einer Ereignisprozedur ist nicht frei wählbar, sondern hängt vom Ereignis ab, an das die betreffende Prozedur gebunden ist, und besitzt die Form

`Steuerelementname_Ereignisname`

Beispielsweise besitzt eine Ereignisprozedur, die an das Ereignis *Selection_Change* einer Tabelle gebunden ist und das eintritt, wenn eine andere Zelle oder ein anderer Bereich selektiert wird, den Namen

`Worksheet_SelectionChange`

- ➔ Sub-Prozeduren übergeben ebenfalls keinen Funktionswert und sind nicht an Ereignisse gebunden. Sie werden meist von Funktionen oder Ereignisprozeduren aus aufgerufen. Nach dem Aufruf werden die im betreffenden Block enthaltenen Anweisungen ausgeführt, danach wird die Ausführung der übergeordneten Funktion oder Ereignisprozedur fortgesetzt.

Anweisungen

Prozeduren enthalten Anweisungen. Anweisungen besitzen entweder wie *Dim* deklarativen Charakter und beschreiben einen Sachverhalt oder sind wie *Do...While* ausführbar und bewirken eine sofortige Aktion, schließen beispielsweise ein Fenster oder tragen einen Wert in eine Tabelle ein.

Ausführbare Anweisungen müssen sich innerhalb einer Prozedur befinden (auf Prozedurebene), nicht ausführbare Anweisungen können sich auch außerhalb befinden (auf Modulebene, im Abschnitt (*Deklarationen*)).

**Objekte,
Methoden und
Eigenschaften**

Prozeduren und Anweisungen kennt praktisch jede Programmiersprache. VBA ist jedoch zusätzlich objektorientiert: Es gibt Objekte und spezielle Sprachelemente, die den Umgang mit diesen Objekten vereinfachen sollen.

Ausgangspunkt ist die Überlegung, dass ein Objekt bestimmte Eigenschaften besitzt und mit bestimmten Methoden manipuliert werden kann:

- ➔ **Objekte:** Praktisch alles, womit Sie es in Excel zu tun haben, ist ein Objekt: ein Tabellenblatt, ein Zellbereich, ein Textfeld, ein Menü, eine Symbolleiste oder eine Diagrammlegende.

Es gibt Objekttypen oder Klassen, beispielsweise den Typ *Worksheets* (=Tabellenblätter), und einzelne Objekte des betreffenden Typs, beispielsweise ein ganz bestimmtes Tabellenblatt.

- ➔ **Eigenschaften:** Jedes Objekt besitzt bestimmte Eigenschaften, eine Legende unter anderem die Eigenschaften *Height*, *Font* etc.
- ➔ **Methoden:** Methoden sind Prozeduren, die auf bestimmte Objekte spezialisiert sind und damit etwas anstellen.

Beispielsweise übergibt die Methode *Value* den Inhalt eines Textfeldes oder einer Zelle. Die Methode *LegendEntries* wiederum übergibt eine einzelne Legendenbeschriftung oder alle Beschriftungen der Legende.

Beides sind somit Funktionen, die einen Funktionswert übergeben, im Gegensatz zu normalen Funktionen allerdings nur mit ganz bestimmten Objekten umgehen können.

32.1.1 Funktionsprozeduren

Bei der Eingabe Ihrer ersten Prozeduren wird Seltsames passieren: Immer wieder wird Excel an der aktuellen Cursorposition ein Fensterchen oder eine Liste mit höchst merkwürdigem Inhalt öffnen.



Kümmern Sie sich nicht darum, tippen Sie einfach weiter. Es handelt sich um verschiedene Automatik-Hilfen, die mit dem Befehl EXTRAS|OPTIONEN... ein- oder ausgeschaltet werden können und eben standardmäßig aktiviert sind. Ich erläutere diese Hilfestellungen in Kapitel 32.2.1, »Bearbeitungsmöglichkeiten«.



Die auf den folgenden Seiten erstellten Prozeduren befinden sich in der Mappe VB.XLS, genauer im Modul »VB_Einführung«, das somit auch die nun erläuterte Funktionsprozedur USt enthält.



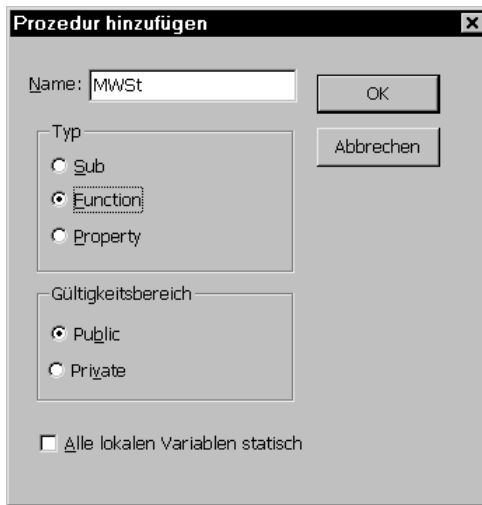
Ein Modul darf jedoch keine gleichnamigen Prozeduren enthalten. Um beim Nachvollziehen der folgenden Schritte nicht versehentlich in diesem Modul eine weitere Prozedur namens *USt* zu erstellen, empfehle ich Ihnen folgende Vorgehensweise:

- ➔ Schließen Sie alle momentan geöffneten Arbeitsmappen.
- ➔ Legen Sie eine neue Mappe an und wählen Sie EXTRAS|MAKRO|VISUAL BASIC-EDITOR.
- ➔ Erstellen Sie mit EINFÜGEN|MODUL ein neues Modul.

Neue Prozedur

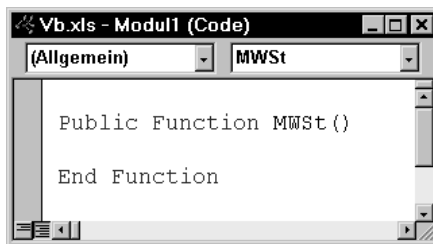
EINFÜGEN|PROZEDUR... fügt in das momentan geöffnete Modulfenster eine neue Prozedur ein (Abbildung 32.6).

Abbildung 32.6:
Eigendefinierte
Funktion



Tippen Sie den Funktionsnamen *USt* ein, selektieren Sie unter *Typ* »Function« und unter *Gültigkeitsbereich* »Public« und wählen Sie *OK* (Abbildung 32.7).

Abbildung 32.7:
Vb.xls,
»VB_Einführung«,
Prozedur USt



Excel fügt daraufhin einen Prozedurrumpf ein. Er enthält den Prozedurkopf, der aus den Schlüsselwörtern *Public* und *Function* besteht, gefolgt von dem Funktionsnamen und einem leeren Klammerpaar. Den Prozedurfuß bildet die Anweisung *End Function*.

Statt mit EINFÜGEN|PROZEDUR... können Sie eine Prozedur einfügen, indem Sie einfach in irgendeiner leeren Zeile (die sich nicht in einem bestehenden Prozedurblock befinden darf) die Anweisung

```
Public Function USt()
```

eintippen und am Ende der Zeile  drücken.

Füllen Sie die vorgegebene Schablone nun bitte folgendermaßen aus (Abbildung 32.8):

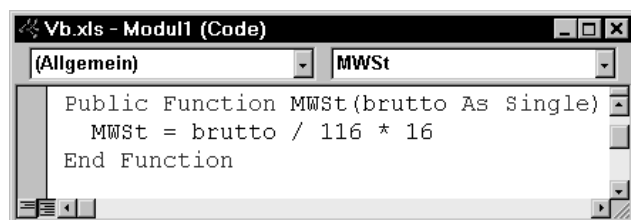


Abbildung 32.8:
Vb.xls,
»VB_Einführung«,
Prozedur USt

Kümmern Sie sich zunächst nicht um die kleinen Fenster, die während der Eingabe gelegentlich eingeblendet werden, sondern tippen Sie einfach weiter.

Die Leerzeichen vor und nach den Operatoren »=«, »/« und »*« müssen Sie nicht eintippen. Nach dem Verlassen einer Zeile mit dem Cursor führt Excel automatisch eine Syntaxanalyse durch, mit folgenden Resultaten:

- ➔ Vor und nach Operatoren werden Leerzeichen eingefügt.
- ➔ Für alle Schlüsselwörter wird Groß-/Kleinschreibung verwendet.
- ➔ Syntaktisch fehlerhafte Anweisungen führen zu einer entsprechenden Fehlermeldung.

Tippen Sie

```
if a=1 then b=2
```

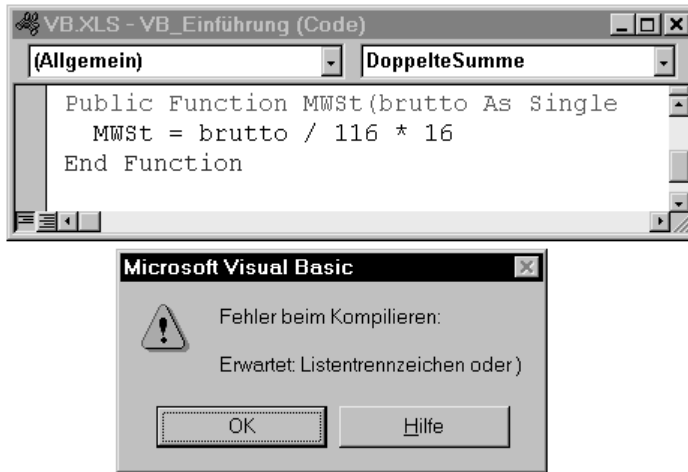
so macht Excel daraus

```
If a = 1 Then b = 2
```

Die Syntaxanalyse bewirkt zusätzlich eine automatische Groß-/Kleinschreibung aller Schlüsselwörter. Würden Sie ein Schlüsselwort wie *function* eingeben, wandelte VBA dieses Schlüsselwort beim Verlassen der Zeile in *Function* um.

Ebenso automatisch würde eine Fehlermeldung erscheinen, wenn die eingegebene Codezeile offensichtlich fehlerhaft wäre (Abbildung 32.9).

Abbildung 32.9:
Syntaxanalyse und
Fehlermeldung



VBA erkennt, dass die schließende Klammer fehlt und gibt eine entsprechende Meldung aus. Haben Sie die Meldung akzeptiert, können Sie den Fehler korrigieren oder aber die rot hervorgehobene Zeile unverändert lassen, ohne dass die Meldung erneut erscheint.

Sie erscheint erst dann wieder, wenn Sie diese Zeile später editieren und beim Verlassen der Zeile daraufhin erneut die Syntaxanalyse durchgeführt wird.

VBA-Optionen

Die Syntaxanalyse ist an sich außerordentlich nützlich. Empfinden Sie sie dennoch als störend, können Sie das Kontrollkästchen *Automatische Syntaxüberprüfung* im Register *Editor* des Befehls **EXTRAS|OPTIONEN...** deaktivieren.

Die Syntaxprüfung findet dann zwar immer noch statt und die fehlerhafte Zeile wird farbig hervorgehoben; allerdings erscheint keine Fehlermeldung mehr.

Editieren

Zum Editieren eines Textes stehen Ihnen die üblichen Möglichkeiten zur Verfügung: die Cursortasten, die Taste (←), (Entf), (Einf) (Umschalten zwischen Einfüge- und Überschreibmodus) und das Markieren von Buchstaben, Wörtern und kompletten Zeilen durch Überstreichen mit der Maus bei gedrückter Maustaste oder Bewegen des Cursors bei gedrückter (↑)-Taste.

Entsprechend der Abbildung sollten Sie alle Anweisungen innerhalb der Prozedur ein wenig einrücken, indem Sie am Anfang der betreffenden Zeile einmal (Tab) drücken. Die Option *Tab-Schrittweite* des Registers *Editor* des Befehls **EXTRAS|OPTIONEN...** legt fest, wie weit die Einrückung ausfällt.

Später noch sehr nützlich ist die Möglichkeit, mehrere Zeilen zu markieren und mit BEARBEITEN|EINZUG VERGRÖßERN bzw. durch Drücken der -Taste gemeinsam einzurücken bzw. mit BEARBEITEN|EINZUG VERKLEINERN oder  +  wieder nach links zu verschieben.



Die soeben erstellte Funktionsprozedur besitzt den Namen *USt*. Ihre Aufgabe: Sie soll nach dem Aufruf mit einem Ausdruck wie *USt(50)* die im übergebenen Wert, einem Bruttobetrag, enthaltene Umsatzsteuer ausrechnen.

Das Wort »brutto« in der Klammer ist eine Variable, praktisch ein Stellvertreter für den beim Aufruf der Funktion zu übergebenden Betrag. Der Ausdruck $Ust = brutto / 116 * 16$ teilt diesen übergebenen Bruttobetrag durch 116 und multipliziert das Ergebnis mit 16, um die darin enthaltene Umsatzsteuer zu ermitteln.

Allgemein: Für jedes einer Prozedur zu übergebende Argument verwenden Sie als Stellvertreter eine Variable, einen frei wählbaren Namen im Prozedurkopf:



Function Funktionsname (Argument1; Argument2; ...; ArgumentN)

Diesen Stellvertreter, diese Variable, können Sie in den folgenden Berechnungen wie die Zahl selbst behandeln, die sie vertritt.

Den zu übergebenden Funktionswert definieren Sie innerhalb der Funktion mit einem Ausdruck der Art

Funktionsname = Ausdruck

Dabei ist *Funktionsname* der Name der Funktion und *Ausdruck* ein beliebiger Ausdruck, dessen Resultat als Funktionswert übergeben werden soll.

Die selbstdefinierte Funktion können Sie wie jede integrierte Excel-Funktion anwenden (Abbildung 32.10):

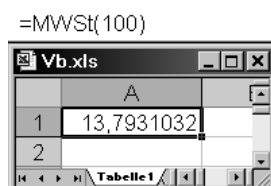


Abbildung 32.10:
Selbstdefinierte
Funktion anwenden

Die Formel

=USt(100)

ruft die Funktion *USt* auf, übergibt ihr die Zahl 100 und zeigt den von *USt* übergebenen Funktionswert 13,7931032 an.



Excel bekommt es nicht automatisch mit, wenn Sie die Funktionsdefinition ändern.

Ändern Sie *USt* derart, dass nicht die Umsatzsteuer, sondern der Nettobetrag ermittelt wird, wird in der Zelle weiterhin der alte Wert angezeigt. Erst nach einer Aktualisierung (Zelle markieren und F9 drücken) wird der neue resultierende Wert angezeigt.

Vor der Anwendung in einem Ausdruck muss eine Prozedur kompiliert werden. VBA führt diesen Vorgang automatisch aus, wenn es in einem Ausdruck auf eine noch nicht kompilierte Prozedur trifft.

Beim Kompilieren einer Prozedur werden häufig darin noch enthaltene Fehler entdeckt. Eine fehlerhafte Prozedur führt daher oft erst dann zu einer entsprechenden Fehlermeldung, wenn diese Prozedur zum ersten Mal benutzt wird.

Um diese zeitlich verlagerte Fehlerprüfung zu vermeiden, sollten Sie nach umfangreichen Programmeditierungen **DEBUGGEN|KOMPIlieren von VBA-Projekt** wählen, um sofort zu erfahren, ob eine Prozedur für VBA offensichtliche Fehler enthält.



Vergessen Sie beim Aufruf von Funktionen, denen keine Argumente übergeben werden, das dann leere Klammerpaar nicht! Angenommen, USt würden keinerlei Argumente übergeben: Im Gegensatz zu einer Excel-eigenen Funktion würde der Ausdruck =USt statt korrekt =USt() zu einer Fehlermeldung in der Zelle führen, in der Sie diese Funktion anwenden.

32.1.2 Sub-Prozeduren

Funktionen und Ereignisprozeduren können mit der Anweisung

Call Prozedurname

selbst wieder Prozeduren aufrufen. Um eine solche Sub-Prozedur zu erstellen, wählen Sie wieder **EINFÜGEN|PROZEDUR...**, selektieren im zugehörigen Dialogfeld jedoch nicht *Function*, sondern *Sub*.

VBA gibt erneut einen Prozedurrumpf vor, der diesmal jedoch statt *Function* das Schlüsselwort *Sub* enthält. Sie können erneut beliebige auszuführende Anweisungen einfügen.

Schauen Sie sich als Beispiel die Sub-Prozedur *Doppelt* an, die sich ebenso wie *USt* im Modul »VB_Einführung« von VB.XLS befindet.

Öffnen Sie dazu bitte das Modul »VB_Einführung« von VB.XLS. Öffnen Sie danach das rechte der beiden Listenfelder, das Prozedur-Listenfeld, und klicken Sie darin auf den Eintrag »USt«, um diese Prozedur einzublenden.

Ergänzen Sie darin die Zeile *Call Doppelt(brutto)* (Abbildung 32.11).

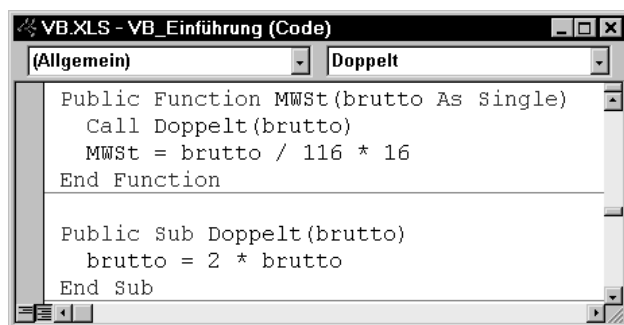


Abbildung 32.11:
Vb.xls,
»VB_Einführung«,
Prozedur »Doppelt«

Wechseln Sie danach zu Excel und geben Sie in irgendeiner Zelle einer Tabelle die Formel *=USt(100)* ein (oder aktualisieren Sie diese bereits zuvor eingegebene Formel mit **[F9]**).

Diesmal wird das Doppelte der in 100 enthaltenen Umsatzsteuer angezeigt (Abbildung 32.12).

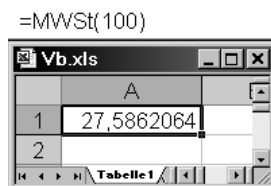


Abbildung 32.12:
Aufruf der Prozedur
»Doppelt«

Der Ablauf: Die Funktion *USt* ruft mit der *Call*-Anweisung die Prozedur *Doppelt* auf und übergibt ihr dabei in der Variablen *brutto* die im Textfeld enthaltene Zahl 100.

Die Anweisung *brutto = 2 * brutto* verdoppelt den in dieser Variablen enthaltenen Wert. Nach Rückkehr zur Funktion *USt* ermittelt diese somit die in dem nun verdoppelten Bruttobetrag enthaltene Umsatzsteuer.

Statt eine *Sub*-Prozedur mit *Call* von einer anderen Prozedur aus oder aus dem in Kürze erläuterten Direktfenster heraus aufzurufen, ist es zu Testzwecken oft einfacher, den Cursor in eine beliebige Prozedurzeile zu setzen und



AUSFÜHREN|SUB|USERFORM AUSFÜHREN zu wählen oder auf das zugehörige Symbol zu klicken; allerdings setzt das voraus, dass der betreffenden Prozedur beim Aufruf keine Argumente übergeben werden müssen.

32.1.3 Ereignisprozeduren

Zu jeder Tabelle und jedem Formular gehört ein spezielles Modul, ein so genanntes Klassenmodul. Es enthält alle Ereignisprozeduren, die mit irgendwelchen Ereignissen des betreffenden Objekts verknüpft sind.

Beispielsweise gehört auch zur Tabelle »Objekte« von VB.XLS ein solches Klassenmodul. Um es zu öffnen, doppelklicken Sie im Projekt-Explorer auf den Eintrag »Tabelle11 (Objekte)«.

Das Objekt-Listenfeld des Moduls enthält außer (*Allgemein*) zusätzlich den Eintrag *Worksheet*, der für das Objekt *Tabellenblatt* steht, praktisch für die Tabelle selbst. Mit dieser Tabelle können verschiedene Ereignisse eintreten. Beispielsweise kann in der Tabelle ein Bereich selektiert werden, wodurch das Ereignis *SelectionChange* eintritt. Die Tabelle kann aktiviert werden, wodurch das Ereignis *Activate* eintritt und so weiter.

An jedes dieser Ereignisse können Sie eine Prozedur binden, die von VBA automatisch aufgerufen wird, wenn mit dem Objekt das betreffende Ereignis eintritt. Die Namen dieser Ereignisprozeduren sind nicht frei wählbar, sondern werden nach einem festen Schema gebildet:

Objektname_Ereignis

Ereignisprozedur erstellen

Für jedes dieser Ereignisse, das mit dem Objekt *Worksheet* eintreten kann, enthält das Prozedur-Listenfeld einen eigenen Eintrag. Um eine neue Ereignisprozedur zu erstellen, selektieren Sie den gewünschten Eintrag, beispielsweise *SelectionChange* (Abbildung 32.13).

Excel gibt den Rumpf der neuen Ereignisprozedur *Worksheet_SelectionChange* vor und Sie können ihn mit den Anweisungen ausfüllen, die ausgeführt werden sollen, wenn das Ereignis *SelectionChange* mit dem Objekt *Worksheet* eintritt, wenn also irgendeine Zelle oder ein Bereich in der Tabelle »Objekte« selektiert wird.

Ist an ein Ereignis bereits eine Prozedur gebunden, wird der Eintrag fett hervorgehoben. Anklicken erstellt dann keinen neuen Prozedurrumpf, sondern öffnet die bereits vorhandene Prozedur, die nun editiert werden kann.

Beispielsweise enthält die Prozedur *Worksheet_SelectionChange*, die zur Tabelle »Objekte« der Demodatei VB.XLS gehört, bereits eine Anweisung (Abbildung 32.14).

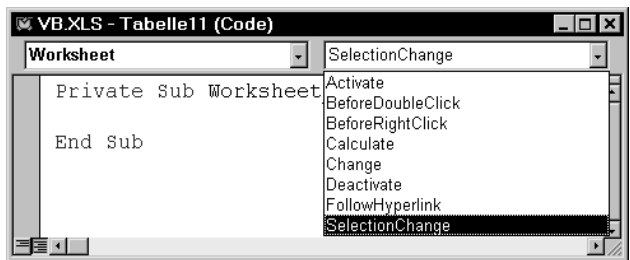


Abbildung 32.13:
Klassenmodul der
Tabelle »Objekte«

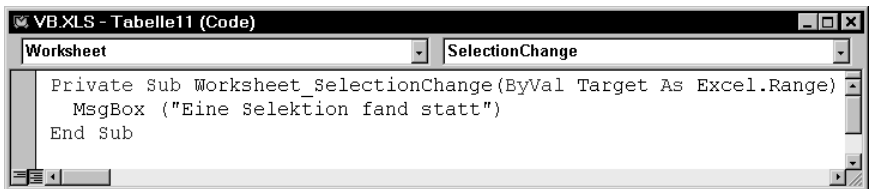


Abbildung 32.14:
Vb.xls, Klassen-
modul »Tabelle11
(Objekte)«, Prozedur
»Worksheet_
SelectionChange«

Die Prozedur wird wie erläutert aufgerufen, wenn Sie in der Tabelle »Objekte«, zu der das Modul gehört, irgendeine Zelle oder einen Bereich selektierten. Dann wird die in der Prozedur enthaltene Anweisung `MsgBox` ausgeführt, deren Aufgabe darin besteht, einen entsprechenden Hinweis auszugeben (Abbildung 32.15).

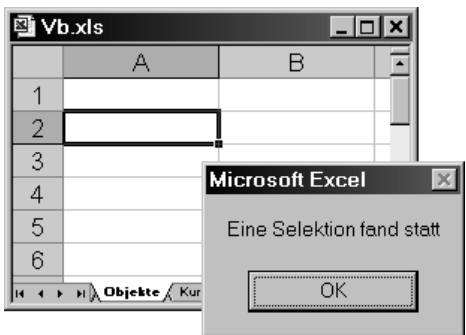


Abbildung 32.15:
Aufruf der
Ereignisprozedur

Sie dürfen keinesfalls den Namen einer Ereignisprozedur verändern! An einem Prozedurnamen wie `Worksheet_SelectionChange` erkennt VBA, welchem Objekt und welchem Ereignis die betreffende Prozedur zugeordnet ist.



32.2 Programme eingeben und testen

32.2.1 Bearbeitungsmöglichkeiten

Drag&Drop

In komplexen Programmen müssen häufig Prozeduren oder Teile von Prozeduren in andere Module verschoben oder kopiert werden. VBA vereinfacht das durch Drag&Drop: Sie öffnen die beiden Module, um die es geht, markieren in Modul A die Sie interessierenden Programmzeilen, lassen die Maustaste los, klicken auf den markierten Bereich, ziehen ihn dorthin ins Modul B, wo er eingefügt werden soll, und lassen ihn fallen.

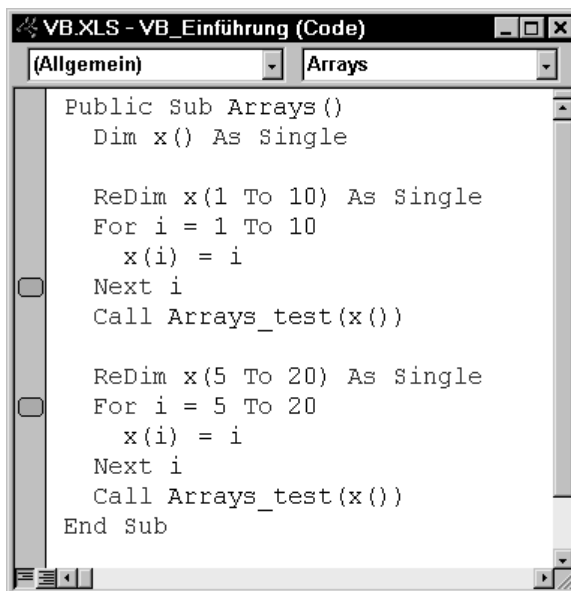
Halten Sie dabei die `[Strg]`-Taste gedrückt, werden die Programmzeilen kopiert statt verschoben, bleiben also in Modul A erhalten. Selbstverständlich können Sie Drag&Drop auch innerhalb eines Moduls zum Verschieben/Kopieren von Programmzeilen verwenden.

Lesezeichen

In umfangreichen Programmen ist es sehr nützlich, Lesezeichen setzen, löschen und sich zu gesetzten Lesezeichen bewegen zu können. BEARBEITEN|LESEZEICHEN|LESEZEICHEN SETZEN/ZURÜCKSETZEN setzt in der aktuellen Zeile ein Lesezeichen bzw. löscht ein bereits darin vorhandenes Lesezeichen.

Sie können beliebig viele Lesezeichen setzen. Die betreffenden Zeilen werden durch eine Markierung am linken Rand hervorgehoben (Abbildung 32.16).

Abbildung 32.16:
Lesezeichen



BEARBEITEN|LESEZEICHEN|NÄCHSTES LESEZEICHEN setzt den Cursor in die nächste Programmzeile, die ein Lesezeichen enthält, BEARBEITEN|LESEZEICHEN|VORHERIGES LESEZEICHEN in die vorhergehende Programmzeile und BEARBEITEN|LESEZEICHEN|ALLE LESEZEICHEN LÖSCHEN löscht alle Lesezeichen.

Bestimmt sind Ihnen inzwischen die verschiedenen automatisch erscheinenden Hilfestellungen von VBA aufgefallen: manchmal taucht während der Eingabe ein Fensterchen auf oder aber eine Auswahlliste, in der der Eintrag selektiert ist, der laut VBA am ehesten passt.

Automatik-Hilfen

Um sich damit einverstanden zu erklären und ihn in Ihr Programm einzufügen, drücken Sie **Strg** + **↵**, **↵** oder **↵**, um den Eintrag zu übernehmen und gleichzeitig eine neue Zeile zu beginnen.

Sie können auch per Doppelklick darauf einen anderen Eintrag einfügen lassen oder das Fenster bzw. die Liste mit **Esc** ohne Übernahme eines Eintrags schließen – oder einfach weitertippen und die Hilfe komplett ignorieren.

Werden Ihnen diese Automatik-Funktionen lästig, können Sie sie mit den verschiedenen Optionen im Register *Editor* des Befehls EXTRAS|OPTIONEN... einzeln deaktivieren. Bei Bedarf können Sie die Hilfestellungen dennoch mit verschiedenen Befehlen manuell in Anspruch nehmen.

Ich erläutere nun diese Befehle, wodurch gleichzeitig die verschiedenen Automaten klar werden.

Am harmlosesten ist der Befehl BEARBEITEN|QUICKINFO, der in Kurzform nützliche Informationen zu Variablen, Konstanten und Prozeduren liefert.

Tippen Sie beispielsweise *vbCancel* ein und wählen Sie diesen Befehl, erscheint ein Fensterchen (das Sie mit **Esc** schließen können) mit dem Hinweis »vbCancel = 2«, der besagt, dass diese eingebaute Konstante den Wert 2 besitzt (Abbildung 32.17).

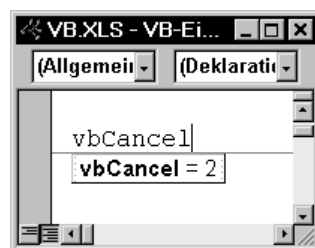
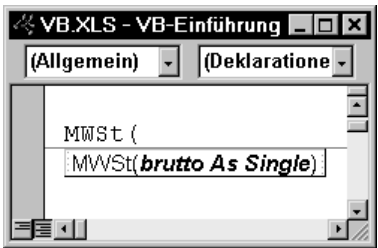


Abbildung 32.17:
Hinweis zu
Konstanten

Tippen Sie stattdessen den Aufruf einer Funktion bis vor die öffnende Klammer ein, beispielsweise »USt«, erscheint als Hinweis die Syntax dieser von Ihnen definierten Prozedur (Abbildung 32.18).

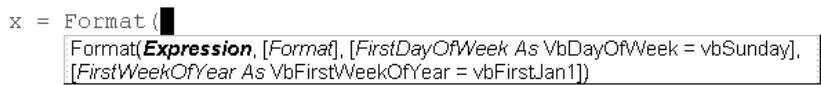
Abbildung 32.18:
Prozedursyntax



Wollen Sie statt einer selbstdefinierten eine eingebaute Prozedur wie die Funktion *Format* aufrufen, können Sie nach dem Eintippen des Prozedurnamens »Format(« inklusive öffnender Klammer BEARBEITEN|PARAMETER-INFO wählen.

Daraufhin erscheint eine ausführliche Erläuterung der Syntax dieser Prozedur, die die Festlegung der zu übergebenden Argumente deutlich erleichtert (Abbildung 32.19).

Abbildung 32.19:
Parameterinfo



Darüber hinaus können Sie verschiedenste einzutippende Objekt-, Prozedur- oder Eigenschaftsnamen aus Listen auswählen, statt sie einzutippen. Tippen Sie beispielsweise »vb« ein und wählen Sie BEARBEITEN|WORT VERVOLLSTÄNDIGEN, so erscheint eine Liste aller möglichen mit diesen beiden Zeichen beginnenden Wörter (Abbildung 32.20).

Abbildung 32.20:
Wörter ergänzen



Sie wählen den gewünschten Eintrag daraus per Doppelklick oder durch Ablaufen der Liste mit den Cursortasten und Wahl des betreffenden Eintrags mit der Leertaste aus, beispielsweise *vbCancel*, der daraufhin Ihr »vb« ersetzt; oder Sie schließen die Liste mit Esc.

Analog dazu zeigt BEARBEITEN|KONSTANTEN ANZEIGEN eine demgegenüber eingeschränkte Liste an, die nur die mit »vb« beginnenden Konstanten enthält und BEARBEITEN|EIGENSCHAFTEN/METHODEN ANZEIGEN eine Auswahlliste aller in Frage kommenden Eigenschaften und Methoden.

Mit FENSTER|TEILEN können Sie das Modulfenster in zwei voneinander unabhängige Teile mit separaten Bildlaufleisten unterteilen, um darin zwei Prozeduren des gleichen Moduls zu bearbeiten (Abbildung 32.21).

*Zwei Prozeduren
bearbeiten*

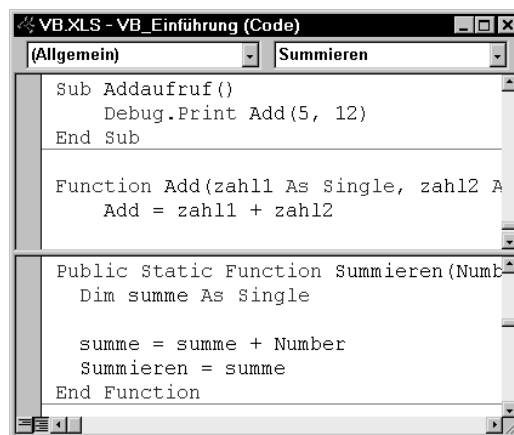


Abbildung 32.21:
Fenster teilen

Wählen Sie in den beiden Listenfeldern eine Prozedur aus, wird sie im aktiven Fensterteil angezeigt, in dem sich momentan der Cursor befindet. Der andere Fensterteil bleibt davon unbeeinflusst.

Durch Ziehen an der dünnen Teilungslinie können Sie die Position der Fensterteilung verschieben. Um die Zweiteilung aufzuheben, doppelklicken Sie auf die Teilungslinie oder ziehen sie ganz nach oben (oder wählen erneut FENSTER|TEILEN).



Umgekehrt können Sie das Fenster auch teilen, indem Sie auf das winzige Kästchen doppelklicken oder es nach unten ziehen, das sich bei nicht geteiltem Fenster unmittelbar über dem oberen Bildlaufpfeil der vertikalen Bildlaufleiste befindet.

ANSICHT|LETZTE POSITION ist nützlich, wenn Sie sich ständig andere Prozeduren anschauen, also dauernd umherspringen. VBA merkt sich jeweils, wo Sie zuvor waren, in welcher Zeile welcher Prozedur, und kehrt nach Wahl dieses Befehls dorthin zurück.

*Rückkehr zur
letzten Position*

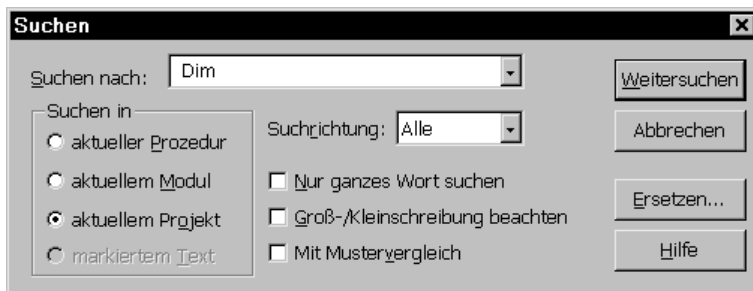
Sie können diesen Befehl mehrfach nacheinander wählen und immer weiter zurückgehen. Bei Bedarf öffnet VBA dazu sogar die Module, in denen Sie sich zuletzt befanden.

Prozedurdefinition ANSICHT|DEFINITION ermöglicht Ihnen, blitzschnell zu einer aufgerufenen Prozedur zu wechseln, um sie sich näher anzuschauen – auch wenn sie sich in einem Modul befindet, das momentan nicht geöffnet ist.

Beispielsweise zeigte Abbildung 32.11 die Prozedur *USt*, nachdem Sie darin die Zeile *Call Doppelt(brutto)* einfügten. Setzen Sie den Cursor auf irgendein Zeichen des Words *Doppelt* und wählen Sie ANSICHT|DEFINITION ($\diamond + F2$), so zeigt VBA diese Prozedur an und öffnet dazu gegebenenfalls zuvor das Modul, in dem sie sich befindet.

Suchen und ersetzen BEARBEITEN|SUCHEN... ermöglicht Ihnen, in umfangreichen Projekten einen bestimmten Textteil wie »Dim« wahlweise nur in der gerade angezeigten aktuellen Prozedur zu suchen oder in allen Prozeduren des aktuellen Moduls oder gar in allen Prozeduren des aktuellen Projekts (Abbildung 32.22).

Abbildung 32.22:
Bearbeiten
|Suchen...



Die Schaltfläche *Ersetzen...* öffnet ebenso wie der Befehl BEARBEITEN|ERSETZEN... ein Dialogfeld, in dem Sie analog dazu den gesuchten Text durch einen anderen ersetzen und dabei erneut wahlweise nur die aktuelle Prozedur, das aktuelle Modul oder alle Prozeduren aller Module durchsuchen lassen können.

Datei einfügen Mit EINFÜGEN|DATEI... können Sie bestehende Visual Basic-, QBasic-, QuickBasic- oder PDS-Programme einfügen: Dieser Befehl lädt die im zugehörigen Dialogfeld anzugebende Datei, bei der es sich um eine beliebige Textdatei oder aber um ein zuvor exportiertes Modul handeln kann.

32.2.2 Objektkatalog

Der Befehl ANSICHT|OBJEKTKATALOG ($F2$) erleichtert den Umgang mit einer Vielzahl von Prozeduren in unterschiedlichen Modulen (Abbildung 32.23).

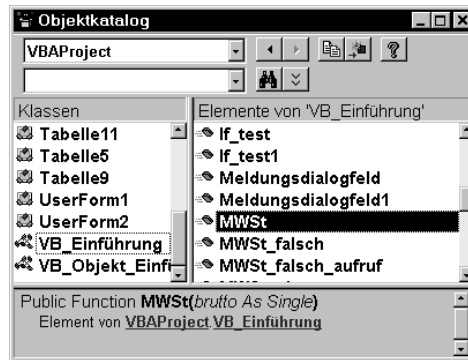


Abbildung 32.23:
Der Objektkatalog

Im rechten der beiden großen Listenfelder werden Prozeduren angezeigt (und Methoden und Eigenschaften, die letztlich auch nur Sonderformen von Prozeduren sind). Darunter finden Sie die Syntax und eine Kurzbeschreibung der betreffenden Prozedur, die Sie in Ihr Programm einfügen können, um die betreffende Prozedur anschließend aufzurufen.

Der Weg zu einer bestimmten Prozedur ist allerdings recht kompliziert. Prozeduren befinden sich immer in Standard- oder Klassenmodulen. Diese so genannten Klassen befinden sich wiederum in Excel-Dateien oder in Bibliotheken.

Das Resultat: Um die Syntax einer bestimmten Prozedur einzusehen, sind mehrere Schritte notwendig. Am Beispiel der Prozedur *USt*, die sich im Modul »VB_Einführung« der Mappe VB.XLS befindet:

- ➔ Im obersten Listenfeld selektieren Sie zunächst die Bibliothek oder Datenbank, in der sich die Prozedur befindet, im Beispiel *VBAPProject* das aktuelle Projekt.

Zusätzlich befinden sich in diesem Listenfeld unter anderem die Einträge *VBA* (Visual Basic für Anwendungen), *Excel* und *MSForms*, die für die Visual Basic-Bibliothek, die Excel-Bibliothek und die Formularobjekte-Bibliothek von Microsoft stehen – und für all die in diesen Bibliotheken enthaltenen Prozeduren.

- ➔ Haben Sie die gewünschte Bibliothek/Excel-Datei selektiert, müssen Sie im linken der beiden großen Listenfelder die Klasse dieser Bibliothek/Datenbank selektieren, in der sich die Sie interessierende Prozedur befindet, im Beispiel also das Modul »VB_Einführung«, das sich in der Excel-Datei VB.XLS befindet, dem aktuellen Projekt.
- ➔ Daraufhin werden im unteren rechten Listenfeld alle in diesem Modul enthaltenen Prozeduren aufgelistet, im Beispiel also unter anderem die Prozedur *USt*.



In diesem Listenfeld werden alle Bibliotheken und Datenbanken aufgelistet, die momentan verfügbare Prozeduren enthalten. Dabei handelt es sich zunächst um die momentan geöffnete Mappe, mit EXTRAS\VERWEISE... können Sie jedoch weitere Mappen bekanntgeben, deren Prozeduren Sie nutzen wollen und die anschließend ebenfalls hier aufgelistet werden (siehe Kapitel 33.7.3, »Bibliotheken und Prozedurbezüge«).

Doppelklicken Sie auf einen Eintrag in diesem Listenfeld, dann springen Sie im Modulfenster direkt zum betreffenden Objekt, beispielsweise zur Prozedur *USt*. Das klappt jedoch nur mit Objekten in Excel-Dateien, nicht mit Prozeduren, die in einer der Excel- oder VBA-Bibliotheken enthalten sind.

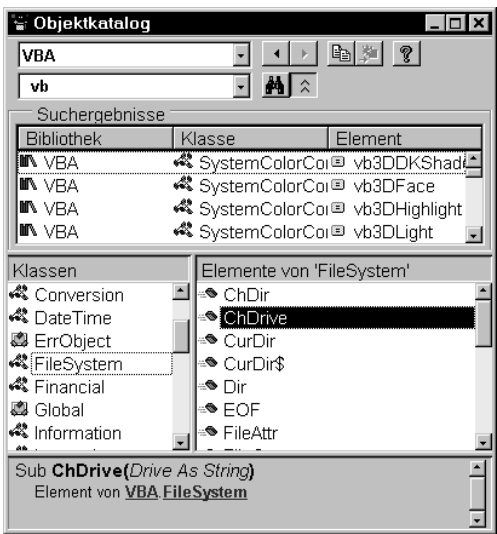
Wesentlich interessanter ist, dass Sie durch Überstreichen der Syntax bei gedrückter linker Maustaste diese markieren und danach durch Klicken auf das bekannte *Kopieren*-Symbol in die Zwischenablage befördern können.

Anschließend fügen Sie diese Syntax dort in Ihr Programm ein, wo Sie die Prozedur *USt* aufrufen wollen und müssen nur noch die in der Syntax schematisch vorgegebenen Argumente durch individuelle Argumente ersetzen, um die Prozedur *USt* aufzurufen.

Sie können sich die Suche nach einer Prozedur erleichtern, indem Sie im zweitoberen Listenfeld einen Teil des Prozedurnamens eintippen, beispielsweise »vb«, und danach  drücken oder auf das Fernglas-Symbol klicken.

Daraufhin wird das Dialogfeld um eine Liste aller Fundstellen in der darüber ausgewählten Bibliothek/Datenbank erweitert (Abbildung 32.24).

Abbildung 32.24:
Objekt suchen



Selektieren Sie entsprechend der Abbildung die VBA-Bibliothek und suchen darin nach »vb«, so werden darin alle Klassen und Prozeduren aufgelistet, deren Namen die Zeichenkette »vb« enthält, beispielsweise auch die abgebildete Prozedur *ChDrive*. Selektieren Sie sie im unteren rechten Listefeld, so wird Ihnen erklärt, dass Sie durch den Aufruf von *ChDrive* das aktuelle Laufwerk wechseln können, und wenn Sie das wollen, können Sie die Syntax wie erläutert kopieren und in Ihr Programm einfügen.

Klicken Sie auf das Fragezeichen-Symbol, so öffnet sich das Hilfefenster mit weiteren Informationen zum betreffenden Objekt.

Übrigens ist nun das Symbol rechts neben dem Fernglas hervorgehoben. Klicken Sie darauf, wird die Suchergebnisliste wieder ausgeblendet.

Die beiden nach rechts/links zeigenden Pfeile blättern übrigens wieder zum zuletzt betrachteten Objekt zurück bzw. vorwärts.

32.2.3 Das Direktfenster

Der Befehl ANSICHT|DIREKTFENSTER bzw. das zugehörige Symbol blendet das Direktfenster ein bzw. wieder aus.



Dieses Fenster ist hervorragend dazu geeignet, einzelne Teile von Prozeduren auszuprobieren und fehlerhafte Ausdrücke zu entdecken. Jede Anweisung, die Sie darin eingeben, wird unmittelbar nach Betätigung von  ausgeführt.

Am häufigsten werden Sie sicherlich die Anweisungen *Call* und vor allem *Print* einsetzen (Kurzfassung: ?). Auf *Print* folgt ein praktisch beliebig komplexer Ausdruck, der ausgewertet und dessen Ergebnis im Direktfenster ausgegeben wird, beispielsweise *Print 15*24/3* (Abbildung 32.25).

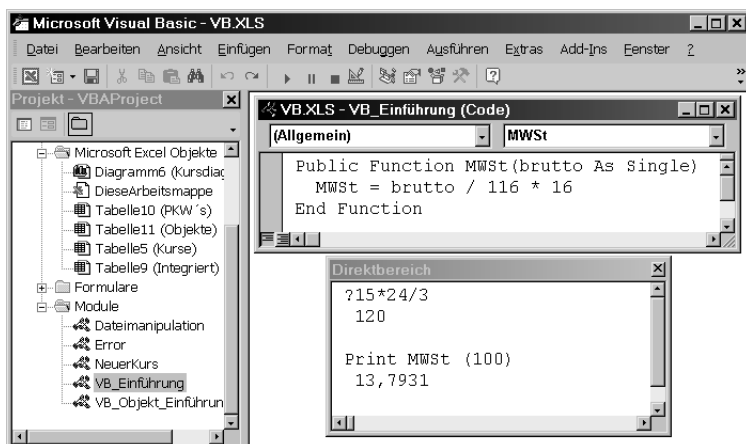


Abbildung 32.25:
Direktfenster

Das Direktfenster ist nach dem Einblenden immer im Vordergrund sichtbar und üblicherweise kein eigenständiges, sondern wie der Projekt-Explorer ein in die Umgebung integriertes Fenster.

Per Doppelklick auf die Titelleiste »Direktbereich« wird es jedoch zum eigenständigen Fenster, das Sie wie gewohnt verschieben, verkleinern oder vergrößern können. Ein erneuter Doppelklick auf die Titelleiste stellt den ursprünglichen Zustand wieder her.

Der auszuwertende Ausdruck darf auch einen Funktionsnamen enthalten. Das können Sie benutzen, um im Direktfenster eine Funktionsprozedur zu testen, beispielsweise wie in dieser Abbildung die selbstdefinierte Funktion *USt*. Der Ausdruck

```
Print USt(100)
```

(Kurzfassung: ? *USt(100)*) ruft die Funktion *USt* auf und übergibt ihr den Wert 100. Anschließend gibt die *Print*-Anweisung den von *USt* zurückübergebenen Funktionswert aus.

Um stattdessen eine Sub- oder eine Ereignisprozedur zu testen, die nur irgendwelche Aktionen ausführen, aber keinen Funktionswert übergeben kann, benutzen Sie zum Aufruf die Anweisung *Call*, gefolgt vom Prozedurnamen und eventuell noch von den zu übergebenden Argumenten:

```
Call Prozedurname(Argumente)
```

Werden der Prozedur keine Argumente übergeben, entfallen die Klammern (im Gegensatz zum Aufruf einer Funktionsprozedur, bei dem zumindest ein leeres Klammerpaar vorhanden sein muss). Ein Beispiel (Abbildung 32.26) sehen Sie hier:

Abbildung 32.26:
Sub- und Ereignis-
prozeduren testen

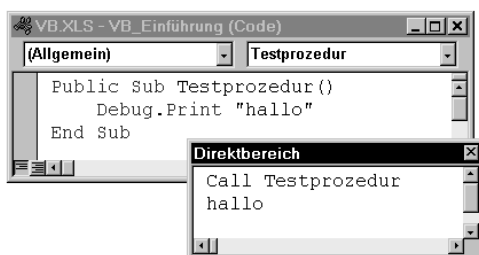


Für diese Abbildung fügte ich in ein Modul eine Sub-Prozedur namens *Test-prozedur* ein, die die Anweisung *MsgBox* (»Test«) enthält, die in einem kleinen Fenster die Meldung »Test« ausgibt. Um zu prüfen, ob die Prozedur korrekt arbeitet, öffnen Sie einfach das Direktfenster und rufen darin entsprechend der Abbildung mit *Call Testprozedur* auf.

Sollen Ihre Prozeduren zu Testzwecken Daten im Direktfenster ausgeben, müssen Sie darin den Ausdruck *Debug.Print* verwenden.

Debug.Print

Debug repräsentiert das Direktfenster und *Debug.Print* bedeutet so viel wie »Gib den nachfolgenden Ausdruck im Direktfenster aus« – das übrigens nicht im gleichen Moment geöffnet sein muss, sondern auch anschließend noch von Ihnen geöffnet werden kann (Abbildung 32.27).

**Abbildung 32.27:**Ausgaben im
Direktfenster

Sie können eine im Direktfenster eingegebene und ausgeführte Anweisung jederzeit editieren und mit  erneut ausführen lassen.



32.2.4 Fehlerverfolgung

Die abgebildeten Symbole befinden sich in der VBA-Symboleiste *Debuggen*.



In umfangreichen Programmen ist es häufig sehr schwer, die Ursache eines Fehlers zu entdecken. Als Hilfe enthält VBA einen integrierten Debugger. Debuggen (Entwanzen) ist ein historisch bedingter Ausdruck für die Entdeckung und Behebung von Programmfehlern, da angeblich in einem der ersten Rechner eine Wanze, die sich einschlich, für Fehlfunktionen sorgte.

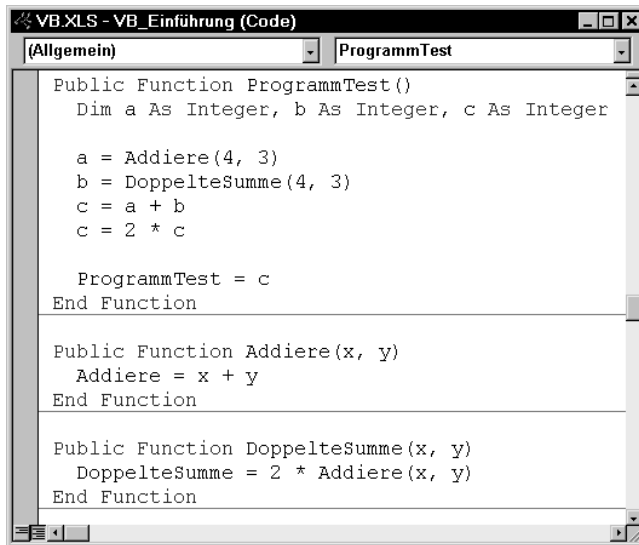
Auch bei wanzenfreien Rechnern ist der Debugger ein sehr nützliches Werkzeug. Zur Demonstration benötigen wir jedoch ein Programm, das aus mehr als nur zwei oder drei Zeilen besteht, beispielsweise die folgenden drei Prozeduren (Abbildung 32.28).

Bitte lassen Sie sich als Programmierneuling nicht dadurch frustrieren, dass Sie dieses Programm noch nicht in allen Einzelheiten verstehen. Das ist keineswegs notwendig!



Abbildung 32.28:

Vb.xls, Modul
»VB_Einführung«,
Prozeduren
»ProgrammTest«,
»Addiere« und
»DoppelteSumme«



Wichtig ist nur, zu akzeptieren, dass darin mehrere Funktionen aufgerufen werden, und vor allem: dass *a*, *b* und *c* Namen sind, hinter denen sich Zahlen verbergen, Zwischenergebnisse.

Wird die Funktion *ProgrammTest* aufgerufen, ruft sie wiederum mit der Anweisung *a = Addiere(4, 3)* die weiter unten abgebildete Funktion *Addiere* auf und übergibt ihr dabei diese beiden Zahlen.

Addiere hat die Aufgabe, die beiden beim Aufruf übergebenen Zahlen zu addieren, hier also die Zahlen 4 und 3. *Addiere* merkt sich diese Zahlen unter den Namen *x* und *y*. *x* steht praktisch stellvertretend für die erste übergebene Zahl, die 4, und *y* für die zweite Zahl, die 3.

Die in *Addiere* enthaltene Anweisung *Addiere = x + y* addiert daher 4 und 3 und übergibt zusätzlich das Resultat, im Beispiel also die Zahl 7 als Funktionswert an die aufrufende Prozedur *ProgrammTest* zurück. *ProgrammTest* merkt sich dieses Zwischenergebnis unter dem Namen *a*.

Danach ruft *ProgrammTest* mit *b = DoppelteSumme(4, 3)* die Prozedur *DoppelteSumme* auf, der die gleichen beiden Zahlen 4 und 3 übergeben werden. *DoppelteSumme* hat die Aufgabe, das Doppelte der Summe der beiden Zahlen zu errechnen.

Dazu ruft *DoppelteSumme* zunächst *Addiere* auf, um die Summe zu ermitteln. Der von *Addiere* übergebene Funktionswert 7 wird mit 2 multipliziert und das Ergebnis 14 an *ProgrammTest* als Funktionswert zurückübergeben. *ProgrammTest* merkt sich dieses zweite Zwischenergebnis unter dem Namen *b*.

Der folgende Ausdruck $c = a + b$ addiert die beiden Zwischenergebnisse 7 und 14 und gibt dem Resultat 21 den Namen c . Dieses Resultat wird wiederum mit $c = 2 * c$ verdoppelt und danach mit der letzten Anweisung das Ergebnis 42 als zurückzugebender Funktionswert definiert.

Angenommen, Sie rufen die Prozedur *ProgrammTest* auf und die Funktion übergibt nicht den von Ihnen erwarteten Wert 42, dann steckt irgendwo steckt also ein Fehler, den Sie finden müssen. Dazu können Sie den Ablauf Ihres Programms Schritt für Schritt mithilfe eines Haltepunkts nachvollziehen.

Zunächst öffnen Sie das Modul »VB_Einführung« von VB.XLS und selektieren im Prozedur-Listenfeld die Prozedur *ProgrammTest*.

Setzen Sie den Cursor irgendwo in die erste Zeile *Public Function ProgrammTest()* und wählen Sie DEBUGGEN|HALTEPUNKT EIN-/AUS (F9). Einfacher: Klicken Sie auf die graue Leiste links neben der betreffenden Zeile.



Die Zeile wird hervorgehoben. Öffnen Sie nun das Direktfenster und rufen Sie darin die Funktion mit

```
Print ProgrammTest()
```

auf oder indem Sie den Cursor in irgendeine Programmzeile setzen und DEBUGGEN|EINZELSCHRITT wählen (Abbildung 32.29).

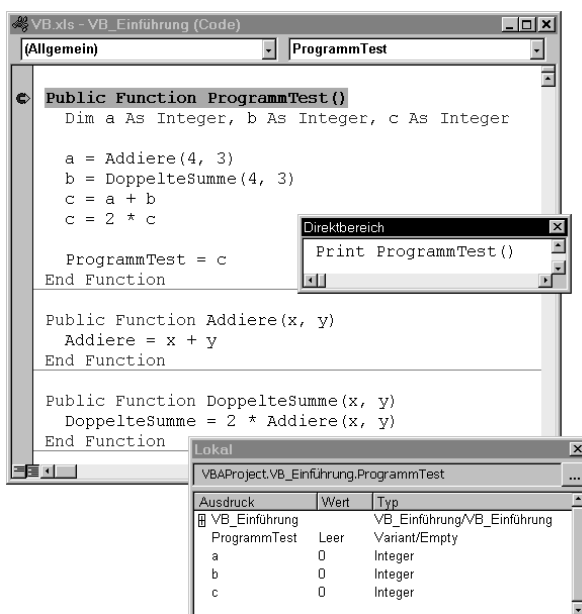


Abbildung 32.29:
Haltepunkt

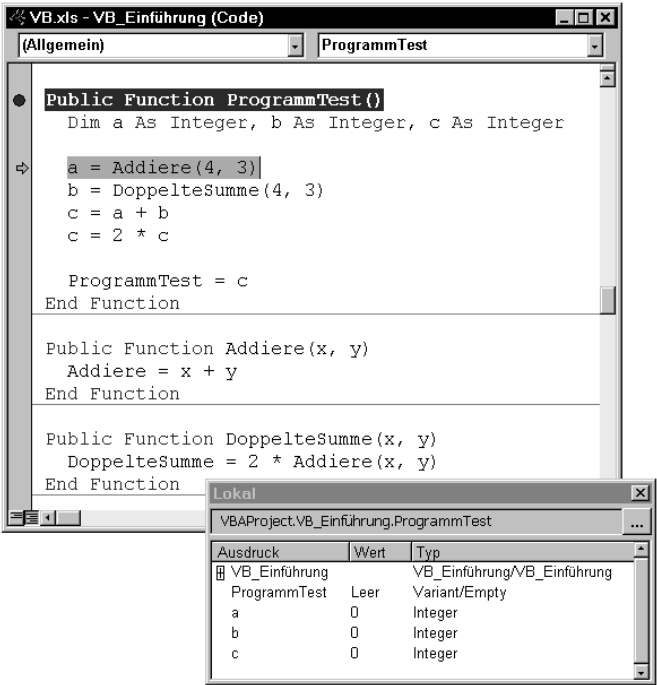


Die Programmausführung wird am zuvor definierten Haltepunkt unterbrochen. Öffnen Sie mit ANSICHT|LOKAL-FENSTER das Fenster *Lokal*, so erhalten Sie zusätzliche Informationen: Der Hinweis »VBAProject.VB_Einführung. ProgrammTest« am oberen Rand des Lokalfensters besagt, dass momentan die Prozedur *ProgrammTest* ausgeführt wird, die sich im Modul »VB_Einführung« von VB.XLS befindet. Darunter werden die Werte angezeigt, die die lokalen Variablen *a*, *b* und *c* dieser Prozedur momentan besitzen.



Die nächste auszuführende Programmzeile, also die Zeile, an der die Unterbrechung erfolgte, wird farblich und durch einen Pfeil hervorgehoben. Um die darin enthaltene Anweisung ausführen zu lassen, wählen Sie DEBUGGEN|EINZELSCHRITT (F8) (Abbildung 32.30).

Abbildung 32.30:
Einzelschritt



Nur diese eine Zeile mit der darin enthaltenen *Function...-Anweisung* wird ausgeführt, was jedoch keine offensichtlichen Auswirkungen hat.

Die Haltepunktzeile ist hervorgehoben wie zuvor, der Pfeil weist nun jedoch nicht mehr auf diese, sondern auf die nächste auszuführende Programmzeile (und zwar nicht auf *Dim a As Integer...*, sondern auf die folgende Zeile *a = Addiere(4, 3)*, da *Dim* keine ausführbare Anweisung ist).

Die Zwischenergebnisse *a*, *b* und *c* besitzen immer noch die Werte 0. Das ändert sich jedoch gleich. Zurzeit ist die Zeile

```
a = Addiere(4, 3)
```

die hervorgehobene Anweisung, die als nächste ausgeführt wird.

Wäre die Funktion *Addiere* uninteressant für Sie, könnten Sie nun **DEBUGGEN** | **PROZEDURSCHRITT** ( + **F8**) wählen: Die komplette Funktionsprozedur *Addiere* würde daraufhin als ein Schritt behandelt und im Eiltempo durchlaufen, ohne Unterbrechung nach jeder darin enthaltenen Programmzeile.



Halten Sie es jedoch für möglich, dass sich der Fehler, für den Sie sich interessieren, gerade in dieser Funktionsprozedur verbirgt, also in *Addiere*, dann durchlaufen Sie auch die in dieser Prozedur enthaltenen Anweisungen einzeln und drücken erneut **F8**.

Dadurch verzweigt die Programmausführung zur aufgerufenen Prozedur *Addiere* und in der oberen Hälfte des Lokalfensters werden nun die Werte der lokalen Variablen dieser Prozedur angezeigt (Abbildung 32.31).

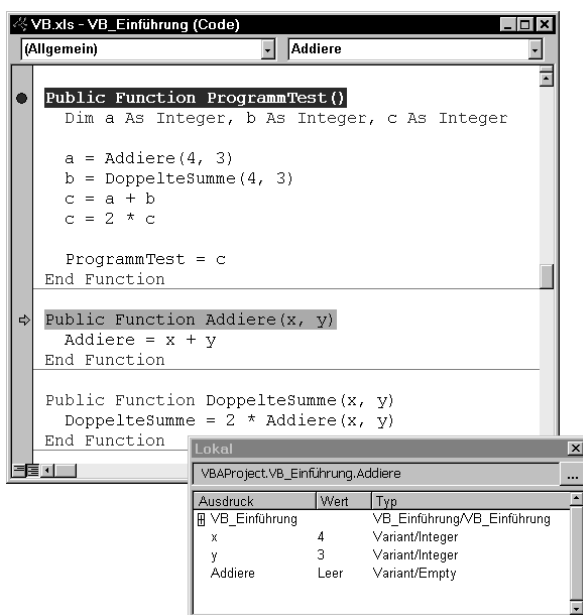
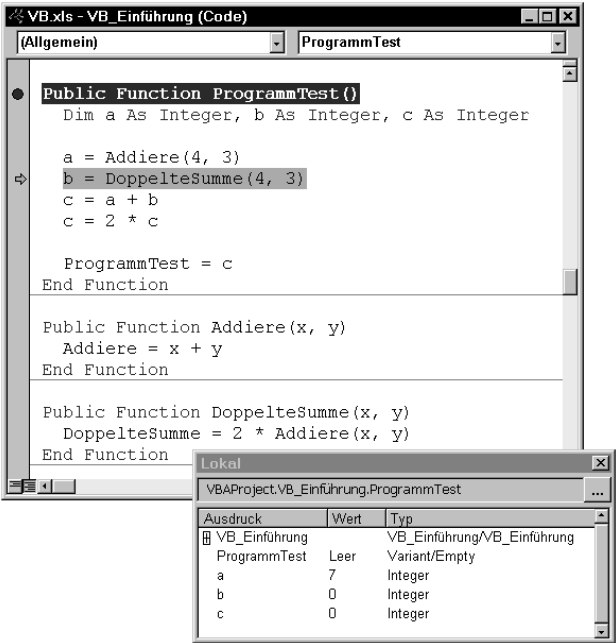


Abbildung 32.31:
Verzweigung in
Prozedur

Mit **F8** können Sie nun die darin enthaltenen Anweisungen Schritt für Schritt ausführen lassen. Nach Ausführung der letzten darin enthaltenen Anweisung *End Function* wird wieder zur aufrufenden Funktion zurückgekehrt und darin die nächste auszuführende Zeile (Abbildung 32.32) hervorgehoben, *b = DoppelteSumme(4, 3)*.

Abbildung 32.32:
Geänderter
Variablenwert



Nach Ausführung der Funktion *Addiere* hat sich der Wert der Variablen *a* geändert, da darin das von *Addiere* ermittelte Resultat 7 gespeichert wurde.

Derartige Änderungen von Zwischenergebnissen können Sie im Lokalfenster Schritt für Schritt verfolgen, um herauszubekommen, ab welchem Punkt ein bestimmtes Zwischenergebnis einen falschen Wert enthält.

Zusätzliche Möglichkeiten:

-  **AUSFÜHREN|UNTERBRECHEN** (**Strg** + **Pause**) unterbricht die Programmausführung, was vor allem dann nützlich ist, wenn sich Ihr Programm offenbar in einer Endlosschleife befindet und sonst ewig weiterlaufen würde.
-  **AUSFÜHREN|FORTSETZEN** (**F5**) führt den Rest des Programms im gewohnten Tempo aus, ohne Unterbrechung.
-  **AUSFÜHREN|ZURÜCKSETZEN** beendet die Programmausführung vorzeitig und setzt zusätzlich alle Variablen auf ihre Ausgangswerte (0) zurück.



Diesen Befehl müssen Sie wählen, wenn Sie das Modul- oder ein sonstiges Fenster nicht mehr verlassen und kein anderes Objekt mehr aktivieren können, wenn VBA blockiert erscheint!

DEBUGGEN|NÄCHSTE ANWEISUNG FESTLEGEN (**Strg** + **F9**) ignoriert alle Programmzeilen ab der aktuellen bis zu jener, in der sich momentan der Cursor befindet und die dadurch als nächste ausgeführt wird (sie muss sich in der gleichen Prozedur befinden).



DEBUGGEN|NÄCHSTE ANWEISUNG ANZEIGEN hebt die nächste auszuführende Anweisung hervor und ist nützlich, wenn Sie sich momentan woanders befinden, sich also irgendeine andere Programmstelle anschauen.



DEBUGGEN|AUSFÜHREN BIS CURSOR-POSITION (**Strg** + **F8**) führt alle Programmzeilen ab der aktuellen bis zu jener, in der sich zurzeit der Cursor befindet, im Normalmodus aus, ohne Unterbrechung. Erst bei jener Programmzeile wird wieder angehalten.



DEBUGGEN|PROZEDUR ABSCHLIESSEN (**Strg** + **⇧** + **F8**) ist nützlich, wenn Prozeduren andere Prozeduren aufrufen, diese wiederum weitere Prozeduren etc., Sie das alles jedoch nicht im Einzelnen schrittweise durchlaufen wollen: ab der aktuellen Anweisung wird der Prozedur-Baum ohne Unterbrechung nach unten durchlaufen, danach wieder nach oben bis zur Prozedur, in der Sie diesen Befehl wählten, diese wird bis zum Ende durchlaufen und danach in die darüberliegende Prozedur zurückgekehrt.



DEBUGGEN|HALTEPUNKT EIN-/AUS (**F9**) setzt in der aktuellen Programmzeile einen Haltepunkt bzw. löscht ihn, falls diese Programmzeile bereits einen Haltepunkt enthält (einfacher: Setzen = auf die graue Leiste links neben der betreffenden Zeile klicken; Löschen = dort erneut klicken).



DEBUGGEN|ALLE HALTEPUNKTE LÖSCHEN (**Strg** + **⇧** + **F9**) löscht alle bisher definierten Haltepunkte.

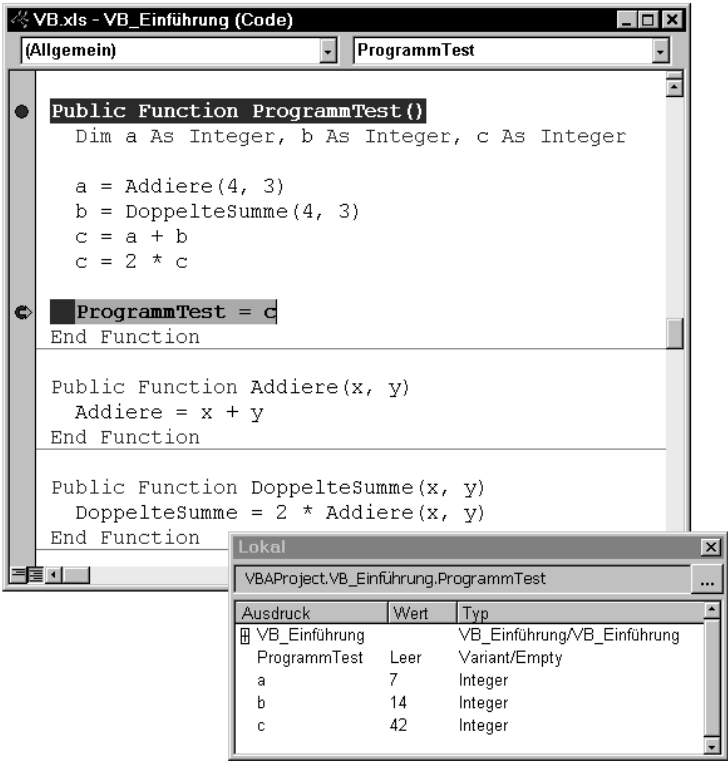


Der letzte Befehl deutet an, dass Sie beliebig viele Haltepunkte definieren können, beispielsweise außer dem bereits erläuterten einen zusätzlichen Haltepunkt, der sich am Ende von *ProgrammTest* befindet, in der Zeile *ProgrammTest = c*.

Angenommen, nach der Unterbrechung in der ersten Zeile lassen Sie einige Zeilen schrittweise ausführen und wählen irgendwann AUSFÜHREN|FORTSETZEN: Das Programm wird daraufhin zwar in vollem Tempo ohne Unterbrechung ausgeführt, aber nur bis der zweite Haltepunkt erreicht ist und dadurch eine neue Unterbrechung eintritt (Abbildung 32.33).

Sie können somit vor allen besonders interessanten Programmteilen Haltepunkte setzen und die dazwischen liegenden uninteressanten Teile bis zum Erreichen des jeweils nächsten Haltepunkts mit voller Geschwindigkeit ausführen lassen. Sie können sogar während einer Programmunterbrechung jederzeit Haltepunkte setzen oder löschen.

Abbildung 32.33:
Mehrere
Haltepunkte



Der entscheidende Punkt bei der schrittweisen Ausführung eines Programms ist die ständige Überwachung bestimmter Werte. Im Lokalfenster werden zwar automatisch die aktuellen Werte aller Variablen angezeigt, die die gerade ausgeführte Prozedur enthält.

Was aber, wenn Sie in einer Prozedur, die Dutzende von Variablen enthält, nur einen Teil davon sehen wollen, beispielsweise nur *a* und *b*, oder weniger am Wert von *a* interessiert sind als vielmehr am Doppelten der Summe der Variablen *a* und *b*, also am aktuellen Wert des Ausdrucks $2 * (a + b)$?

Kein Problem, wenn im Register *Editor* des Befehls EXTRAS|OPTIONEN... *Automatische Daten-Tipps* aktiviert ist: Ist der Einzelschrittmodus aktiv und befindet sich der Cursor auf einer Variablen, wird dadurch unmittelbar daneben der aktuelle Wert der Variablen angezeigt.

Sie können sogar einen Ausdruck markieren und den Cursor danach über den Ausdruck bewegen, um sich daneben automatisch den aktuellen Wert anzeigen zu lassen (Abbildung 32.34).

Abbildung 32.34:
Automatische
Daten-Tipps

c = a + b
c = a + b = 21

DEBUGGEN|AKTUELLEN WERT ANZEIGEN... ($\square + F9$) zeigt übrigens ebenfalls den aktuellen Wert eines zuvor markierten Ausdrucks an, ermöglicht jedoch im zugehörigen Dialogfeld, diesen Ausdruck mit der Schaltfläche *Hinzufügen* wie nun besprochen während der Programmausführung ständig überwachen zu lassen.

Wert anzeigen

Für die ständige Überwachung beliebiger Ausdrücke während der Programmausführung ist der Befehl DEBUGGEN|ÜBERWACHUNG HINZUFÜGEN... zuständig (Abbildung 32.35).

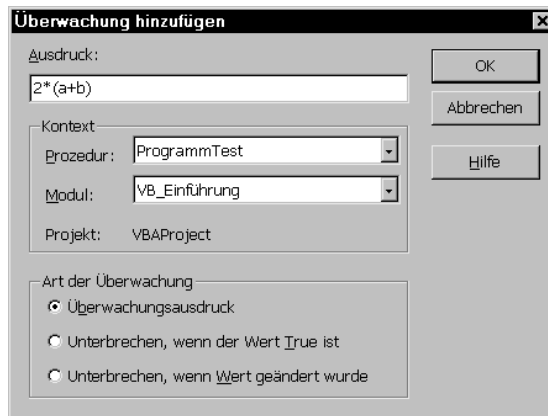


Abbildung 32.35:
Überwachung
hinzufügen

Ausdruck kann ein Zwischenergebnis sein, eine Variable, aber auch ein komplexer Ausdruck wie $2 * (a + b)$. Definieren Sie bitte nacheinander die folgenden Ausdrücke, indem Sie nach der Eingabe jeweils *OK* klicken und danach wieder DEBUGGEN|ÜBERWACHUNG HINZUFÜGEN... wählen, um den nächsten Ausdruck einzugeben:

$2 * (a + b)$

a

b

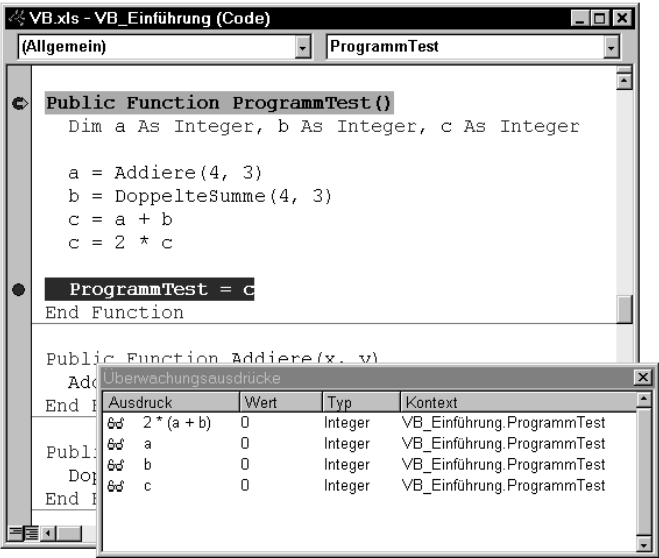
c

Rufen Sie nach der Festlegung der zu überwachenden Ausdrücke die Funktion *ProgrammTest* erneut im Direktfenster auf, um sie vom Beginn an ausführen zu lassen (Abbildung 32.36).

Nach der Definition zu überwachender Ausdrücke wird automatisch das Überwachungsfenster eingeblendet. Wenn Sie es schließen, können Sie es jederzeit mit ANSICHT|ÜBERWACHUNGSFENSTER erneut einblenden.

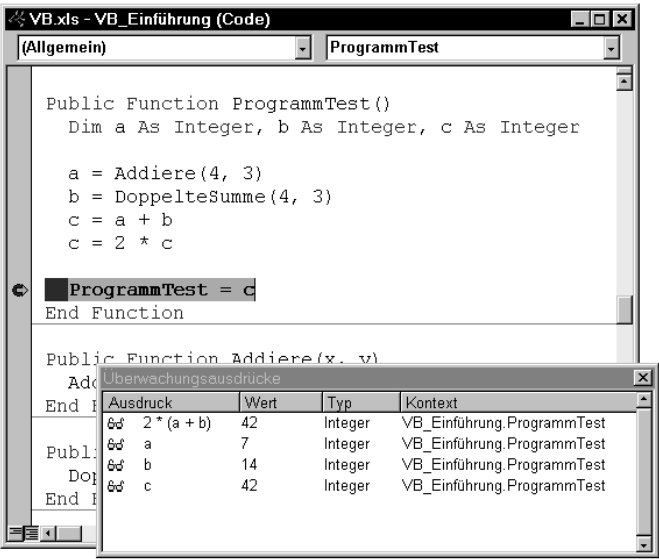
In diesem Fenster werden die überwachten Ausdrücke angezeigt, inklusive den aktuellen Werten und dem Typ dieser Ausdrücke und dem Kontext, den ich gleich näher erläutere. Ändert sich der Wert eines überwachten Ausdrucks, zeigt VBA automatisch den neuen Wert an.

Abbildung 32.36:
Überwachte
Ausdrücke anzeigen



Nach einigen Einzel- oder Prozedurschritten sieht das Ergebnis beispielsweise so aus (Abbildung 32.37).

Abbildung 32.37:
Aktuelle Werte der
überwachten
Ausdrücke



Überwachungsort

Zur Bedeutung der *Kontext*-Optionen im *Überwachung hinzufügen*-Dialogfeld: Mehrere Prozeduren können gleichnamige Variablen enthalten. Vielleicht enthält nicht nur *ProgrammTest* eine Variable namens *a*, sondern auch eine Prozedur namens *Demo*, die sich im gleichen oder in einem anderen Modul befindet.

Um dieser Situation gerecht zu werden, enthält das Listenfeld *Prozedur* alle Prozeduren des Moduls, das Sie im Listenfeld *Modul* auswählen. Sie können also mit Hilfe beider Listenfelder angeben, wo sich die zu überwachende Variable *a* befindet, in welcher Prozedur welchen Moduls.

Wollen Sie den Wert der Variablen *a* der Prozedur *ProgrammTest* überwachen, muss im Listenfeld *Modul* das Modul »VB_Einführung« selektiert sein, in dem sich diese Prozedur befindet, und im Listenfeld *Prozedur* die Prozedur *ProgrammTest*.

Wollen Sie stattdessen den Wert der Variablen *a* einer Prozedur *Demo* überwachen, die sich in einem Modul namens »Test« befindet, so selektieren Sie entsprechend als Modul »Test« und als Prozedur *Demo*.

Alternativ dazu können Sie in beiden Listenfeldern die Einträge (*Alle Prozeduren*) bzw. (*Alle Module*) auswählen. Dann ist es VBA egal, wo sich die Variable *a* befindet: Wird gerade die Prozedur *Demo* des Moduls »Test« ausgeführt, wird der Wert dieser Variablen *a* angezeigt. Wird jedoch gerade die Prozedur *ProgrammTest* des Moduls »VB_Einführung« ausgeführt, wird eben der Wert jener gleichnamigen Variablen *a* angezeigt.

Nun wissen Sie auch, welche Bedeutung die Spalte *Kontext* des Überwachungsfensters hat: Sie in derartigen Fällen darüber zu informieren, *welche* von mehreren gleichnamigen Variablen gerade überwacht wird.

Unter *Art der Überwachung* ist *Überwachungsausdruck* vorselektiert. Mit dem Resultat, dass wie gezeigt der aktuelle Wert des betreffenden Ausdrucks angezeigt wird.

*Überwachungs-
arten*

Die Option *Unterbrechen, wenn der Wert True ist* unterbricht dagegen die Programmausführung, wenn der Ausdruck den Wahrheitswert WAHR annimmt.

Ein Beispiel: Am Ende der Prozedur *ProgrammTest* besitzt die Variable *b* den Wert 14. Angenommen, dieser Wert ist absolut unerwünscht und beruht vermutlich auf einem zuvor aufgetretenen Fehler. Statt diese Variable nun ständig zu überwachen und zu prüfen, wann sie den Wert 14 annimmt, geben Sie unter *Ausdruck*

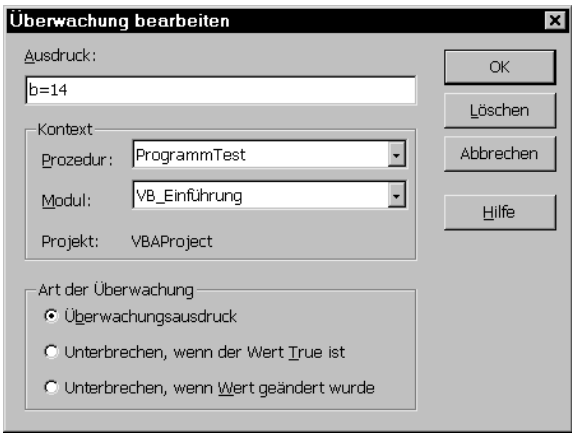
b = 14

ein und aktivieren *Unterbrechen, wenn der Wert True ist*.

Um die Überwachung des Ausdrucks *b* zu ändern, klicken Sie im Überwachungsfenster auf die betreffende Zeile und wählen danach DEBUGGEN|ÜBERWACHUNG BEARBEITEN... . Das gleiche Dialogfeld wie bei DEBUGGEN|

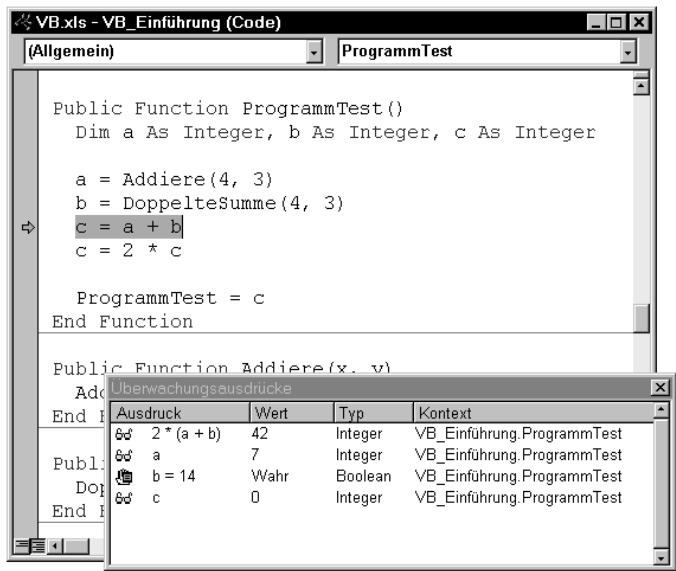
ÜBERWACHUNG HINZUFÜGEN... erscheint und Sie können den zuvor selektierten Ausdruck nun in $b = 14$ editieren und die Option *Unterbrechen* wählen, *wenn der Wert True ist* (Abbildung 32.38).

Abbildung 32.38:
Unterbrechung,
wenn definiertes
Ereignis eintritt



Statt im Einzelschrittmodus führen Sie die Prozedur in normalem Tempo aus, entfernen also vor dem erneuten Aufruf alle Haltepunkte oder wählen nach der Unterbrechung AUSFÜHREN|FORTSETZEN (Abbildung 32.39).

Abbildung 32.39:
Eintreten des
Ereignisses



Die nächste auszuführende Anweisung ist

$c = a + b$

Das bedeutet, dass die vorhergehende Anweisung

```
b = DoppelteSumme(4, 3)
```

bewirkte, dass *b* den Wert 14 annahm, der die Unterbrechung hervorrief. Dass die Unterbrechung durch die Änderung von *b* verursacht wurde, wird übrigens durch das Spezial-Symbol neben diesem Ausdruck angezeigt.

Die Option *Unterbrechen, wenn der Wert geändert wurde* arbeitet ähnlich: Jedes Mal, wenn sich der Wert des angegebenen Ausdrucks ändert, tritt eine Programmunterbrechung auf. Diese Option ist ideal, wenn Sie den Wert einer bestimmten Variable verfolgen und wissen wollen, welche Anweisungen ihn verändern.

Um einen überwachten Ausdruck zu löschen, selektieren Sie ihn im Überwachungsfenster und drücken Entf. Um ihn zu bearbeiten, selektieren Sie ihn ebenfalls und wählen danach den Befehl DEBUGGEN|ÜBERWACHUNG BEARBEITEN..., der das gleiche Dialogfeld wie DEBUGGEN|ÜBERWACHUNG HINZUFÜGEN... öffnet und in dem Sie den zuvor selektierten Ausdruck nun editieren oder andere Optionen wählen können.

*Ausdruck löschen
und bearbeiten*

32.3 Add-Ins erstellen

Ihre Programme können alles Mögliche enthalten: selbstdefinierte Funktionen, Prozeduren, die eine komplette Lagerverwaltung übernehmen etc.

Bei der Weitergabe komplexer Programme, in denen viel geistige Arbeit steckt, gibt es ein Problem: Geben Sie einfach die XLS-Datei weiter, dann kann jeder Anwender Ihre Programme nicht nur benutzen, sondern auch verändern, beispielsweise Ihre Copyright-Hinweise durch andere ersetzen.

Um das zu verhindern, sollten Sie die Datei stattdessen mit DATEI|SPEICHERN UNTER... und der Auswahl des Dateiformats »Microsoft Excel-Add-In (*.xla)« als Add-In speichern.

Das Resultat: Die Datei wird im Add-In-Format gespeichert, das weder Sie noch irgendein Benutzer jemals wieder einsehen oder verändern können. Sie können sie nun unbesorgt weitergeben.

Für den Benutzer hat das Ganze sogar noch einen Vorteil: Er kann Ihr Add-In ebenso wie jedes andere mit dem Add-In-Manager verwalten. Es ist darin zwar zunächst noch nicht aufgeführt, das kann der Benutzer jedoch leicht mithilfe der Schaltfläche *Durchsuchen...* ändern. Selektiert er Ihr Add-In, das er auf seine Festplatte kopierte, wird dieses Add-In danach wie jedes andere auch in der Liste aufgeführt und kann wahlweise eingebunden oder wieder deaktiviert werden.

33 Sprachelemente

Nachdem Sie inzwischen mit der VBA-Oberfläche umgehen können, wird es Zeit, sich mit der Sprache selbst zu beschäftigen.

Das folgende Kapitel beschäftigt sich zunächst mit den grundlegenden Elementen, mit Anweisungen zur Ausgabe von Daten auf dem Bildschirm und zur Eingabe von Daten über die Tastatur. Das versetzt Sie bereits in die Lage, Programme zu schreiben, die den Anwender zur Eingabe bestimmter Texte oder Zahlen auffordern, diese Eingaben weiterverarbeiten und die Ergebnisse auf dem Bildschirm ausgeben.

Wirklich sinnvolle Programme zu schreiben erfordert jedoch den Einsatz von Variablen, in denen Eingaben, Zwischenergebnisse oder Resultate bis zur Weiterverarbeitung gespeichert werden. Also erläutere ich anschließend die Eigenschaften der verschiedenen Daten- und Variablentypen, die dazu verwendet werden.

Danach geht es um Konstanten, vor allem um die Unmenge an bereits vordefinierten Excel- und VBA-Konstanten, die das Programmiererleben ungemein vereinfachen.

Das wichtigste Objekt beim Programmieren sind Prozeduren. Ich erläutere im Detail, welche Prozedurarten es gibt und welche Eigenschaften sie besitzen.

Vor allem gehe ich im Detail auf die verschiedenen Möglichkeiten der Parameterübergabe (Übergabe als Referenz, als Wert etc.) und den Geltungsbereich sowohl von Variablen als auch von Prozeduren ein (lokale, globale und statische Variablen bzw. Prozeduren).

All das ermöglicht Ihnen, auch komplexeste Programme wohl strukturiert und somit gut änderungs- und erweiterbar zu erstellen.

Danach sind Sie beispielsweise in der Lage, Excel mit selbsterstellten Funktionsprozeduren um zusätzliche Funktionen zu erweitern, die Sie wie die eingebauten Funktionen (*Summe*, *Mittelwert* etc.) benutzen können.

Im letzten Teil geht es um spezielle Formen von Variablen, um Datenfelder oder Arrays, die untrennbar mit den erst später erläuterten Schleifenstrukturen verbunden sind, und um selbstdefinierte Datentypen, so genannte Verbunde oder Records, die aus einem von Ihnen frei definierbaren Mischmasch grundlegender Datentypen bestehen und mit denen Sie sich praktisch Ihre eigenen Variablentypen basteln können.

33.1 Ein-/Ausgabeeweisungen

33.1.1 Debug.Print

Die grundlegende Ausgabeanweisung ist die *Print*-Methode. Wie Sie wissen, können Sie Aufrufe dieser Methode im Direktfenster eintippen, um damit in diesem Fenster Daten auszugeben.

Zusätzlich können Sie diese Anweisung auch in Ihren Programmen verwenden, um Daten im Direktfenster auszugeben, während ihr Programm abgearbeitet wird. Dann müssen Sie jedoch angeben, dass als Ausgabeobjekt das Direktfenster gemeint ist, und Sie müssen folgende Form verwenden:

```
Debug.Print Ausdruck
```

Debug repräsentiert das Direktfenster. *Debug.Print* bedeutet somit, dass die im nachfolgenden *Ausdruck* angegebenen Daten in diesem Fenster ausgegeben werden sollen. Das passiert, auch wenn das Direktfenster gerade geschlossen ist: Öffnen Sie es, nachdem Ihre Prozedur beendet ist, so sehen Sie darin die zuvor von der Prozedur im Direktfenster ausgegebenen Daten.

Jedem Aufruf von *Print* folgt ein Zeilenumbruch. Zwei aufeinander folgende Aufrufe wie

```
Debug.Print "Test"  
Debug.Print "Noch ein Test"
```

geben daher die beiden Zeichenketten »Test« und »Noch ein Test« nicht neben-, sondern untereinander aus.

Endet *Ausdruck* mit einem Semikolon, wird dieser Zeilenumbruch unterdrückt. Daher gibt

```
Debug.Print "Test";  
Debug.Print "Noch ein Test"
```

im Direktfenster die Zeichenkette »TestNoch ein Test« aus.

Endet *Ausdruck* mit einem Komma, wird der Zeilenvorschub ebenfalls unterdrückt, die folgende Ausgabe jedoch in die nächste Bildschirmzone verlegt, wodurch für ein wenig Abstand zwischen aufeinander folgenden Ausgaben gesorgt wird:

```
Debug.Print "Test",  
Debug.Print "Noch ein Test"
```

gibt somit »Test Noch ein Test« aus.

Mit der *Print*-Methode können beliebig viele Ausgaben aneinander gereiht werden. Beispielsweise gibt

```
Debug.Print "Ein"; " "; "Test"
```

die Zeichenkette »Ein Test« aus und

```
Debug.Print "USt: "; x
```

die Zeichenkette »USt: 5«, falls die Variable *x* momentan den Wert 5 repräsentiert.

Um Ausgaben exakt zu positionieren, verwenden Sie die *Tab*-Anweisung:

```
Debug.Print Tab(x); "Zeichenkette"
```

Tab positioniert den Cursor vor der nächsten Ausgabe auf das Zeichen *Tab*
Nummer *x* der Spalte. Beispielsweise gibt

```
Debug.Print Tab(10); "Willi"; Tab(20); "Maier"
```

ab Spalte 10 die Zeichenkette »Willi« und ab Spalte »20« die Zeichenkette »Maier« aus.

33.1.2 MsgBox

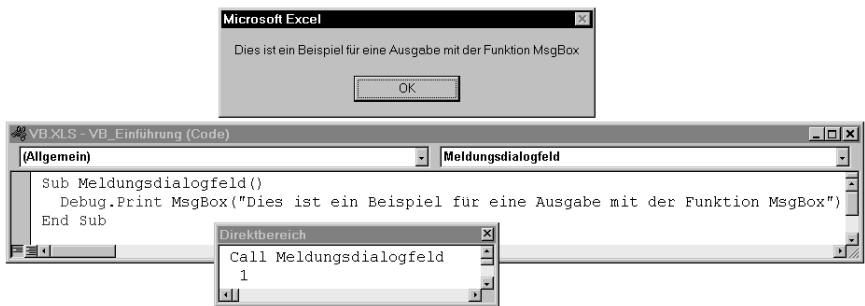
Debug.Print erfordert, dass Sie das Direktfenster öffnen, um sich die Ausgaben Ihres Programms anzuschauen.

Die Funktion *MsgBox* ist wesentlich angenehmer zu handhaben, da die Ausgaben Windows-gerecht in einem kleinen Dialogfeld präsentiert werden. Die Syntax (leicht vereinfacht) lautet

```
MsgBox(Meldung [,Schaltflächen] [,Titel])
```

Meldung ist die Zeichenkette, die in Dialogfeld erscheinen soll. Sie kann maximal 1024 Zeichen enthalten und wird von Excel bei Bedarf automatisch umbrochen. Ein Beispiel (Abbildung 33.1):

Abbildung 33.1:
Vb.xls, Modul
»VB_Einführung«,
Prozedur
»Meldungs-
Dialogfeld«



Um die abgebildete Funktion *Meldungs-Dialogfeld* aufzurufen, öffnen Sie das Direktfenster und geben darin *Call Meldungs-Dialogfeld* ein.

Daraufhin öffnet die in der Prozedur aufgerufene *MsgBox*-Funktion das abgebildete Dialogfeld (auf den nach dem Schließen des Dialogfelds übergebenen und dank *Debug.Print* im Direktfenster ausgegebenen Funktionswert 1 gehe ich in Kürze ein).

Das Dialogfeld erscheint übrigens nicht im VBA-, sondern im Excel-Fenster, auf das dabei automatisch umgeschaltet wird.

Zeilenumbruch Durch Einfügen des ASCII-Codes 13 (Wagenrücklauf) in die Zeichenkette können Sie Zeilenumbrüche selbst definieren. Dazu fügen Sie dort, wo eine neue Ausgabezeile beginnen soll, den ASCII-Code 13 ein und benutzen dazu folgende Syntax:

```
MsgBox("Zeichenkettel" + Chr(13) + "Zeichenkette2" + Chr(13) +  
"Zeichenkette3" + Chr(13) + " ... ")
```

Titel *Titel* ist eine Zeichenkette, die den Inhalt der Titelzeile des Dialogfelds bestimmt, beispielsweise »Mein Dialogfeld«. Ohne dieses Argument verwendet Excel immer die Überschrift »Microsoft Excel«.

Schaltflächen *Schaltflächen* ist eine Zahl, die bestimmt, welche Schaltflächen das Dialogfeld enthält. Die Zahl ergibt sich aus der Addition einzelner Werte, wobei folgende Werte zur Verfügung stehen:

Tabelle 33.1:
Argument
»Schaltflächen«

Wert	Bedeutung
Schaltflächen	
0	Nur Schaltfläche OK anzeigen
1	OK und Abbrechen anzeigen
2	Beenden, Wiederholen und Ignorieren anzeigen

Wert	Bedeutung
Schaltflächen	
3	<i>Ja, Nein</i> und <i>Abbrechen</i> anzeigen
4	<i>Ja</i> und <i>Nein</i> anzeigen
5	<i>Wiederholen</i> und <i>Abbrechen</i> anzeigen
Symbolart	
0	Kein Symbol
16	Stopp-Symbol
32	Fragezeichen-Symbol
48	Ausrufezeichen-Symbol
64	Informations-Symbol
Standardschaltfläche	
0	Erste Schaltfläche
256	Zweite Schaltfläche
512	Dritte Schaltfläche
768	Vierte Schaltfläche
Bindungsart	
0	Anwendungsgebunden (bis zur Beantwortung nur Wechsel zu anderer Anwendung möglich)
4096	Systemgebunden (auch kein Wechsel zu anderer Anwendung möglich)

Tabelle 33.1:
Argument
»Schaltflächen«
(Forts.)

Ohne das Argument *Schaltflächen* bzw. mit dem Wert 0 wird nur die OK-Schaltfläche angezeigt. Der Wert 1 würde beispielsweise zusätzlich die Schaltfläche *Abbrechen* einfügen.

Soll nicht die erste Schaltfläche OK, sondern die zweite Schaltfläche *Abbrechen* die durch  aktivierte Standardschaltfläche sein, so addieren Sie zu dieser 1 den Wert 256, übergeben also 257.

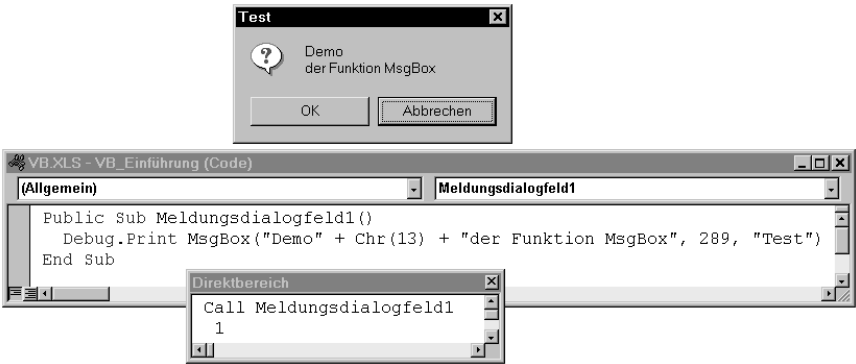
Standard-
schaltfläche

Soll zusätzlich ein Fragezeichen-Symbol im Dialogfeld erscheinen, addieren Sie zusätzlich 32 und übergeben somit 289 (Abbildung 33.2):

Symbole

```
MsgBox("Demo" + Chr(13) + " der Funktion MsgBox", 289, "Test")
```

Abbildung 33.2:
Vb.xls, Modul
»VB_Einführung«,
Prozedur
»Meldungs-
Dialogfeld1«



Da unter anderem eine 256 addiert wurde, ist die zweite Schaltfläche *Abbrechen* vorselektiert und kann mit  aktiviert werden.

Das Argument »Test« für *Titel* gibt die Zeichenkette »Test« in der Titelleiste aus.

Die Funktion *MsgBox* übergibt nach dem Schließen des Dialogfelds einen der folgenden Funktionswerte, der im Beispiel nach dem Schließen des Dialogfelds mit *Debug.Print* im Direktfenster ausgegeben wird:

Tabelle 33.2:
Übergebener
Funktionswert

Wert	Schaltfläche
1	OK
2	Abbrechen
3	Beenden
4	Wiederholen
5	Ignorieren
6	Ja
7	Nein

33.1.3 InputDialog

Ebenso einfach anzuwenden ist die Funktion *InputDialog*. Mit ihr können Sie Eingaben in einem Dialogfeld entgegennehmen:

`InputDialog(Eingabeaufforderung [, [,Titel][[,Standard][,XPosition,
YPosition]])`

Eingabeaufforderung ist wieder eine im Dialogfeld anzuzeigende Zeichenkette und *Titel* erneut die Dialogfeldüberschrift.

XPosition und *YPosition* legen den horizontalen/vertikalen Abstand des linken/oberen Dialogfeldrands vom linken/oberen Bildschirmrand fest.

Titel bestimmt wieder den Inhalt der Titelzeile des Dialogfelds und ist eine Zeichenkette wie »Mein Dialogfeld«. Ohne Positionsangaben wird das Dialogfeld in der Bildschirmmitte zentriert (Abbildung 33.3).

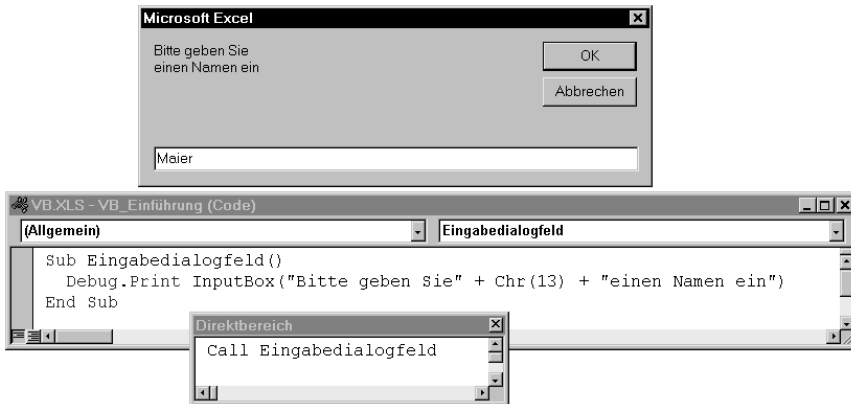


Abbildung 33.3:
Vb.xls, Modul
»VB_Einführung«,
Prozedur »Eingabe-
Dialogfeld«

InputBox fordert den Anwender zu einer Eingabe auf. Gibt er wie hier die Zeichenkette »Maier« ein und aktiviert er *OK*, so übergibt *InputBox* die eingegebene Zeichenkette als Funktionswert.

In diesem Beispiel wird die Eingabe anschließend mit *Debug.Print* im Direktfenster ausgegeben. In der Praxis können Sie den eingegebenen Namen stattdessen beispielsweise in einer Tabelle speichern.

Gibt der Anwender nichts ein oder bricht er die Eingabe mit *Abbrechen* ab, übergibt *InputBox* eine leere Zeichenkette » " « der Länge 0.

InputBox gibt immer eine Zeichenkette zurück (den Datentyp *Char*) – auch dann, wenn der Anwender eine Zahl wie 22,3 eingab.



Benötigen Sie die Zahl als echte *Zahl* und nicht als Folge einzelner Zeichen, weil Sie sie anschließend einer so genannten numerischen Variablen zuweisen wollen, so müssen Sie sie mit der *Value*-Funktion in eine echte Zahl umwandeln.

Beispielsweise ermittelt *x# = Value(s\$)* den numerischen Wert der in einer Stringvariablen *s\$* gespeicherten Zeichenfolge wie »23,4« und weist der numerischen Variablen *x#* daher den Wert 23,3 zu.

33.2 Codezeilen, Anweisungen und Aktionen

Grundlage von VBA-Programmen sind einzelne Codezeilen, die eine oder mehrere durch Doppelpunkte voneinander getrennte Anweisungen enthalten:

```
Debug.Print "Ein Test": Debug.Print "Noch ein Test"
```

Beide Anweisungen werden nacheinander ausgeführt, so als würden sie sich in zwei getrennten Programmzeilen befinden.

Kommentare Nicht jede Anweisung bewirkt etwas. Die Anweisung *Rem* leitet ebenso wie das Zeichen »'« (⏏+⏏) einen Kommentar ein:

```
Rem Dies ist ein Kommentar  
' Dies ist ein Kommentar
```

Trifft VBA beim Interpretieren einer Programmzeile auf die Kommentaranweisung *Rem* oder auf das Zeichen »'«, so ignoriert es den gesamten Rest der Zeile. Diese Eigenschaft betrifft jedoch nicht den Programmtext *vor* dem Kommentarzeichen. Daher können Sie eine Programmzeile zum Beispiel so kommentieren:

```
Debug.Print "Gleich kommt ein Kommentar" : Rem Ein Kommentar
```

oder

```
Debug.Print "Gleich kommt ein Kommentar" 'Ein Kommentar
```

Einen mit dem Kommentarzeichen »'« eingeleiteten Kommentar müssen Sie ausnahmsweise nicht per Doppelpunkt von dem zu kommentierenden Programmcode trennen, sodass es sich aus optischen Gründen anbietet, statt *Rem* dieses Zeichen zu verwenden.

33.3 Variablen

Jede Programmiersprache kennt zwei grundlegende Datentypen, Zahlen und Zeichen, besser gesagt *Zeichenketten*, so genannte Strings (»Maier«, »Willi« etc.), die in Anführungszeichen eingeschlossen werden:

```
Debug.Print "10"  
Debug.Print "Test"
```

Auf beide Datenarten kann auch indirekt über so genannte Variablen zugegriffen werden. Dazu werden die Daten irgendwo im Speicher mit der *Let*-Anweisung unter einem frei wählbaren Namen abgelegt, dem Variablennamen:

Let Variablenname = Ausdruck

Man sagt, der Variablen *Variablenname* wird der Wert *Ausdruck* zugewiesen. Anschließend können Sie in beliebigen Anweisungen unter Angabe dieses Namens auf die gespeicherten Daten zugreifen, sie lesen oder verändern. Das Schlüsselwort *Let* ist optional und wird meist weggelassen, sodass sich folgende Syntax ergibt:

Variablenname = Ausdruck

Variablennamen

- ➔ müssen mit einem Buchstaben beginnen,
- ➔ dürfen nur Buchstaben, Ziffern und das Zeichen »_« enthalten,
- ➔ können maximal 200 Zeichen lang sein,
- ➔ dürfen keine Satzzeichen (».«, »;« etc.) und keine Leerzeichen enthalten
- ➔ und dürfen nicht mit einem reservierten Schlüsselwort identisch sein (versuchen Sie also nicht, eine Variable *Print*, *Debug*, *MsgBox* oder ähnlich zu nennen).

Die gleichen Regeln gelten übrigens auch für Konstanten- und Prozedurnamen!

Ein kleines Beispiel für die Anwendung von Variablen (Abbildung 33.4):

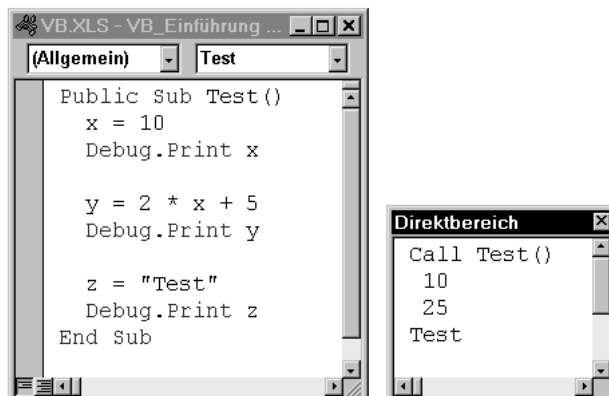


Abbildung 33.4:
Vb.xls, Modul
»VB_Einführung«,
Prozedur »Test«

Nach dem Aufruf der Prozedur *Test* im Direktfenster mit *Call Test()* weist die Anweisung *x = 10* der Variablen *x* den Wert 10 zu, den VBA irgendwo im Rechnerspeicher ablegt. *Debug.Print x* greift unter Angabe dieses Namens auf den zugehörigen Wert zu und gibt ihn im Direktfenster aus.

Die zweite Anweisung *y = 2 * x + 5* zeigt, dass bei einer Zuweisung auch rechts vom Gleichheitszeichen, also im Zuweisungsausdruck, der den zu speichernden Wert festlegt, Variablen verwendet werden können: *x* besitzt nach der vorhergehenden Zuweisung den Wert 10. Der Ausdruck *2 * x + 5* besitzt daher den Wert 25. Er wird einer weiteren Variablen *y* zugewiesen und der dadurch definierte Inhalt 25 dieser Variablen wird ebenfalls im Direktfenster ausgegeben.

Die folgende Anweisung zeigt, dass Sie einer Variablen auch Zeichenketten zuweisen können – und dass die betreffende Zeichenkette dabei in Anführungszeichen eingeschlossen werden muss.



*Jede Zuweisung an eine bereits bestehende Variable überschreibt den momentan darin gespeicherten Wert. Weisen Sie mit *x = 10* der Variablen *x* den Wert 10 und sofort danach mit *x = 20* den Wert 20 zu, wird eine darauf folgende Anweisung *Debug.Print x* die Zahl 20 ausgeben, also jene Zahl, die *x* zuletzt zugewiesen wurde und die daher den zuvor in *x* gespeicherten Wert 10 überschrieb.*

33.3.1 Datentypen



*Ist vor dem Anlegen eines neuen Moduls die Option Variablendeklaration erforderlich nicht aktiviert (Register Editor des Befehls EXTRAS|OPTIONEN...), so können Sie darauf verzichten, Variablen wie in Kürze erläutert mit dem Befehl *Dim* zu deklarieren (siehe Kapitel 33.3.2, »Deklarationen«).*

Variant

Ohne eine solche Deklaration und ohne zusätzliches Typdeklarationszeichen besitzt eine Variable immer den Standarddatentyp *Variant*, der beliebige Datenarten aufnehmen kann, Zahlen und Zeichen, ein Datum oder eine Zeitangabe.

Dieser Typ ist äußerst komfortabel, da VBA – falls notwendig – automatisch einen Datentyp in einen benötigten anderen Typ umwandelt. Ein Beispiel:

Listing 33.1:
Typkonvertierung

```
x = "12"
y = x - 10
Debug.Print y
```

x = »12« weist der Variablen *x* die Zeichenkette »12« zu und *y = x - 10* subtrahiert davon 10 und weist das Ergebnis der Variablen *y* zu – subtrahiert also eine Zahl von einer Zeichenkette.

Diesen Unfug würde außer VBA jede andere Programmiersprache mit einer Fehlermeldung ahnden. VBA nimmt jedoch eine automatische Typkonvertierung vor: Der Inhalt von x soll bei dieser Subtraktion offenbar als Zahl verwendet werden. Daher wandelt VBA die *Zeichenkette* »12« in die *Zahl* 12 um, von der nun problemlos die Zahl 10 subtrahiert werden kann.

Typkonvertierung

Eine Fehlermeldung erhalten Sie nur mit Ausdrücken, bei denen eine solche Konvertierung nicht möglich ist:

```
x = "Otto"
y = x - 10
Debug.Print y
```

Listing 33.2:
Typkonvertierung
nicht möglich

Dieser Versuch, von der Zeichenkette »Otto« die Zahl 10 zu subtrahieren, ist selbst VBA zuviel und Sie erhalten die Fehlermeldung, dass die Datentypen unverträglich sind.

Der Datentyp *Variant* ist zwar sehr komfortabel, aber VBA ist ständig mit der Überprüfung von Datentypen und gegebenenfalls mit entsprechenden Typumwandlungen beschäftigt, was Rechenzeit kostet.

All das entfällt mit eingeschränkten Variablentypen. Geben Sie von vornherein bekannt, dass in x nur Zahlen gespeichert sind, muss VBA in einem Ausdruck der Art $y = x - 10$ die in x enthaltene Datenart nicht mehr überprüfen, um beispielsweise festzustellen, dass der Inhalt von x möglicherweise eine Zeichenkette wie »12« ist, die erst in die Zahl 12 umgewandelt werden muss.

Aus Effizienzgründen sollten Sie nicht ständig den Datentyp *Variant* verwenden, sondern mit einem Typdeklarationszeichen bekannt geben, welche Datenarten die betreffende Variable aufnehmen soll:

Typdeklarationszeichen

Datentypname	Zeichen	Zulässige Werte	Belegter Platz
Byte	keines	0 bis 255 (nur ganze Zahlen)	1 Byte
Boolean	keines	Einer der beiden Wahrheitswerte <i>True</i> und <i>False</i>	2 Byte
Integer	%	-32.768 bis 32.767 (nur ganze Zahlen)	2 Byte
Long	&	-2.147.483.648 bis 2.147.483.647 (nur ganze Zahlen)	4 Byte
Single	!	-3,402823E38 bis 3,402823E38 (auch Dezimalzahlen; sechsstellige Genauigkeit)	4 Byte

Tabelle 33.3:
Datentypen

Tabelle 33.3:
Datentypen
(Forts.)

Datentypname	Zeichen	Zulässige Werte	Belegter Platz
Double	#	-1,79769313486232E308 bis 1,79769313486232E308 (auch Dezimalzahlen; zehnstellige Genauigkeit)	8 Byte
Currency	@	Zwischen - 922337203685477,5808 und 922337203685477,5808	8 Byte
Date	keines	1.Jan 100 bis 31.Dez 9999	8 Byte
Object	keines	Verweis auf ein beliebiges Objekt	4 Byte
String variabler Länge	\$	Zeichenkette, max. 2 Milliarden Zeichen	10 Byte plus die Textlänge
String fester Länge	keines	Zeichenkette, max. 2 Milliarden Zeichen	Die Textlänge
Variant	keines	beliebige Daten	je nach Datenart (Zahlen: 16 Byte; Zeichenketten: 22 Byte plus die Textlänge)

Bei den Datentypen ohne spezielles Typdeklarationszeichen müssen Sie die in Kapitel 33.3.2, »Deklarationen«, erläuterte Deklarationstechnik verwenden.



Datum/Uhrzeit-
eingabe

Beachten Sie, dass ein Datum oder eine Uhrzeit bei der Zuweisung in das Zeichen »#« eingeschlossen werden muss, um als Datum oder Uhrzeit erkannt zu werden:

Falsch: x = 1.Jan 2009
Richtig: x = #1.Jan 2009#

Zeichenketten

Zeichenketten müssen wie bereits erwähnt in Anführungszeichen eingeschlossen werden:

Falsch: x\$ = Hallo
Richtig: x\$ = "Hallo"

Sie können mehrere Zeichenketten zu einer einzigen verknüpfen, entweder mit dem Zeichen »&« oder mit dem Operator »+«. Die Ausdrücke


```
x$ = "Gerd" & " " & "Maier"
```

und

```
x$ = "Gerd" + " " + "Maier"
```

sind funktional äquivalent. Beide weisen der Stringvariablen (Zeichenkettenvariablen) *x\$* die Zeichenkette »Gerd Maier« zu.

Analog zum Tabellenentwurf sollten Sie immer jenen Typ wählen, der für die betreffende Anwendung gerade noch ausreicht. Verwenden Sie beispielsweise keine *Long*-Variable, um ganze Zahlen zwischen 100 und 200 zu speichern, sondern benutzen Sie dazu den Typ *Integer*.

Die einfachste Möglichkeit zur Festlegung des gewünschten Variablentyps besteht darin, an den Variablennamen das betreffende Typkennzeichen anzuhängen und die Variable einfach zu verwenden, wo sie gebraucht wird. Im gleichen Moment, in dem der betreffende Variablenname zum ersten Mal benutzt wird, reserviert VBA im Rechnerspeicher den dafür benötigten Platz.

Im Gegensatz zu vielen anderen Programmiersprachen können Sie nicht mit

```
x% = 10  
x$ = "Test"
```



*der Integervariablen *x%* die Zahl 10 und der Stringvariablen *x\$* die Zeichenkette »Test« zuweisen. Stattdessen erhalten Sie eine Fehlermeldung, die darauf beruht, dass es nur eine Variable namens *x* geben kann!*

33.3.2 Deklarationen

Sie können Variablen in einer Prozedur ohne besondere Umstände einfach benutzen und den Typ der Variablen mit dem zugehörigen Typkennzeichen festlegen, wenn Sie wollen (vorausgesetzt, vor dem Anlegen des Moduls war im Register *Editor* des Befehls *EXTRAS|OPTIONEN...* die Option *Variablen-deklaration erforderlich* deaktiviert).

Alternativ dazu können Sie eine Variable jedoch auch noch vor der ersten Benutzung mit einer der Anweisungen *Dim*, *Public*, *Private* oder *Static* deklarieren, wobei *Dim* die gebräuchlichste Deklarationsanweisung ist (die restlichen Anweisungen erläutere ich in Kapitel 33.7.2, »Variablen und Konstanten«):

Dim

```
Dim Variablenname[Typkennzeichen]
```

Zum Beispiel sehen Sie hier den Anfang einer Prozedur:

Listing 33.3:
Variablen
deklarieren

```
Function Test ()  
    Dim x%  
  
    x% = 10  
    Debug.Print x%  
End Function
```

Die *Dim*-Anweisung reserviert den für die Variable benötigten Platz im Rechner Speicher, obwohl er erst in der nächsten Programmzeile benötigt wird.



Ohne Typkennzeichen wird eine Variable vom Standardtyp Variant deklariert. Dim x deklariert daher eine Variable namens x dieses Typs.

Dim As

Wirklich nützlich ist die Deklaration mit *Dim* gegenüber der einfachen Anwendung einer Variablen jedoch erst mit folgender Syntax:

```
Dim Variablenname As Typ
```

Damit ist es möglich, den Datentyp einer Variablen festzulegen und dennoch nicht bei jeder Benutzung der Variablen das Typkennzeichen angeben zu müssen:

Listing 33.4:
Deklaration mit As

```
Function Test ()  
    Dim x As Integer  
  
    x = 10  
    Debug.Print x  
End Function
```

Nach der Deklaration mit *Dim x As Integer* steht für VBA ein für allemal fest, dass die in dieser Prozedur verwendete Variable *x* eine Integervariable ist, und Sie können bei folgenden Bezügen auf diese Variable auf das Typkennzeichen verzichten.

Im Gegensatz zu numerischen Variablen besitzen Stringvariablen normalerweise keine feste Länge. Nach der Deklaration einer Stringvariablen *s* mit

```
Dim s As String
```

speichert eine Zuweisung wie *s = »Test«* darin vier Zeichen und eine Anweisung wie *s = »Gerd Maier«* zehn Zeichen. Die Größe der Stringvariablen und der von ihr belegte Speicherplatz wird von VBA bei jeder neuen Zuweisung entsprechend angepasst – was allerdings viel Zeit kostet.

Wissen Sie genau, wie viele Zeichen eine bestimmte Stringvariable maximal aufnehmen wird, können Sie stattdessen Stringvariablen fester Länge verwenden, die mit einer Anweisung der Art

*Strings fester
Länge*

```
Dim Variablenname As String * Länge
```

deklariert werden. Zum Beispiel deklariert

```
Dim s As String * 10
```

eine Stringvariable *s* mit einer Länge von zehn Zeichen. Weisen Sie ihr mit *s* = »Otto« nur vier Zeichen zu, so füllt VBA die restlichen sechs Zeichen mit Leerzeichen auf, sodass darin tatsächlich die Zeichenkette »Otto« gespeichert ist. Weisen Sie ihr mit *s* = »Otto Maierbach« mehr als zehn Zeichen zu, so werden die überzähligen Zeichen abgeschnitten und darin nur »Otto Maier« gespeichert.

Sie können mit *Dim* auch mehrere Variablen auf einmal deklarieren:

```
Dim Variablenname As Datentypname, Variablenname As Datentypname,  
Variablenname As Datentypname, ...
```

Zum Beispiel deklariert die Anweisung

```
Dim x As Double, y As String, z As String * 10
```

drei Variablen: *x* ist eine Gleitkommavariablen doppelter Genauigkeit, *y* ist eine Stringvariable variabler Länge und *z* ist eine Stringvariable mit einer festen Länge von zehn Zeichen.

Die explizite Deklaration von Variablen scheint zunächst umständlich zu sein, besitzt in der Praxis jedoch nur Vorteile. Ich empfehle Ihnen dringend, dafür zu sorgen, dass VBA Sie dazu zwingt!

*Deklarations-
zwang*

Dazu fügen Sie in den Deklarationsabschnitt eines Moduls (Eintrag (*Deklarationen*) im *Prozeduren*-Listenfeld selektieren) die Anweisung *Option Explicit* ein. Wird irgendwo in diesem Modul eine nicht deklarierte Variable verwendet, erhalten Sie später bei der Ausführung der betreffenden Prozedur eine entsprechende Fehlermeldung.

Solange Sie das Kontrollkästchen *Variablendeklaration erforderlich* im Register *Editor* des Befehls EXTRAS|OPTIONEN... aktiviert lassen, fügt VBA diese Anweisung sogar automatisch in jedes neue Modul ein.

Dieser freiwillige Zwang zur Deklaration ist äußerst sinnvoll, wenn Sie Wert auf übersichtliche und gut strukturierte Programme legen, in denen unnötige Fehler von vornherein vermieden werden!

Geltungsbereich

Werden Variablen am Anfang einer Prozedur mit *Dim* deklariert, sind sie nur dieser einen Prozedur bekannt und können nur in ihr verwendet werden.

Alternativ dazu können Sie auch öffentliche Variablen deklarieren (siehe Kapitel 33.7.2, »Variablen und Konstanten«).



33.4 Konstanten

Jede Zahl oder Zeichenkette, die sich in Ihrem Programmtext befindet, ist eine *iterale Konstante*. Die Anweisung

```
Debug.Print "Hallo"; 3 * 7
```

enthält drei *iterale Konstanten*: die Stringkonstante »Hallo« und die beiden Zahlenkonstanten 3 und 7.

33.4.1 Selbstdefinierte Konstanten

Wesentlich interessanter sind *symbolische Konstanten*, die mit der Anweisung *Const* deklariert werden:

```
Const Konstantenname = Ausdruck
```

Das Argument *Konstantenname* unterliegt den gleichen Regeln wie ein Variablenname. Ohne Typkennzeichen wird der am besten passende Datentyp verwendet. *Ausdruck* ist ein beliebiger String- oder numerischer Ausdruck, der den Wert der Konstanten angibt, zum Beispiel »Hallo« oder $3 * 7$. Hier sehen Sie ein Beispiel, in dem zwei Konstanten deklariert werden:

```
Const a = "Hallo"  
Const x = 10 * 5
```

Wie bei *Dim* können Sie den Zusatz *As Typ* verwenden, um den Typ der Konstanten festzulegen:

```
Const Konstantenname As Typ = Ausdruck
```

Beispielsweise deklariert

```
Const a As String = "Hallo"
```

eine Stringkonstante.



Wie der Name bereits sagt, ist der Wert einer Konstanten konstant, also unveränderlich. Im Gegensatz zu einer Variablen kann er nach der Festlegung nicht mehr verändert werden. Es ist daher nicht möglich, einer Konstanten einmal mit der Anweisung `Const x = 5` den Wert 5 zuzuweisen und mit einer folgenden Anweisung `Const x = 10` einen anderen Wert zuweisen zu wollen!

Diese Eigenschaft von Konstanten ist immer dann nützlich, wenn Sie mit Werten hantieren (Zahlen oder Zeichenketten), die unveränderlich sind. Dann definieren Sie eine Konstante mit dem entsprechenden Wert und verwenden in Ihrem Programm statt der Zahl/Zeichenkette selbst den bequemer zu benutzenden symbolischen Namen, den Sie der Konstanten gaben.

Ein Beispiel für einen solchen normalerweise unveränderlichen Wert ist der Umsatzsteuersatz. Es bietet sich an, für entsprechende Berechnungen am Prozeduranfang eine Konstante zu verwenden, die folgendermaßen deklariert wird:

```
Const USt = 16
```

In Ihrem Programm können Sie nun in Berechnungen statt der Zahl 16 den Konstantennamen `USt` verwenden. Ändert sich irgendwann der Umsatzsteuersatz, müssen Sie nur diese Deklaration ändern.

Sie können Konstanten in Abhängigkeit von anderen Konstanten deklarieren, beispielsweise mit `Const A = 20` eine Konstante `A` und danach mit `Const B = A / 2` eine Konstante `B`, die den zuvor `A` zugewiesenen Wert benutzt. Ändern Sie irgendwann die Zuweisung an die Konstante `A`, ändert sich automatisch der `B` zugewiesene Wert entsprechend.

Für den Geltungsbereich von Konstanten gilt das Gleiche wie bei Variablen: Werden sie mit `Const` am Anfang einer Prozedur deklariert, sind sie nur lokal gültig und nur dieser einen Prozedur bekannt (siehe Kapitel 33.7.2, »Variablen und Konstanten«).

Geltungsbereich

33.4.2 Excel- und VBA-Konstanten

Excel und VBA stellen Ihnen eine Unmenge vordefinierter und global gültiger Konstanten zur Verfügung. Damit sind Konstanten gemeint, die einen von VBA oder von Excel vorgegebenen Wert besitzen und die Sie in beliebigen Prozeduren Ihrer Programme anstelle dieses Werts einsetzen können.

Die Namen dieser Konstanten beginnen immer mit zwei Buchstaben, die die Objektbibliothek bezeichnen, die sie zur Verfügung stellt. Excel-Konstanten beginnen beispielsweise mit `xl...` und VBA-Konstanten beginnen mit `vb...`.

Die Frage ist, was Sie mit diesen Konstanten anfangen können. Nehmen wir an, Sie benutzen häufig die Funktion `MsgBox`, der wie erläutert unterschiedliche Zahlenwerte übergeben werden, um ihr Verhalten näher festzulegen. Diese Funktion wird von der VBA-Objektbibliothek zur Verfügung gestellt, die zusätzlich mehrere Konstanten definiert, die den Umgang mit diesen Zahlenwerten erleichtert und Ihre Programme wesentlich lesbarer macht.

Beispielsweise besitzt die VBA-Konstante `vbOKOnly` den Wert 0, die Konstante `vbOKCancel` den Wert 1 und `vbQuestion` den Wert 32. Wollen Sie die Funktion `Msgbox` aufrufen und soll darin nur die OK-Schaltfläche erscheinen, so können Sie daher statt

```
MsgBox("Dateiname:", 0)
```

alternativ folgenden Ausdruck verwenden:

```
MsgBox("Dateiname:", vbOKOnly)
```

Sollen die OK- und die *Abbrechen*-Schaltfläche und zusätzlich ein Fragezeichen-Symbol erscheinen, können Sie statt

```
MsgBox("Dateiname:", 1 + 32)
```

beziehungsweise

```
MsgBox("Dateiname:", 33)
```

entsprechend folgende aussagekräftigere Alternative verwenden:

```
MsgBox("Dateiname:", vbOKCancel + vbQuestion)
```



Eine Übersicht über die vordefinierten MsgBox-Konstanten erhalten Sie im Objektkatalog, indem Sie die Sie interessierende Objektbibliothek auswählen, im Beispiel VBA, und im Listefeld darunter `vbMsgBoxResult` oder `vbMsgBoxStyle` selektieren. Im rechten Listefeld werden daraufhin die von der betreffenden Bibliothek zur Verfügung gestellten Konstanten aufgelistet (Abbildung 33.5).

33.5 VBA-Prozeduren

Excel (beziehungsweise VBA) stellt Ihnen eine ganze Menge eingebauter Prozeduren in Form von Funktionsprozeduren zur Verfügung, genauer: als numerische Funktionen und als Stringfunktionen:

- ➡ Numerische Funktionen übergeben als Funktionswert eine Zahl.
- ➡ Stringfunktionen übergeben eine Zeichenkette.

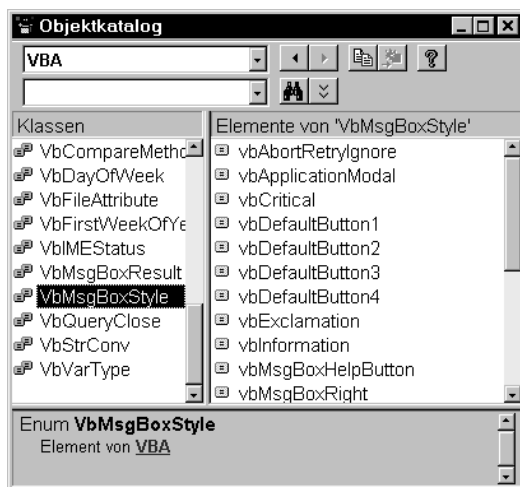


Abbildung 33.5:
Vordefinierte
MsgBox-Konstanten

Ein praktisches Beispiel für eine numerische Funktion ist *Sqr*, die die Quadratwurzel einer Zahl ermittelt:

Sqr

$v = \text{Sqr}(x)$

Sqr wird als Argument eine Zahl x übergeben. *Sqr* ermittelt die Quadratwurzel dieser Zahl und übergibt sie als Funktionswert.

Eine ebenfalls sehr häufig benutzte Funktion ist *Len*:

Len

$v = \text{Len}(x\$)$

Len ermittelt die Länge einer Zeichenkette, eines Strings, und übergibt als Funktionswert die Anzahl der im betreffenden String enthaltenen Zeichen. Zum Beispiel ermittelt *Len*(»Test«) die Länge der Zeichenkette »Test« und übergibt als Funktionswert die Zahl 4.

Der Funktion *Int* wird als Argument ein numerischer Wert x übergeben:

Int

$v = \text{Int}(x)$

Als Funktionswert v liefert *Int* wieder einen numerischen Wert (fehlendes Dollarzeichen), den Integer- oder ganzzahligen Anteil der übergebenen Zahl: die größte ganze Zahl, die kleiner oder gleich x ist. Nehmen wir als Beispiel die Zahl 3.45. Die größte ganze Zahl, die kleiner oder gleich 3.45 ist, ist 3, denn 4 ist bereits größer. Oder nehmen wir 167.12345. Der ganzzahlige Anteil dieser Zahl ist 167.

Bei negativen Zahlen wird ebenfalls der Nachkomma-Anteil entfernt. Aus -23,9 macht *Int* daher -24.

Stringfunktionen

Mithilfe von Stringfunktionen können Sie prüfen, ob in einer Zeichenkette eine bestimmte andere Zeichenkette enthalten ist, Teile aus einer Zeichenkette heraustrennen und so weiter, kurz – Zeichenketten auf vielfältigste Art und Weise manipulieren. Die drei am häufigsten verwendeten Stringfunktionen sind *Mid*, *Left* und *Right*:

```
v = Left(x$, n)
v = Right(x$, n)
v = Mid(x$, n [,m])
```



Einige Funktionen, die Zeichenfolgen liefern, gibt es in zwei Versionen: einmal als Funktion des Typs Variant und ein weiteres Mal als Funktion vom Typ String, die explizit eine Zeichenkette übergibt. Left, Right und Mid sind Beispiele für derartige Funktionen. Hängen Sie an den Funktionsnamen ein Dollarzeichen »\$« an (Left\$, Right\$, Mid\$), so wird der Datentyp String statt Variant zurückgegeben.

Left Die *Left*-Funktion übergibt die ersten *n* Zeichen der Zeichenkette *x\$*. Wie bei jeder Funktion kann das numerische Argument *n* eine Konstante (3, 7, 23), eine numerische Variable (*x*, *zahl%*) oder ein komplexer Ausdruck sein ($3 + x$, $7 * zahl\%$).

Der Stringausdruck *x\$* darf ebenfalls eine Stringkonstante (»Maier«, »Müller«), eine Stringvariable (*x\$*, *name\$*) oder ein komplexer Ausdruck sein (»Müller« + *x\$* oder *a\$* + *b\$* + *c\$*). Zum Beispiel gibt der Aufruf

```
Debug.Print Left("Hallo", 1)
```

das erste Zeichen der Zeichenkette »Hallo« im Direktfenster aus, also »H«, und der Aufruf

```
Debug.Print Left("Hallo", 2)
```

gibt entsprechend die ersten zwei Zeichen von »Hallo« aus, also »Ha«. Interessant wird es, wenn das Argument *n* größer ist als die Länge der Zeichenkette. Die Anweisung

```
Debug.Print Left("Hallo", 6)
```

soll VBA veranlassen, die ersten sechs Zeichen einer Zeichenkette auszugeben, die aus nur fünf Zeichen besteht. Das Resultat: Es werden einfach nur die fünf möglichen Zeichen ausgegeben, also »Hallo«.

Right *Right* wird mit den gleichen Argumenten aufgerufen wie *Left*, übergibt jedoch nicht den linken, sondern den rechten Teil der Zeichenkette *x\$* in der Länge *n*. Die Anweisung


```
Debug.Print Right("Hallo", 3)
```

gibt die letzten drei Zeichen von »Hallo« aus, »llo«.

Mid übergibt die ersten m Zeichen des Strings $x\$$ ab Position n . Zur Position ist zu sagen, dass das erste Zeichen in einem String die Nummer 1 besitzt, das Zeichen rechts davon die Nummer 2 und so weiter. Die Anweisung

```
Debug.Print Mid("Hallo", 3, 2)
```

übergibt die beiden Zeichen ($m = 2$), die sich ab dem dritten Zeichen ($n = 3$) im String »Hallo« befinden, also »ll«.

Angenommen, Sie speichern in der Variablen $a\$$ die Zeichenkette »Mein Personal Computer ist der Beste«. Welche Anweisung übergibt »Personal Computer«? Er übergibt eine *Mid*-Anweisung, in der als Argument $x\$$ die Variable $a\$$, als Argument n der Wert 6 (die betreffende Teilzeichenkette beginnt ab dem sechsten Zeichen von $a\$$) und als Argument m der Wert 17 übergeben wird (»Personal Computer« besteht aus 17 Zeichen):

```
a$ = "Mein Personal Computer ist der Beste"
Debug.Print Mid(a$, 6, 17)
```

Wenn das Argument m – die optionale Längenangabe – entfällt, übergibt *Mid* einfach den gesamten Rest der Zeichenkette ab der angegebenen Position n . Daher übergibt *Mid*(»Hallo«, 3) die Zeichenkette »llo«, alle Zeichen ab Position 3 bis zum Ende des Strings.

InStr ist eine numerische Funktion (fehlendes Dollarzeichen), die angibt, ob und – wenn ja – wo in einer Zeichenkette $x\$$ eine zweite Zeichenkette $y\$$ enthalten ist.

```
v = InStr([n,] x$, y$)
      |      |      |
      |      |      |
Suchstart  |      String, der gesucht wird
           |
           |
String, der
durchsucht wird
```

Ohne den optionalen Parameter n ist die Anwendung der *InStr*-Funktion einfach: *InStr*(»Hallo«, »al«) übergibt den numerischen Wert 2, da die Zeichenkette »al« ab Position Nummer 2 im String »Hallo« enthalten ist. Und *InStr*(»Dies ist ein Test«, »kein«) übergibt 0, da die Zeichenkette »kein« in »Dies ist ein Test« nicht enthalten ist (mit »ein« statt »kein« als gesuchter Zeichenkette übergibt *InStr* als Resultat 10).

Mit dem Parameter n können Sie festlegen, wo die Suche beginnt. Ohne diesen Parameter beginnt sie am Anfang des zu durchsuchenden Strings, gesucht wird also ab dem ersten Zeichen.

n ist eine beliebige Zahl zwischen 1 und der Länge des durchsuchten Strings. Zum Beispiel durchsucht `InStr(2, »Ein Test«, »Ein«)` die Zeichenkette »Ein Test« ab dem zweiten Zeichen nach »Ein« und übergibt daher 0.

Str Die Funktion `Str` übergibt einen String, der die einzelnen Ziffern der Zahl x enthält. Zum Beispiel weist die Anweisung

```
a$ = Str(23)
```

der Stringvariablen `a$` die Zeichenkette » 23« zu. Beachten Sie bitte das erste Zeichen dieses Strings, das Leerzeichen vor der eigentlichen Zahl!

33.6 Selbstdefinierte Prozeduren



Wie Sie wissen, erstellt EINFÜGEN|PROZEDUR... bzw. das zugehörige Symbol eine neue Prozedur (Abbildung 33.6).

Abbildung 33.6:
Prozedur erstellen

**Typ und Geltungs-
bereich**

Wie der eingefügte Prozedurrumpf aussieht, hängt von den ausgewählten Optionen ab. Die unter *Typ* gewählte Option bestimmt die Art der Prozedur: Je nach Option fügt VBA vor einem Prozedurnamen wie *Test* eines der drei Schlüsselwörter *Sub* (Sub-Prozedur), *Function* (Funktionsprozedur) oder *Property* (Eigenschaftsprozedur) ein.

Gleiches gilt für die unter *Gültigkeitsbereich* gewählte Option: *Public* fügt zusätzlich das Schlüsselwort *Public* ein, *Private* das Schlüsselwort *Private*.

Aktivieren Sie *Alle lokalen Variablen statisch*, wird darüber hinaus noch das Schlüsselwort *Static* eingefügt.

Prozedurdeklarationen können also recht komplex sein. Geben Sie Ihrer Prozedur den Namen *Test*, wählen Sie *Function* und *Private* und aktivieren Sie zusätzlich das Kontrollkästchen *Alle lokalen Variablen...*, dann lautet der eingefügte Prozedurrumpf

```
Private Static Function Test().
```

Damit wird eine private Funktionsprozedur namens *Test* deklariert, deren Variablen statisch sind. Was all das bedeutet, müssen Sie glücklicherweise jetzt noch nicht wissen. Die drei Schlüsselwörter *Public*, *Private* und *Static* sind alle optional, können also auch weggelassen werden. Die Angabe dieser Schlüsselwörter ist nicht verpflichtend.

Der Beweis dafür ist einfach zu führen: VBA fügt automatisch zumindest eines der beiden Schlüsselwörter *Public* oder *Private* ein; löschen Sie das betreffende Schlüsselwort anschließend, funktioniert die betreffende Prozedur genauso wie zuvor.

Ich will damit keinesfalls sagen, dass diese Schlüsselwörter bedeutungslos seien. Da die Schlüsselwörter Public, Private und Static aber zumindest nicht lebenswichtig für eine Prozedur sind, bespreche ich sie später, in Kapitel 33.7.1, »Prozeduren«. Auch auf Property werde ich erst später eingehen.



Das reduziert die Syntax einer Prozedur zunächst auf die beiden folgenden wesentlich harmloseren Varianten:

```
'Syntax einer Sub-Prozedur  
'-----
```

```
Sub Prozedurname [(Argumentliste)]  
    [Anweisung 1]  
    [Anweisung 2]  
    ...  
    ...  
    [Anweisung N]  
End Sub
```

```
'Syntax einer Funktionsprozedur  
'-----
```

```
Function Prozedurname [(Argumentliste)]  
    [Anweisung 1]  
    [Anweisung 2]  
    ...  
    ...
```

Listing 33.5:
Sub-Prozeduren

Listing 33.6:
Funktions-
prozeduren

Listing 33.6: [Prozedurname = Ausdruck]
 Funktions- ...
 prozeduren ...
 (Forts.) [Anweisung N]
 End Function

Beide Varianten, sowohl Sub-Prozeduren als auch Funktionsprozeduren, bestehen aus einem Prozedurrumpf, zwischen dem sich beliebig viele auszuführende Anweisungen befinden können. Der entscheidende Unterschied ist der zusätzliche Ausdruck *Prozedurname = Ausdruck*, der in einer Funktionsprozedur an einer beliebigen Stelle vorkommen darf.

Funktionsprozeduren Mit diesem Ausdruck wird in einer Funktionsprozedur der Funktionswert definiert, den die Funktion übergibt. Heißt die Funktion beispielsweise *Test*, weist ihr der Ausdruck *Test = 20* den Funktionswert 20 zu.

Funktionsprozeduren übergeben prinzipiell einen Funktionswert vom allgemeinen Datentyp *Variant*. Sie können den Funktionstyp, also den Datentyp des Funktionswerts, jedoch auch mit einem zusätzlichen *As Datentyp*-Ausdruck festlegen:

```
Function Prozedurname [(Argumentliste)] As Datentyp
```

Beispielsweise übergibt die Funktion

```
Function Test() As Double
```

einen *Double*-Wert als Funktionswert

oder alternativ dazu durch Angabe des zugehörigen Typkennzeichens hinter dem Funktionsnamen

```
Function Test#().
```

Sub-Prozeduren Sub-Prozeduren übergeben zwar keinen Funktionswert, besitzen aber dennoch ihren Sinn. Stellen Sie sich ein größeres Programm mit Unmengen an Funktionsprozeduren vor, die immer wieder irgendwelche Meldungen auf dem Bildschirm ausgeben müssen, entweder im Direktfenster oder in einem kleinen Dialogfeld (*MsgBox*).

Natürlich können Sie in jeder dieser Funktionsprozeduren immer wieder die gleichen dazu benötigten Anweisungen einzutippen. Allerdings ist es einfacher, für diese immer gleiche Aufgabe eine Sub-Prozedur zu erstellen, die die benötigten Ausgabeanweisungen enthält.

Immer dann, wenn eine der Funktionsprozeduren wieder mal die betreffenden Texte ausgeben muss, ruft sie dazu einfach mit der Anweisung *Call* die dafür zuständige Sub-Prozedur auf:

```
Call Prozedurname [(Argumentliste)]
```

Beispielsweise ruft

```
Call Aufgerufen
```

die Sub-Prozedur *Aufgerufen* auf.

Prozeduraufrufe

Enthält eine Funktionsprozedur diese Anweisung, verzweigt der Programmfluss zu dieser Prozedur, das heißt nun werden zunächst die in *Aufgerufen* enthaltenen Anweisungen ausgeführt. Ist das Ende der Prozedur erreicht, wird zur aufrufenden Prozedur zurückgekehrt und die Anweisung ausgeführt, die *Call* folgt (Abbildung 33.7).

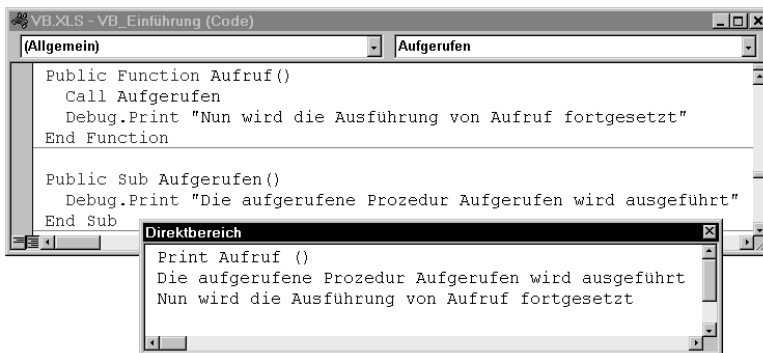


Abbildung 33.7:
Vb.xls, Modul
»VB_Einführung«,
Prozeduren »Aufruf«
und »Aufgerufen«

Die Funktion *Aufruf* wird im Direktfenster mit *Print Aufruf()* ausgeführt. Die erste darin enthaltene Anweisung *Call Aufgerufen* ruft die Sub-Prozedur *Aufgerufen* auf. Dadurch wird die Ausführung von *Aufruf* vorübergehend unterbrochen und nun die Sub-Prozedur *Aufgerufen* ausgeführt, die die Zeichenkette »Die aufgerufene Prozedur Aufgerufen wird ausgeführt« ausgibt.

End Sub beendet die Ausführung der Sub-Prozedur. VBA kehrt zur aufrufenden Funktionsprozedur *Aufruf* zurück und setzt deren Ausführung mit der folgenden Anweisung fort, mit

```
Debug.Print "Nun wird die Ausführung von Aufruf fortgesetzt".
```

Allgemein: Die Anweisung Call unterbricht die Ausführung einer Prozedur und führt die angegebene Prozedur aus. Ist deren Ende erreicht, wird zur aufrufenden Prozedur zurückgekehrt und dort die nächste Anweisung ausgeführt.



Call ist übrigens optional, das heißt, eine Prozedur namens *Aufgerufen* kann statt mit *Call Aufgerufen* ebenso gut einfach mit *Aufgerufen* ausgeführt werden (die Argumentliste darf dann jedoch nicht von Klammern umgeben sein).



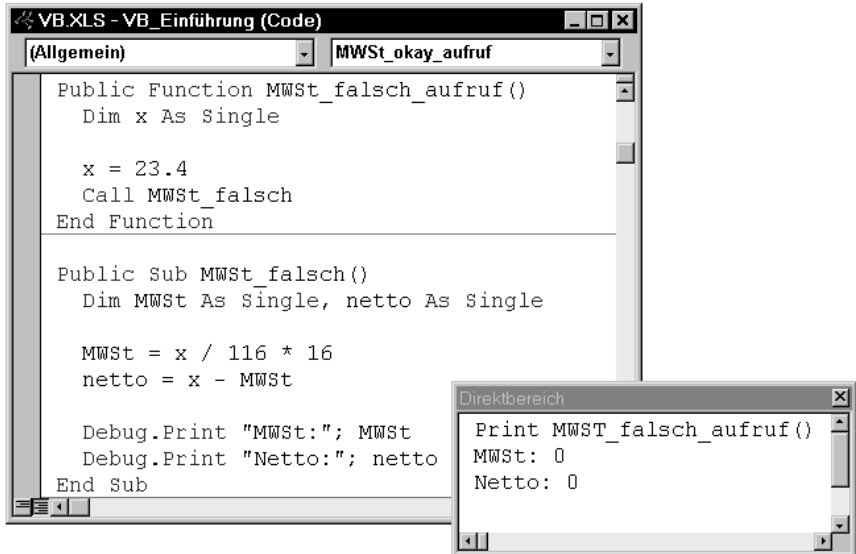
Denken Sie daran, dass Sie eine *Sub*-Prozedur zu Testzwecken statt mit *Call* auch ausführen können, indem Sie den Cursor in eine beliebige Prozedurzeile setzen und AUSFÜHREN|SUB|USERFORM AUSFÜHREN wählen oder auf das abgebildete Symbol klicken.

33.6.1 Lokale Variablen

Nun ein praktisches Beispiel: Eine Sub-Prozedur soll, basierend auf einer Zahl x , die darin enthaltene Umsatzsteuer und den Nettobetrag berechnen. Dazu muss die aufrufende Prozedur der aufgerufenen Sub-Prozedur mitteilen, welche Zahl für diese Berechnung verwendet werden soll.

Die einfachste Lösung scheint zunächst darin zu bestehen, die Zahl in einer Variablen x zu speichern, die die aufgerufene Sub-Prozedur anschließend weiterverwendet, etwa so (Abbildung 33.8):

Abbildung 33.8:
Vb.xls, Modul
»VB_Einführung«,
Prozeduren
»USt_falsch_aufruf«
und »USt_falsch«



USt_falsch_aufruf weist der *Single*-Variablen x den Bruttobetrag 23.4 zu und ruft *USt_falsch* auf. *USt_falsch* ermittelt mit $mwst = x / 116 * 16$ die in x enthaltene Umsatzsteuer und weist sie der *Single*-Variablen $mwst$ zu. Danach ermittelt $netto = x - mwst$ den Nettobetrag und speichert ihn in der *Single*-Variablen $netto$.

Debug.Print »UST:«; mwst gibt die Zeichenkette »USt« und dahinter den Inhalt der Variablen *mwst* aus, *Debug.Print »Netto:«; netto* entsprechend »Netto« und den Inhalt der Variablen *netto*.

Offenbar enthalten jedoch beide Variablen den Wert 0. Die Ursache dafür klärt ein kleiner Versuch (Abbildung 33.9).

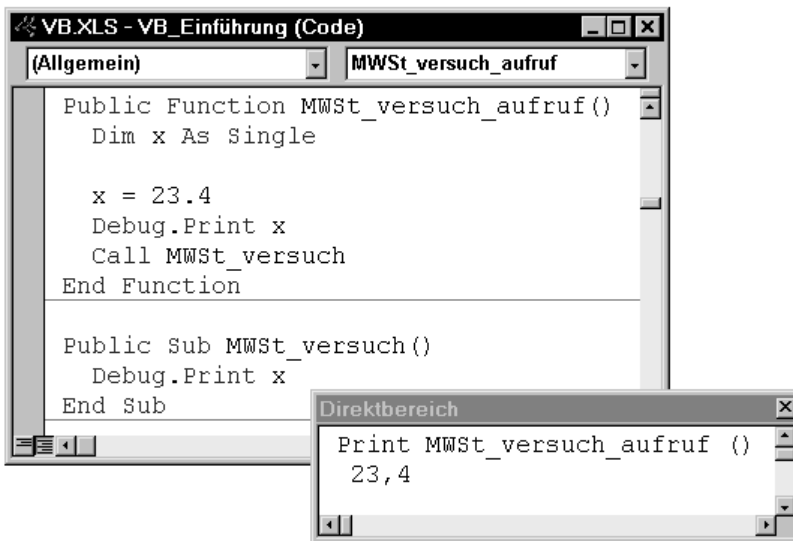


Abbildung 33.9:
Vb.xls, Modul
»VB_Einführung«,
Prozeduren
»USt_versuch_aufruf« und
»USt_versuch«

Der in *USt_versuch_aufruf* deklarierten Variablen *x* wurde der Wert 23.4 zugewiesen, den *USt_versuch_aufruf* auch korrekt ausgibt. Anschließend wird *USt_versuch* aufgerufen. Dort gibt *Debug.Print x* statt 23.4 jedoch überhaupt nichts aus. Offenbar ist die Variable *x* für die Prozedur *USt_versuch* nicht existent! Damit kennen wir nun auch die Ursache der fehlgeschlagenen Umsatzsteuerberechnung:

Eine Variable ist immer lokal zur Prozedur, in der sie deklariert wurde, und nur dieser Prozedur und keiner anderen bekannt! Für andere Prozeduren ist die betreffende Variable nicht existent. Tatsächlich existiert sie nur während der Ausführung der betreffenden Prozedur und wird danach wieder gelöscht.

Sie können daher ohne die geringsten Probleme in beliebig vielen Prozeduren gleichnamige Variablen deklarieren und verwenden, ohne dass sich diese Variablen gegenseitig beeinflussen.

Jede der Prozeduren sieht nur ihre eigene Variable *x* und weiß nichts von der Existenz einer gleichnamigen Variablen in einer anderen Prozedur.



Diese Eigenschaft von Variablen ist vor allem in großen und komplexen Programmen ein enormer Vorteil, da dadurch versehentliche Veränderungen von Variablen durch die Verwendung gleichnamiger Variablen in anderen Prozeduren von vornherein ausgeschlossen sind.

33.6.2 Übergabe von Argumenten

Sie ist jedoch von Nachteil, wenn wie bei der geplanten Umsatzsteuerberechnung der Inhalt einer Variablen ganz bewusst an eine andere Prozedur übergeben werden soll. Glücklicherweise stellt VBA für die Variablenübergabe einen eigenen Mechanismus zur Verfügung.

Argumentliste Schauen Sie sich noch einmal die vollständige Syntax der *Function* und der *Sub*-Anweisung an:

```
Function Prozedurname [(Argumentliste)]
```

```
Sub Prozedurname [(Argumentliste)]
```

Argumentliste ist eine Anzahl von Variablen, jeweils durch ein Komma getrennt. In diesen Variablen speichert die aufgerufene Prozedur die von der aufrufenden Prozedur übergebenen Werte. Aus der Liste muss eindeutig hervorgehen, welchen Datentyp die übergebene Information besitzt.

Zum Beispiel wird der Prozedur

```
Sub Test (x!)
```

ein *Single*-Wert übergeben, den diese Prozedur in der Variablen *x!* speichert. *x!* ist die lokale Prozedurvariable, die die übergebene Information aufnimmt.

As Alternativ dazu können Sie den Typ der übergebenen Variablen ähnlich wie in der *Dim*-Anweisung mit dem Zusatz *As Typ* bekannt geben:

```
Sub Test (x As Single)
```

Diese Methode werde ich bevorzugen, da sie klarer ist und eher allgemeinen Standards entspricht, die beispielsweise auch für Pascal oder C gelten.

Damit ist unser Umsatzsteuerproblem gelöst (Abbildung 33.10).

Der Aufruf *Call USt_okay(23.4)* übergibt der Prozedur *USt_okay* den Wert 23.4, den diese in der lokalen Variablen *brutto* übernimmt, die anschließend zur Ermittlung der darin enthaltenen Umsatzsteuer und des zugehörigen Nettobetrags benutzt wird.

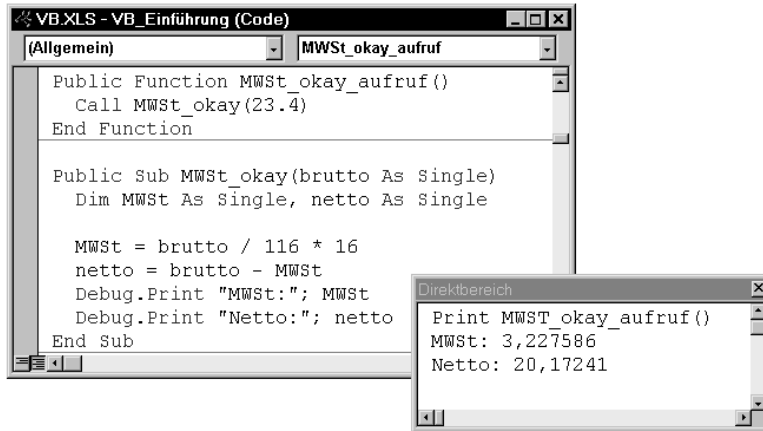


Abbildung 33.10:

Vb.xls, Modul
»VB_Einführung«,
Prozeduren
»USt_okay_aufruf«
und »USt_okay«

Sie können einer Prozedur beliebig viele Argumente übergeben. Die folgende Prozedur erwartet beispielsweise die Übergabe zweier Zahlen:

```
Sub Testproc(Einkaufspreis As Single, Verkaufspreis As Single)
    Debug.Print "Gewinn:", Verkaufspreis - Einkaufspreis
End Sub
```

Um sie aufzurufen, verwenden Sie einen Ausdruck wie

```
Call Testproc(200, 300)
```

Nach dem Aufruf befindet sich in der Variablen *Einkaufspreis* der Prozedur der Wert 200 und in der Variablen *Verkaufspreis* der Wert 300. Die Prozedur verwendet diese Variablen, um den Gewinn zu ermitteln und im Direktfenster auszugeben.

Die Übergabe von Konstanten ist der einfachste Fall. Sie können einer Prozedur beliebige Ausdrücke übergeben, Konstanten, Variablen oder Mischungen von beidem. Zum Beispiel übergibt der Aufruf

```
Call USt_okay(20 + 3.4)
```

der Prozedur *USt_okay* ebenfalls die Zahl 23.4, nur etwas umständlicher formuliert. Auch

```
x = 23.4
Call USt_okay(x)
```

übergibt diese Werte, ebenso wie

```
x = 20
Call USt_okay(x + 3.4)
```

**Konstanten
übergeben**

**Zeichenketten
übergeben**

Natürlich können Sie auch Stringargumente übergeben. Der Aufruf

```
Call Stringtest ("Hallo")
```

übergibt einer Prozedur *Stringtest* die Zeichenkette »Hallo«, ebenso wie der Aufruf

```
a$ = "Hallo"  
Call Stringtest (a$)
```

Entsprechend muss in der Deklaration von *Stringtest* die Übergabe einer Stringvariablen vorgesehen werden, in der diese Zeichenkette bei der Übergabe gespeichert wird:

```
Sub Stringtest (s$)
```

oder

```
Sub Stringtest (s As String)
```

Übergabeausdrücke dürfen beliebig komplex sein:

```
Call Stringtest(a$ + "x" + b$ + "y")
```

Hier wird der Prozedur *Stringtest* eine Zeichenkette übergeben, die durch die Verkettung mehrerer Stringvariablen und Stringkonstanten entsteht.

33.6.3 Benannte und optionale Argumente

Wenn Sie wollen, brauchen Sie sich beim Aufruf einer Prozedur nicht unbedingt an die in der Deklaration festgelegte Reihenfolge zu halten. Angenommen, eine Prozedur erwartet zwei Zahlen:

```
Sub Testproc(Einkaufspreis As Single, Verkaufspreis As Single)
```

Normalerweise rufen Sie diese Prozedur etwa so auf:

```
Call Testproc(200, 300)
```

Dann ist 200 der übergebene Einkaufspreis und 300 der übergebene Verkaufspreis.

**Benannte
Argumente**

Wenn Sie wollen, können Sie die Reihenfolge der Argumente bei der Übergabe vertauschen. Damit VBA weiß, welche der beiden Zahlen in der Prozedurvariablen *Einkaufspreis* gespeichert werden soll und welche in der Prozedur-

variablen *Verkaufspreis*, müssen Sie die Argumente jedoch entsprechend benennen und zum Aufruf folgende Syntax *ohne Call* und ohne Klammern verwenden:

```
Prozedurname Variablenname1:=Wert, Variablenname2=Wert, ...
```

Angewandt auf das Beispiel heißt das:

```
Testproc Verkaufspreis:=300, Einkaufspreis:=200
```

Sollen Argumente optional sein, beim Aufruf also nicht unbedingt angegeben werden müssen, stellen Sie ihnen das Schlüsselwort *Optional* voran. Beachten Sie dabei, dass optionale Argumente vom Typ *Variant* sein müssen!

*Optionale
Argumente*

```
Sub Testproc(Optional Einkaufspreis As Variant, Optional Verkaufspreis As Variant)
```

Wenn Sie wollen, können Sie beim Aufruf nur das zweite Argument angeben und für das erste Argument als Stellvertreter einfach ein Komma einsetzen:

```
Call Testproc(, 300)
```

Wenn Sie das Schlüsselwort *Optional* einsetzen, müssen Sie es übrigens gleich *allen* Argumenten voranstellen.

Die aufgerufene Prozedur kann mit der Funktion IsMissing und der in Kürze erläuterten If-Anweisung ermitteln, ob ein optionales Argument übergeben oder ausgelassen wurde. Beispielsweise gibt



```
If Not IsMissing(Einkaufspreis) Then Debug.Print Einkaufspreis
```

nur dann den Einkaufspreis im Direktfenster aus, wenn diese Variable auch tatsächlich übergeben wurde.

33.6.4 Als Referenz

Wie Sie vor kurzem sahen, kann beim Aufruf einer Prozedur auch eine Variable übergeben werden. Dann ist das Lokalitätsprinzip jedoch durchbrochen: Wird jene Variablen in der Argumentliste der aufgerufenen Prozedur verändert, in der das Argument übernommen wird, wird dadurch auch die korrespondierende Variable der aufrufenden Prozedur verändert.

Nehmen Sie als Beispiel folgende Prozedur *Testproc*:

```
Sub Testproc(zahl)
    zahl = zahl + 10
End Sub
```

Der Aufruf aus einer anderen Prozedur heraus mit

```
x = 50
Call Testproc(x)
Debug.Print x
```

übergibt *Testproc* in der Prozedurvariablen *zahl* den Wert 50. *Testproc* addiert 10 und weist das Resultat 60 wieder *zahl* zu.

Diese Zuweisung betrifft jedoch gleichzeitig die korrespondierende Variable *x* der aufrufenden Prozedur, sodass *Debug.Print x* nun entsprechend den Wert 60 ausgibt und damit beweist, dass *x* jetzt den Wert 60 enthält, diese Variable also durch die aufgerufene Prozedur beeinflusst wurde.

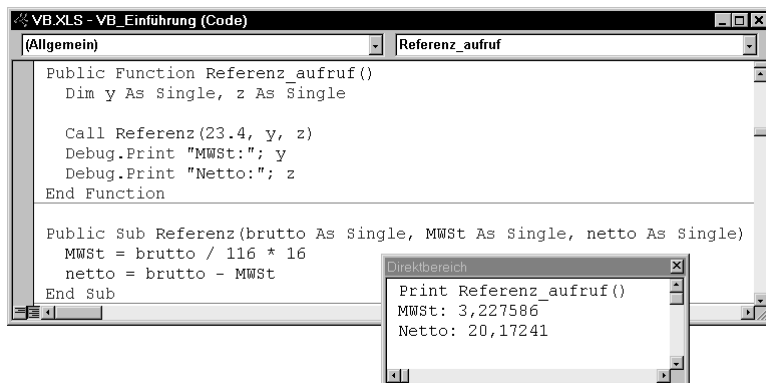
Diesen Mechanismus nennt man Variablenübergabe als Referenz. Er kann eingesetzt werden, um es einer aufgerufenen Prozedur zu ermöglichen, an die aufrufende Prozedur Informationen *zurückzugeben*.

Angenommen, eine Prozedur soll zwar die Umsatzsteuer und den Bruttobetrag ermitteln, die Ergebnisse jedoch nicht auf dem Bildschirm ausdrucken, sondern nur der aufrufenden Prozedur zurückübergeben. Dazu übergeben Sie beim Aufruf eine Variable. Jede Veränderung der zugehörigen Variablen in der aufgerufenen Prozedur wirkt sich in identischer Weise auf die beim Aufruf angegebene Variable aus.

Soll die aufgerufene Prozedur wie *Umsatzsteuer* zwei Informationen zurückgeben, müssen ihr entsprechend zwei Variablen übergeben werden (Abbildung 33.11).

Abbildung 33.11:

Vb.xls,
»VB_Einführung«,
Prozeduren
»Referenz_aufruf«
und »Referenz«



Referenz_aufruf übergibt *Referenz* außer dem Bruttobetrag 23.4 die beiden zuvor deklarierten *Single*-Variablen *y* und *z*, die in diesem Moment keinerlei definierten Wert enthalten.

Jede Änderung einer der in der Argumentliste von *Referenz* aufgeführten Variablen wirkt auf die korrespondierende Variable der aufrufenden Prozedur zurück: Jede Veränderung von *mwst* führt zur gleichen Veränderung bei *y* und jede Veränderung von *netto* zur entsprechenden Beeinflussung von *z*.

Die Zuweisung $mwst = brutto / 116 * 16$ weist daher nicht nur der Variablen *mwst* von *Referenz* einen Wert zu, sondern gleichzeitig der korrespondierenden Variablen *y* der Prozedur *Referenz_aufruf*. Und jede Veränderung von *netto* weist den betreffenden Wert entsprechend der zugehörigen Variablen *z* der Prozedur *Referenz_aufruf* zu.

Allgemein: Wird beim Aufruf einer Prozedur in der Argumentliste eine Variable verwendet, verändert jede Zuweisung an die zugehörige Variable der Parameterliste diese Variable des Aufrufers.



Für die übernehmende Prozedurvariable muss nicht unbedingt wie im Beispiel ein anderer Name als für die übergebene Variable verwendet werden. Der Name ist bedeutungslos. Es spielt keine Rolle, ob Sie der übergebenen Variablen *y* in der aufgerufenen Prozedur eine Variable namens *mwst*, *xyz* oder einfach eine gleichnamige Variable *y* zuordnen.

VBA interessiert sich ausschließlich für die *Reihenfolge* in der Argument- und der Parameterliste. Wird in einer aufgerufenen Prozedur die zweite angegebene Variable beeinflusst, hat das Auswirkungen auf die zweite beim Aufruf übergebene Variable, unabhängig von den verwendeten Variablennamen.

Um die Übergabe als Referenz ausdrücklich festzulegen (und Ihre Programme lesbarer zu machen), können Sie das Schlüsselwort *ByRef* verwenden:

ByRef

```
Sub Testproc(ByRef zahl As Single)
```

33.6.5 Als Wert

Die Übergabe einer Variablen ermöglicht der aufgerufenen Prozedur, diese Variable zu verändern und auf diese Weise Informationen zurückzugeben.

Geschieht das bewusst, ist alles in Ordnung. Allerdings ermöglicht dieser Mechanismus auch ungewollte Veränderungen, da die lokale Abschottung der Variablen verschiedener Prozeduren durchbrochen wird. Ruft eine Prozedur eine andere auf und übergibt ihr eine Variable, die diese unbeabsichtigt verändert, sind ungewollte Nebenwirkungen unvermeidlich.

Um derartige Effekte zu vermeiden, kann eine Variable als Wert übergeben werden, indem sie beim Aufruf in Klammern gesetzt wird:

```
Call Wert((x))
```

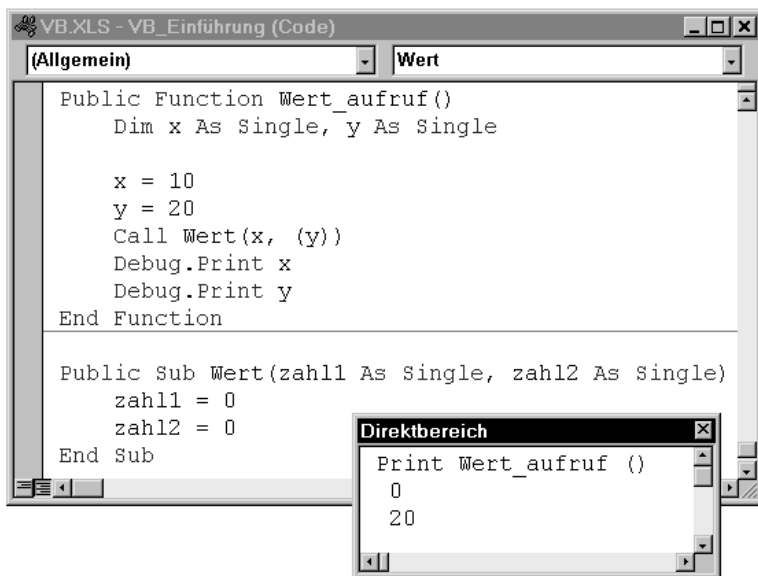
Nehmen wir an, *x* ist eine *Single*-Variable. Dann muss im Prozedurkopf von *Wert* ebenfalls eine *Single*-Variable deklariert werden:

```
Sub Wert (zahl As Single)
```

Wird in der Prozedur *Wert* der Inhalt von *x* verändert, zum Beispiel durch die Zuweisung *zahl* = 10, dann bleibt die Variable *x* der aufrufenden Prozedur diesmal unbeeinflusst. Die Klammerung schützt *x* vor jeder Veränderung durch die aufgerufene Prozedur. *Wert* kann zwar lesend, aber nicht verändernd auf diese Variable zugreifen.

Ein weiteres Beispiel (Abbildung 33.12) sehen Sie hier:

Abbildung 33.12:
Referenz- und
Wert-Übergabe



In diesem Beispiel wird *x* der Prozedur *Wert* von *Wert_aufruf* als Referenz übergeben (keine Klammern), *y* dagegen als Wert (Klammern).

Die korrespondierenden Variablen von *Wert* heißen *zahl1* und *zahl2*. Bei-
den wird der Wert 0 zugewiesen. Die als Referenz übergebene und daher
nicht geschützte korrespondierende Variable *x* der aufrufenden Prozedur
wird dadurch verändert, die als Wert übergebene Variable *y* jedoch nicht.

33.7 Geltungsbereiche

33.7.1 Prozeduren

Die vollständige Syntax der *Sub*- und *Function*-Anweisungen lautet

```
[Public | Private] [Static] Sub Prozedurname [(Argumentliste)]  
[Public | Private] [Static] Function Prozedurname [(Argumentliste)] As Typ
```

Public (*Public Sub Test()*) deklariert öffentliche Prozeduren. Das heißt diese Prozedur kann von allen Prozeduren aus aufgerufen werden, egal in welchen Modulen sie sich befinden.

Public

Prozeduren sind auch ohne dieses Schlüsselwort öffentlich, sodass Sie es auch weglassen können. Ausnahme: Ereignisprozeduren sind standardmäßig privat: in Deklarationen von Ereignisprozeduren fügt VBA automatisch das Schlüsselwort *Private* ein.

Private (*Private Sub Test()*) deklariert private Prozeduren. Das bedeutet, dass die betreffende Prozedur nur von Prozeduren aus aufgerufen werden kann, die sich *im gleichen Modul* befinden.

Private

Egal ob Sie eine Prozedur als *Public* oder als *Private* deklarieren, in jedem Fall dürfen Sie das Schlüsselwort *Static* hinzufügen (*Public Static Sub Test()* oder *Private Static Sub Test()*). Es bewirkt, dass die Inhalte aller lokalen Prozedurvariablen bis zum nächsten Aufruf der Prozedur unverändert erhalten bleiben und nicht wie sonst nach dem Beenden der Prozedur gelöscht werden.

Static

Dieses Schlüsselwort ermöglicht Prozeduren wie die folgende (Abbildung 33.13):

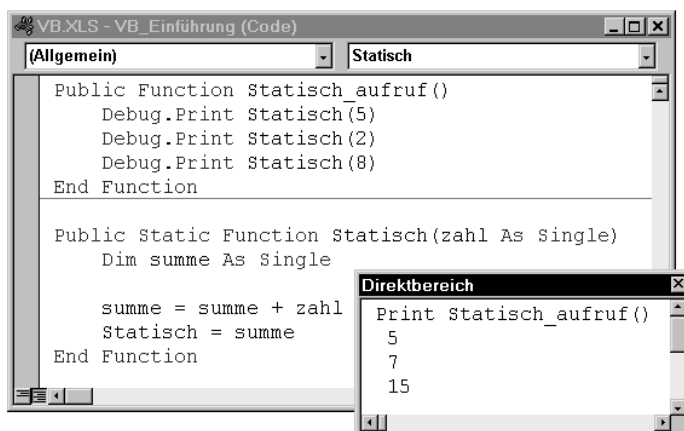


Abbildung 33.13:
Statische
Prozeduren

Der Funktionsprozedur *Statisch* wird ein *Single*-Argument übergeben, das sie zur aktuellen Zwischensumme *Summe* addiert, einer *Single*-Variablen. Das neue Zwischenergebnis wird als Funktionswert übergeben.

Dank des Schlüsselworts *Static* bleibt der aktuelle Inhalt von *Summe* zwischen den einzelnen Aufrufen weiterhin erhalten. Ohne dieses Schlüsselwort würde *Summe* stattdessen jedesmal nach dem Verlassen der Prozedur gelöscht und beim nächsten Aufruf neu angelegt und mit 0 initialisiert werden, sodass statt 5, 7 und 15 einfach die übergebenen Zahlen 5, 2 und 8 ausgegeben würden.



Das Schlüsselwort Static kann mit Private oder Public kombiniert werden. Beispielsweise deklariert

```
Private Static Sub Demo()
```

eine statische Prozedur *Demo*, die nur von den Prozeduren jenes Moduls verwendet werden kann, in dem sie deklariert ist.

33.7.2 Variablen und Konstanten

Prinzipiell sind Prozedurvariablen lokal, das heißt der Geltungsbereich einer Variablen ist auf die Prozedur beschränkt, in der sie deklariert wurde.

Zusätzlich können Variablen jedoch auch öffentlich oder privat sein. Ebenso wie bei Prozeduren bedeutet *Private*, dass alle Prozeduren des betreffenden Moduls die Variable benutzen können, und *Public* bedeutet, dass wirklich alle Prozeduren die Variable nutzen können, auch wenn sich die betreffenden Prozeduren in einem anderen Modul befinden.

Ein Beispiel: Möglicherweise soll ein und dieselbe Variable in einem umfangreichen Programm in vielen verschiedenen Prozeduren verwendet werden, beispielsweise eine Variable *USt*, der mit

```
USt = 16
```

der aktuelle Umsatzsteuersatz zugewiesen wird. Natürlich können Sie diese Variable beim Aufruf der verschiedenen Prozeduren immer wieder explizit übergeben:

Listing 33.7:
Wiederholte
Variablenübergabe

```
Call Prozedur1 (USt)
...
...
Call Prozedur2 (USt)
...
...
Call ProzedurN (USt)
```


Sie können sich diese Arbeit jedoch ersparen, wenn Sie die Variable als privat deklarieren, also als Variable, die im Gegensatz zu lokalen Variablen nicht nur einer, sondern allen Prozeduren des betreffenden Moduls zur Verfügung steht. Entscheidend dafür ist der Deklarationsort.

Um eine private Variable zu deklarieren, muss die Deklaration mit *Dim* außerhalb der Prozeduren erfolgen, im Deklarationsbereich des Moduls.

Dazu wählen Sie im *Prozeduren*-Listenfeld den Eintrag (*Deklarationen*). Im zugehörigen Abschnitt deklarieren Sie die Variable beispielsweise mit *Dim USt_privat As Integer* (Abbildung 33.14).

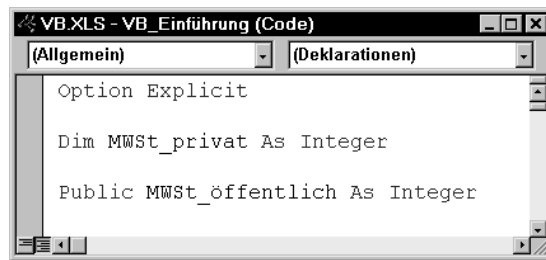


Abbildung 33.14:
Variablen deklarieren im Deklarationsabschnitt eines Moduls

Die Variable *USt_privat* steht nun allen Prozeduren des betreffenden Moduls zur Verfügung.

Deklarieren Sie eine Variable entsprechend der vorhergehenden Abbildung im Deklarationsbereich statt mit *Dim* mit dem Schlüsselwort *Public*, so wird sie dadurch zur öffentlichen Variablen und steht allen Prozeduren aller Module der Datenbank zur Verfügung.

Public

Auf Modulebene (im Abschnitt (*Deklarationen*)) deklarierte Variablen behalten ihren Wert, bis der Befehl AUSFÜHREN|ZURÜCKSETZEN gewählt wird.

Das gleiche erreichen Sie für lokale Variablen, die Sie innerhalb einer Prozedur deklarieren, wenn Sie dazu das Schlüsselwort *Static* verwenden:

Static

Static Variable As Datentyp

Der Inhalt von auf diese Weise deklarierten statischen Variablen bleibt bis zum nächsten Aufruf der Prozedur unverändert erhalten.

Das Ganze ähnelt der Deklaration einer Prozedur mit dem Schlüsselwort Static (siehe vorhergehenden Abschnitt), im Gegensatz dazu werden jedoch alle anderen Prozedurvariablen, die nicht mit Static deklariert wurden, weiterhin bei Erreichen des Prozedurendes gelöscht.



Die Deklaration einer Variablen mit *Static* bezieht sich daher im Gegensatz zur Deklaration einer Prozedur mit *Static* nur auf die deklarierte Variable und nicht auf alle Prozedurvariablen gleichzeitig.

Const Für den Geltungsbereich von Konstanten gilt das gleiche wie bei Variablen: Werden sie mit *Const* am Anfang einer Prozedur deklariert, sind sie nur dieser einen Prozedur bekannt.

Deklarieren Sie Konstanten im Deklarationsbereich eines Moduls, sind diese Konstanten allen in diesem Modul enthaltenen Prozeduren bekannt und können von ihnen verwendet werden.

Public Deklarieren Sie die Konstanten im Deklarationsbereich mit *Public* statt mit *Const*, stehen sie jedem einzelnen Modul und allen darin enthaltenen Prozeduren zur Verfügung (Ausnahme: Klassenmodule).

Zum Beispiel deklariert

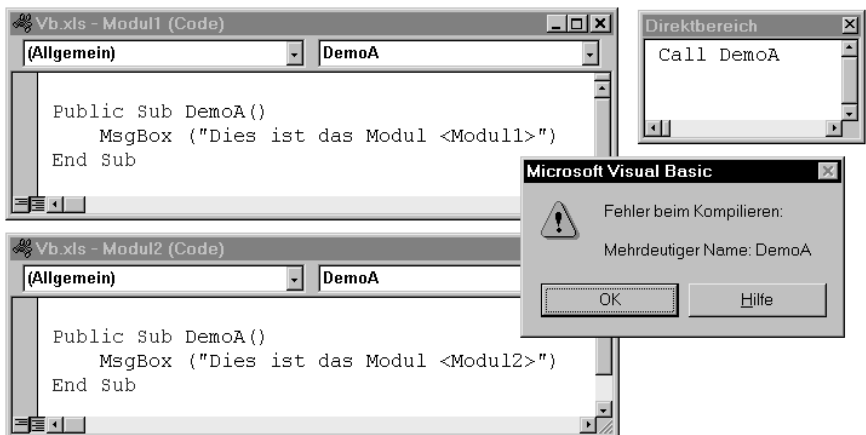
```
Public Const USt = 16
```

eine Konstante namens *USt*, die stellvertretend für die Zahl 16 steht und von allen Prozeduren Ihres VBA-Programms verwendet werden kann, egal in welchen Modulen sie sich befinden.

33.7.3 Bibliotheken und Prozedurbezüge

Öffentliche Prozeduren – Prozeduren ohne den Zusatz *Private* – können von allen Prozeduren aller Module aufgerufen werden. Trotzdem kann es beim Aufruf ein Problem geben (Abbildung 33.15).

Abbildung 33.15:
Mehrdeutiger
Prozedurname



Im Direktfenster wird die Prozedur *DemoA* aufgerufen. Für Excel ist der Prozedurname *DemoA* jedoch nicht eindeutig, da es zwei Prozeduren mit diesem Namen gibt: sowohl das Modul »Modul1« als auch das »Modul2« enthalten jeweils eine Prozedur *DemoA*.

Sie lösen dieses Problem durch einen Prozeduraufruf der Form

```
Call [Modulname].Prozedurname
```

Modulname ist der Name des Moduls, in dem sich die aufzurufende Prozedur befindet (Abbildung 33.16).

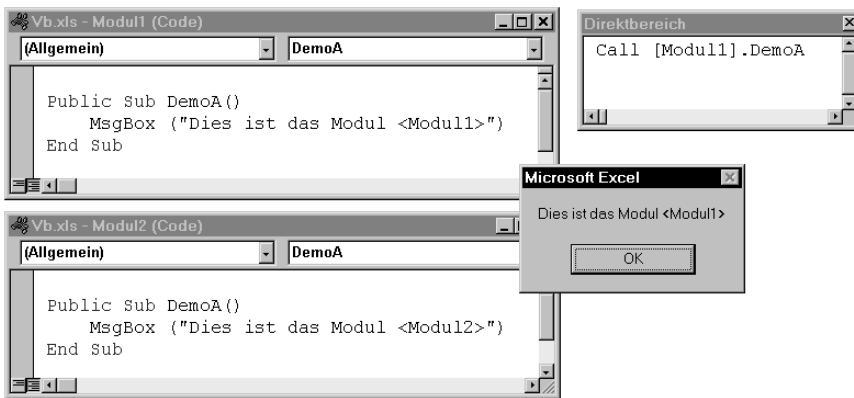


Abbildung 33.16:
Externer Prozedur-
bezug

Der Aufruf

```
Call [Modul1].DemoA
```

bezieht sich eindeutig auf die in »Modul1« enthaltene Prozedur *DemoA*.

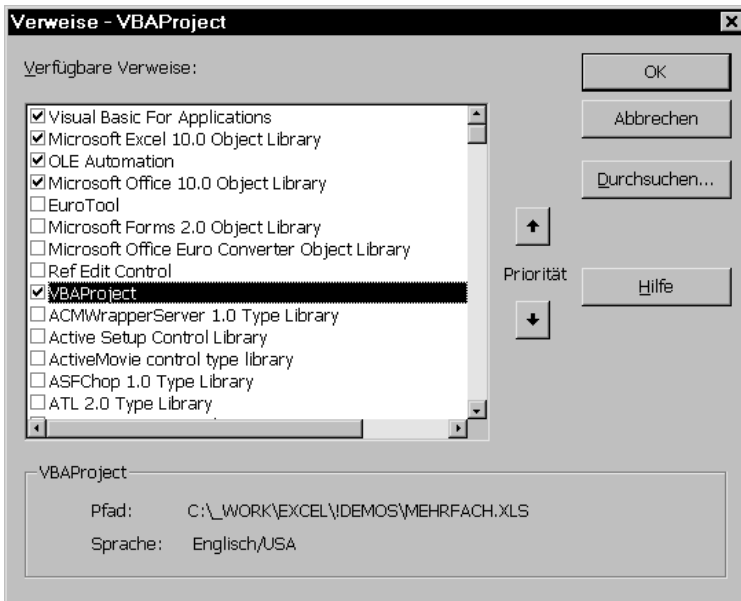
Sie können sogar Prozeduren aufrufen, die sich nicht in der gleichen, sondern in anderen Arbeitsmappen befinden.

*Bezüge auf
externe Bezüge*

Dazu müssen Sie mit dem Befehl EXTRAS|VERWEISE... einen Verweis auf diese Mappe einfügen, das heißt die betreffende Mappe als zusätzliche Bibliothek bekannt geben. Anschließend können öffentliche Variablen und Prozeduren dieser Bibliothek auch von der aktuellen Mappe benutzt werden (Abbildung 33.17).

Dieses Dialogfeld enthält immer aktive Verweise auf die Standard-Bibliotheken, die in jedem Projekt benutzt werden können. Zusätzlich befinden sich darin inaktive Verweise auf alle möglichen anderen Objekt-Bibliotheken. Um einen dieser Verweise zu aktivieren, aktivieren Sie einfach das zugehörige Kontrollkästchen.

Abbildung 33.17:
Verweise auf
Bibliotheken



VBA-Projekte

Diese Bibliotheken können auch aus VBA-Projekten bestehen, das heißt aus den Prozeduren, die eine gespeicherte Excel-Mappe enthält. Sie klicken dazu auf *Durchsuchen...*: Das *Dateiauswahl*-Dialogfeld erscheint, Sie lassen sich darin alle Excel-Dateien anzeigen und wählen die Mappe aus, die das betreffende Projekt enthält, beispielsweise MAPPE2.XLS. Nach *Öffnen* erscheint im übergeordneten Listenfeld ein zusätzlicher Eintrag *VBAProject*. Aktivieren Sie ihn, so wird das betreffende Projekt von VBA in seine Liste zu durchsuchender Bibliotheken aufgenommen. Enthält die Mappe eine Prozedur wie *DemoA*, wird daher ab jetzt auch in dieser Mappe nach der betreffenden Prozedur gefahndet.

Allgemein ausgedrückt: Alle im betreffenden Projekt enthaltenen öffentlichen Prozeduren und Variablen können ab jetzt auch vom aktuellen Projekt benutzt werden.

Ist im Listenfeld ein Eintrag selektiert, erscheint übrigens unterhalb des Listenfelds eine Kurzbeschreibung des Verweises, beispielsweise der Name der zugehörigen Arbeitsmappe. Dadurch wird der Durchblick bei mehreren auf diese Weise eingefügten Verweisen, die allesamt *VBAProject* heißen, deutlich erleichtert.

Mit den beiden nach oben bzw. nach unten weisenden Pfeilen geben Sie die Priorität einer Bibliothek an. Rufen Sie eine Prozedur wie *DemoA* auf, muss VBA alle Bibliotheken nach einer dieser Prozeduren/Variablen absuchen, wobei sich VBA bei der Suche an die Reihenfolge der Bibliotheken in diesem Listenfeld hält.

Verwenden Sie Objekte einer bestimmten Bibliothek sehr häufig, können Sie die Suche nach diesen Objekten beschleunigen, indem Sie der betreffenden Bibliothek eine höhere Priorität geben, beispielsweise, indem Sie den Eintrag *VBAProject* selektieren, der für MAPPE2.MDB steht, und diesen Eintrag mit dem nach oben weisenden Pfeil ein wenig nach oben verschieben.

Prozeduren, die sich in einem Klassenmodul befinden, können nicht von anderen Projekten benutzt werden! Das gleiche gilt für Variablen, die in Klassenmodulen deklariert werden!

Klassenmodule

33.8 Arrays (Datenfelder)

Datenfelder oder Arrays sind untrennbar mit den erst später erläuterten Schleifenstrukturen verbunden. Dieser Abschnitt ist daher vorwiegend für Leser mit Programmiererfahrung gedacht, die wissen, wozu Arrays eingesetzt werden, auf entsprechende Beispiele verzichten können und an den Deklarations- und Anwendungsregeln interessiert sind.

Leser ohne Programmiererfahrung muss ich damit vertrösten, dass sie erst im Laufe der Zeit die Nützlichkeit von Arrays entdecken werden.

Ein Array ist eine Gruppe von Variablen gleichen Datentyps. Alle Elemente (Variablen) einer solchen Gruppe besitzen prinzipiell den gleichen Variablennamen und unterscheiden sich nur durch einen zusätzlichen Index, eine ganze Zahl, die die Nummer des Elements angibt. Der Zugriff auf eine Array-Variable erfolgt mit einem Ausdruck der Form

`Variablenname(Index)`

Variablenname ist ein nach den üblichen Regeln gebildeter Name, *Index* die erwähnte ganze Zahl. Zum Beispiel weist

Index

`x(5) = 100`

der Variablen Nummer 5 des Arrays *x()* den Wert 100 zu. Bei Verwendung eines anderen Index wird entsprechend ein anderes Element der Gruppe angesprochen.

Der Index kann nicht nur eine Konstante, sondern auch eine Variable sein:

`x(zahl) = 100`

Hier wird einer Variablen des Arrays *x()* der Wert 100 zugewiesen. Welcher Variablen, wissen wir erst, wenn wir den aktuellen Wert von *zahl* kennen.

`zahl = 5`

`x(zahl) = 100`

Dieser Fall ist eindeutig. *zahl* enthält eine 5. Also wird in der folgenden Zuweisung das Element Nummer 5 des Arrays *x()* angesprochen.

Arrays gestatten einen sehr flexiblen indizierten Zugriff auf Variablen. Flexibel, da für die Elementnummer – den Index – selbst eine Variable verwendet werden kann.

Den Sinn dieser Variablen macht ein kleines Beispiel deutlich. Stellen Sie sich vor, ein Programm soll die Zahlen 1, 2, 3, ..., 100 in 100 einzelnen Variablen speichern. Ohne Zuhilfenahme von Arrays fällt mir nur folgende höchst triviale Lösung ein:

Listing 33.8:
Lösung ohne Arrays

```
Sub Test ()
    Dim x1 As Integer
    Dim x2 As Integer
    Dim x3 As Integer
    ...
    Dim x100 As Integer

    x1 = 1
    x2 = 2
    x3 = 3
    ...
    x100 = 100
End Sub
```

Die drei Pünktchen »...« stehen jeweils stellvertretend für 96 *gleichartige Anweisungen*, die an der betreffenden Stelle einzufügen sind. Mit Arrays würden Sie die Prozedur anders formulieren:

Listing 33.9:
Lösung mit Arrays

```
Sub Test()
    Dim x(1 To 100)
    Dim i As Integer

    For i = 1 To 100
        x(i) = i
        Debug.Print x(i)
    Next i
End Sub
```

Dieses Programm werden Sie ohne Programmiererfahrung erst nach der Besprechung von Schleifen verstehen. Es soll nur eines verdeutlichen: Schleifen, kombiniert mit Arrays, ersparen Ihnen häufig Hunderte, wenn nicht sogar Tausende von Programmzeilen.

Datenfeld- Deklaration

Arrays können Sie mit *Dim*, *Static*, *Private*, *Public* oder *ReDim* deklarieren:

```
Dim Variablenname(Indizes) As Typ
ReDim Variablenname(Indizes) As Typ
```

```
Public Variablenname(Indizes) As Typ  
Private Variablenname(Indizes) As Typ  
Static Variablenname(Indizes) As Typ
```

Dim wird auf Prozedurebene verwendet und deklariert in einer Prozedur ein lokales Datenfeld, das beim Verlassen der Prozedur wie üblich gelöscht wird.

ReDim ist ebenfalls auf Prozedurebene im Einsatz und kann verwendet werden, um die Größe eines darin zuvor mit *Dim* ohne Indexangaben deklarierten Datenfelds nachträglich neu zu definieren.

Static deklariert auf Prozedurebene ein statisches Datenfeld, dessen Inhalt bis zum nächsten Aufruf der Prozedur unverändert bleibt.

Außerhalb einer Prozedur, also auf Modulebene, mit *Private* deklarierte Datenfelder sind privat verfügbar und können von allen Prozeduren dieses Moduls uneingeschränkt verwendet werden. Auf Modulebene mit *Public* deklarierte Datenfelder sind öffentlich und damit allen Prozeduren aller Module bekannt.

Indizes ist im einfachsten Fall die Anzahl der benötigten Variablen, wobei VBA der ersten Variablen normalerweise die Elementnummer 0 gibt. Die Anweisung

```
Dim x(10) As Single
```

deklariert daher ein Datenfeld mit zehn *Single*-Variablen $x(0)$, $x(1)$, $x(2)$, ..., $x(9)$.

Die Array-Untergrenze 0 können Sie mit der auf Modulebene vor allen Prozeduren eingefügten Anweisung

Datenfeldgrenzen

```
Option Base {0|1}
```

verändern. Fügen Sie in den Deklarationsbereich die Anweisung *Option Base 1* ein, so deklariert die Anweisung *Dim x(10) As Single* zwar immer noch zehn *Single*-Variablen, aufgrund der geänderten Standarduntergrenze 1 nun jedoch die Variablen $x(1)$, $x(2)$, $x(3)$, ..., $x(10)$.

Es ist allerdings aussagekräftiger, bei der Deklaration zusätzlich das Schlüsselwort *To* einzusetzen und für *Indizes* einen Ausdruck der Art *Untergrenze To Obergrenze* zu verwenden; beispielsweise mit

```
Dim x(5 To 8) As Single
```

die vier *Single*-Variablen $x(5)$, $x(6)$, $x(7)$ und $x(8)$ zu deklarieren.

**Datenfelder
übergeben**

Um einer Prozedur eine Variable eines Datenfelds zu übergeben, verwenden Sie einen Ausdruck der Art

```
Call Prozedurname(Arrayname(Index))
```

Beispielsweise ruft

```
Call Test(x(3), x(8))
```

die Prozedur *Test* auf und übergibt ihr die Variablen $x(3)$ und $x(8)$. Der Aufruf

```
Call Test(x(3), (x(8)))
```

würde $x(8)$ dagegen nicht als Referenz, sondern als Wert übergeben.

Die Parameterliste von *Test* muss wie immer eine korrespondierende Variable vom gleichen Typ enthalten, zum Beispiel zwei *Single*-Variablen, wenn das Array $x()$ Variablen dieses Typs enthält:

```
Sub Test (zahl1 As Single, zahl2 As Single)
```

Jede Änderung von *zahl1* beeinflusst die als Referenz übergebene zugehörige Array-Variable $x(3)$. Zuweisungen an *zahl2* können den Wert der als Wert übergebenen Variablen $x(8)$ dagegen nicht verändern.

Komplette Arrays können Sie nur als Referenz übergeben. Der Ausdruck

```
Call Test(x())
```

übergibt *Test* das gesamte Array. Die Argumentliste von *Test* muss ein entsprechendes Übernahme-Array gleichen Typs enthalten:

```
Sub Test (zahl() As Single)
```

Die Größe des Übernahme-Arrays wird nicht angegeben, sondern nur eine Variable mit leeren Klammern als Kennzeichen dafür, dass es sich um ein Array handelt. *Test* besitzt nun Zugriff auf jedes einzelne Array-Element und kann zum Beispiel mit einer Zuweisung wie $zahl(5) = 23,4$ den Inhalt der korrespondierenden Variablen $x(5)$ verändern.

**Datenfeldgrenz-
funktionen**

Bei der Übergabe von Arrays an Prozeduren spielen die Array-Grenzfunktionen *UBound* und *Lbound* eine wichtige Rolle. Sie ermitteln den größten (*UBound*) bzw. kleinsten (*Lbound*) Index eines Arrays und übergeben ihn als Funktionswert:

```
UBound(Arrayname)
```

```
LBound(Arrayname)
```


Wurde ein Array mit *Dim Zahl(5 To 23) As Single* deklariert, übergibt *UBound(x)* den Wert 5 und *LBound(x)* den Wert 23.

UBound und *LBound* ermöglichen, sehr flexible Prozeduren zu schreiben, denen ein Array unbekannter Größe übergeben wird und die dennoch alle Elemente dieses Arrays manipulieren, indem in der Prozedur selbst mit *UBound* und *LBound* die Array-Grenzen ermittelt werden.

Hier sehen Sie ein Beispiel, das die in Kürze erläuterte *For*-Schleife verwendet:

```
Sub Test(zahl() As Single)
    Dim i As Single

    For i = LBound(zahl) To UBound(zahl)
        zahl(i) = 5
    Next i
End Sub
```

Listing 33.10:
Array-Grenz-
funktionen

Dieser Prozedur wird ein Array *zahl* unbekannter Größe übergeben. Die *For*-Schleife ist dennoch in der Lage, jeder Variablen dieses Arrays den Wert 5 zuzuweisen, da *LBound* die untere und *UBound* die obere Array-Grenze ermittelt.

Die Größe von mit *Dim* ohne Indexangaben deklarierten dynamischen Datenfeldern können Sie mit einer *ReDim*-Anweisung beliebig festlegen und mit nachfolgenden *ReDim*-Anweisungen nachträglich verändern.

**Dynamische
Datenfelder**

Wissen Sie am Anfang einer Prozedur noch nicht genau, wie viele Variablen Sie benötigen, deklarieren Sie das Datenfeld zunächst mit *Dim* ohne Größenangabe:

```
Dim Variablenname() As Typ
```

Nur in dieser Form, ohne Größenfestlegung, ist die Datenfeld-Deklaration mit *Dim* innerhalb einer nicht statischen Prozedur zulässig. Die Größe des Datenfelds legen Sie anschließend oder irgendwann später mit der *ReDim*-Anweisung fest:

```
Sub Test ()
    Dim x() As Single

    Anweisung1
    Anweisung2
    ...
    AnweisungN
```

Listing 33.11:
Redimensionieren
von Arrays

Listing 33.11:

Redimensionieren
von Arrays
(Forts.)

```
ReDim x(1 To 10) As Single

Anweisung1
Anweisung2
...
AnweisungN

ReDim x(23 To 54) As Single
Anweisung1
Anweisung2
...
AnweisungN
End Sub
```

Diese nur schematisch angedeutete Prozedur deklariert mit *Dim* das *Single*-Datenfeld *x()*, ohne seine Größe festzulegen. Die benötigte Größe ergibt sich erst nach einem Block von mehreren Anweisungen 1 bis N.

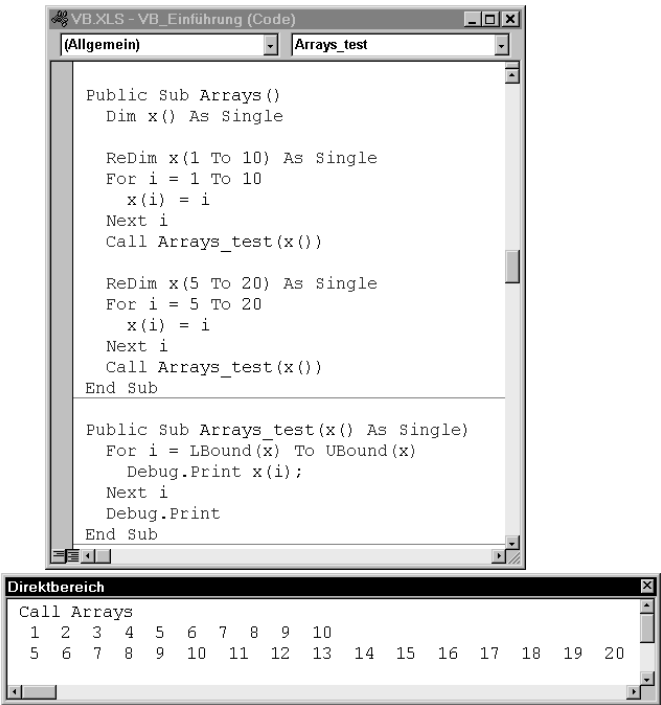
Danach werden mit *ReDim* zehn Variablen *x(1)* bis *x(10)* angelegt.

Später wird die Array-Größe geändert und mit einer weiteren *ReDim*-Anweisung die Variablen *x(23)* bis *x(54)* angelegt.

Ein praktisches Beispiel (Abbildung 33.18):

Abbildung 33.18:

Vb.xls, Modul
»VB_Einführung«,
Prozeduren
»Arrays« und
»Arrays_test«



Dieses Beispiel verwendet so ungefähr alles bisher über Arrays Gesagte: Am Anfang der Prozedur *Arrays* wird mit

```
Dim x() As Single
```

ein Array formal deklariert, ohne seine Größe festzulegen. Das passiert erst in der folgenden Anweisung

```
ReDim x(1 To 10) As Single
```

Den zehn Array-Variablen $x(1)$ bis $x(10)$ werden in einer *For*-Schleife der Reihe nach die Werte 1 bis 10 zugewiesen. Danach wird die Prozedur *Arrays_test* aufgerufen, die das gesamte Array in einer gleichnamigen Array-Variablen $x()$ entgegennimmt. Mit den Array-Grenzfunktionen wird die untere und obere Grenze ermittelt und in einer Schleife der Inhalt jeder Array-Variablen ausgegeben.

Anschließend wird zur Prozedur *Arrays* zurückgekehrt und das Array $x()$ dort mit

```
ReDim x(5 To 20) As Single
```

neu dimensioniert. Es enthält nun die Variablen $x(5)$ bis $x(20)$, denen erneut Werte zugewiesen werden. Danach wird das Array wieder *Arrays_test* übergeben, wo die Grenzen ermittelt und auch das neu dimensionierte Array korrekt behandelt und die Inhalte seiner Variablen ausgegeben werden.

Die bei der ersten Dimensionierung festgelegte Anzahl an Dimensionen kann nicht durch folgende ReDim-Anweisungen geändert werden.



Jede *ReDim*-Anweisung erzeugt das Array neu. Die zuvor in den einzelnen Variablen gespeicherten Werte gehen dabei verloren. Wollen Sie sie erhalten, benötigen Sie den Zusatz *Preserve*, der jedoch nur eingeschränkt anwendbar ist:

Preserve

- ➔ Bei den in Kürze erläuterten mehrdimensionalen Arrays darf nur die letzte Dimension geändert werden
- ➔ und es darf nur die Array-Obergrenze verändert werden.

Es ist daher nicht möglich, im Beispiel zur bestehenden Deklaration einfach *Preserve* hinzuzufügen:

```
ReDim Preserve x(5 To 20) As Single
```

Um *Preserve* einzusetzen, muss wie gesagt die Array-Untergrenze unverändert bleiben:

```
ReDim Preserve x(1 To 20) As Single
```

Dadurch bleiben die Inhalte der Variablen $x(1)$ bis $x(10)$ unverändert erhalten und zusätzlich gibt es nun nach der Neudimensionierung die neu angelegten Variablen $x(11)$ bis $x(20)$, die zunächst den Inhalt 0 besitzen.

Array-Funktion

Die Funktion *Array* übergibt einen Funktionswert vom Typ *Variant*, der ein Datenfeld enthält:

```
Array = Datenfeld(Argumentliste)
```

Beispielsweise übergibt der Ausdruck

```
Dim x As Variant
x = Datenfeld(22; 15; 35)
```

in x ein Datenfeld mit den drei Werten 22, 15 und 35. Obwohl x eine einzelne *Variant*-Variable ist und kein Array, können Sie auf die darin enthaltenen Elemente wie auf ein Array zugreifen und beispielsweise mit

```
y = x(3)
```

der Variablen y das letzte Element des in x enthaltenen Datenfelds zuweisen.

Mehrdimensionale Datenfelder

Kommen wir zu mehrdimensionalen Arrays. Die Anweisung

```
Dim x(1 Bis 10, 1 Bis 3)
```

deklariert ein zweidimensionales Array $x()$ mit dreißig Variablen:

Listing 33.12: Zweidimensionales Array

```
x(1, 1)
x(2, 1)
x(3, 1)
...
...
x(9, 1)
x(10, 1)
x(1, 2)
x(2, 2)
x(3, 2)
...
...
x(9, 2)
x(10, 2)
```

```
x(1, 3)
x(2, 3)
x(3, 3)
...
...
x(9, 3)
x(10, 3)
```

Listing 33.12:
Zweidimensionales
Array
(Forts.)

Vergleichen Sie ein zweidimensionales Array mit einer Tabelle, die in Spalten und Zeilen unterteilt ist.

Der Ausdruck *Spalten * Zeilen* ergibt die möglichen Zellen der Tabelle, die Koordinaten. Jede Zelle wird unter Angabe der Spalten- *und* der Zeilenkoordinaten angesprochen:

1	2	3	4	5	6	7	8	9	10
1	x	x	x	x	x	x	x	x	x
2	x	x	x	x	x	x	x	x	x
3	x	x	x	x	x	x	x	x	x

Listing 33.13:
Schematische Darstellung des Arrays

Beispielsweise weist der Ausdruck

$x(3,2) = 20$

der Variablen $x(3,2)$ dieses Arrays den Wert 20 zu, also dem hervorgehobenen Kästchen der Tabelle.

Sie können auch dreidimensionale Arrays erzeugen:

```
ReDim x(1 To 10, 1 To 3, 5 To 9)
```

Dieses dreidimensionale Array mit 150 Elementen ($10 * 3 * 5$ Variablen) kann man sich mit räumlichem Vorstellungsvermögen noch als Würfel vorstellen.

VBA ermöglicht Arrays mit maximal 60 Dimensionen. Die Gesamtzahl der Elemente (Variablen) hängt vom verfügbaren Speicherplatz ab.

Die vollständige Syntax von *LBound* und *UBound* lautet übrigens:

```
LBound(Arrayname [,Dimension])
UBound(Arrayname [,Dimension])
```

Beispielsweise ermittelt der Ausdruck $UBound(x, 1)$ die obere Grenze 10 der ersten Dimension des als Beispiel verwendeten Arrays x , $UBound(x, 2)$ die obere Grenze 3 der zweiten Dimension und $UBound(x, 3)$ die obere Grenze 9 der dritten Dimension.

Parameter-Arrays

Sie können einer Prozedur beliebig viele nacheinander aufgeführte Argumente als ein Parameter-Array übergeben. Dazu muss die Prozedur folgendermaßen deklariert sein:

```
Sub Testproc(nachname As String, ParamArray x() As Variant)
```

Der Aufruf dieser Prozedur erfolgt beispielsweise so:

```
Call Testproc("Maier", 5, 8, 17, 29, 2)
```

Das Schlüsselwort *ParamArray* vor dem zweiten Argument bewirkt, dass VBA das zweite übergebene Argument und alle folgenden Argumente zu einem Array zusammenfasst und dieses Array der Array-Variablen *x* übergibt.

x muss die letzte Prozedurvariable sein, der keine weitere Argumentdeklaration folgen darf! Außerdem muss x vom Typ Variant sein!



33.9 Selbstdefinierte Datentypen (Verbunde)

VBA besitzt nicht gerade wenige Datentypen. Sollten Ihnen die jedoch nicht ausreichen, deklarieren Sie einfach Ihre eigenen.

Wirklich neue Datentypen können Sie nicht konstruieren, aber immerhin Typen, die sich aus beliebigen Kombinationen der Grundtypen zusammensetzen, zumindest *fast* beliebigen Kombinationen, denn bei Strings dürfen nur Strings fester Länge verwendet werden.

Deklaration

Eigene Datentypen, so genannte Verbunde, Strukturen oder auch Records, die sich aus mehreren Variablen zusammensetzen, definieren Sie mit der *Type*-Anweisung:

Listing 33.14:
Die Type-
Anweisung

```
Type Benutzertyp
    Elementname As Typname
    Elementname As Typname
    ...
    ...
End Typ
```

Benutzertyp ist der Name, den Sie Ihrem Datentyp geben und der den Regeln für Variablennamen entspricht. *Elementname* ist eine Variable, die Bestandteil des Datentyps sein soll. *Typname* ist der Typ der Variable.

Mit *Type* können Sie zum Beispiel einen Datentyp *Adresstyp* konstruieren, der entsprechend der Struktur einer Adresse aus mehreren Stringvariablen (fester Länge) und einer Integervariablen besteht:

```
Type Adresstyp
  Vorname As String * 20
  Nachname As String * 30
  Plz      As Long
  Wohnort  As String * 40
End Type
```

Listing 33.15:
Der selbstdefinierte
Datentyp Adresstyp

Die Deklaration selbstdefinierter Datentypen ist nur auf Modulebene möglich, das heißt im Abschnitt (Deklarationen) des Prozeduren-Listenfelds. Keinesfalls darf eine solche Deklaration innerhalb einer Prozedur erfolgen!



Nach der Deklaration können alle Prozeduren dieses Moduls auf den neuen privaten Datentyp des Moduls zugreifen. Wenn Sie wollen, können Sie das zugehörige Schlüsselwort *Private* verwenden, um diesen Sachverhalt zu verdeutlichen:

```
Private Type Benutzertyp
```

Mit dem Zusatz *Public* deklarieren Sie dagegen Datentypen, auf die anschließend alle Prozeduren aller Module zugreifen können:

```
Public Type Benutzertyp
```

Beachten Sie, dass der neue Datentyp *Adresstyp* bisher nur *deklariert* wurde, also nur bekannt gegeben, aber noch keine Variablen *definiert*, das heißt angelegt wurden!

**Definition von
Variablen**

Benötigen Sie in irgendeiner Prozedur Ihres Programms eine Variable des neuen Typs *Adresstyp*, müssen Sie sie zuvor mit *Dim* und *Als* *definieren*. Zum Beispiel definiert

```
Dim adr As Adresstyp
```

eine Variable namens *adr* an, die weder eine numerische noch eine Stringvariable ist, sondern eben eine Variable vom Typ *Adresstyp*.

adr ist eine zusammengesetzte Variable, ein Verbund mehrerer Variablen. Zum Sprachgebrauch: *adr* ist eine Verbundvariable oder Recordvariable, die aus einzelnen Variablen besteht, den Komponenten dieses Datentyps.

Ein Verbund besitzt eine gewisse Ähnlichkeit mit einem Array, das ja ebenfalls mehrere Elemente enthält. Allerdings kann ein Verbund im Gegensatz zu einem Array Komponenten unterschiedlichsten Typs enthalten.

Die einzelnen Komponenten einer Verbundvariablen werden angesprochen, indem dem Variablennamen ein Punkt folgt und diesem Punkt der Name der angesprochenen Komponente:

Variablenname.Komponente

Ein Beispiel:

```
adr.Vorname = "Willi"
```

Dieser Ausdruck weist der Komponenten *Vorname* der Variablen *adr* die Zeichenkette »Willi« zu.

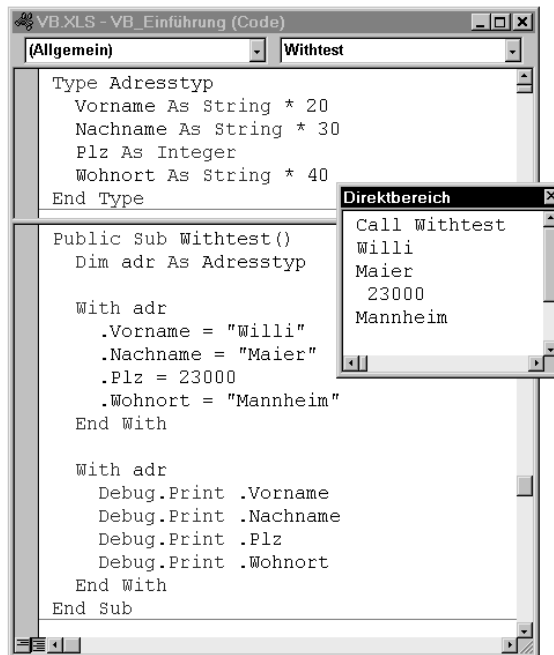
With Wollen Sie gleich mehreren Komponenten etwas zuweisen, können Sie sich das mithilfe der *With*-Anweisung vereinfachen. Normalerweise würden Sie dazu etwa folgende Anweisungen benutzen:

```
adr.Vorname = "Willi"
adr.Nachname = "Maier"
adr.Plz = 23000
adr.Wohnort = "Mannheim"
```

Mithilfe von *With* können Sie sich die immer wiederkehrende Angabe von *adr* ersparen (Abbildung 33.19).

Abbildung 33.19:

Vb.xls, Modul
»VB_Einführung«,
Prozedur »Withtest«



With adr bedeutet, dass sich alle folgenden mit dem Zeichen ».« eingeleiteten Verweise auf die Verbundvariable *adr* beziehen, bis zum Ende des Blocks, also bis zu *End With*. Der Ausdruck *.Plz* ist daher in diesem Block gleichbedeutend mit *adr.Plz*.

Wie Sie sehen, ist der Umgang mit den einzelnen Komponenten eines Verbunds recht einfach. Allerdings sind noch einige Fragen offen, etwa: Kann ich einen Verbund einem anderen Verbund zuweisen? Kann ein Verbund in einem anderen enthalten sein? Wie übergebe ich einen Verbund an eine Prozedur/Funktion und umgekehrt?

Zunächst einmal können Sie eine Verbund- oder Recordvariable einer anderen Recordvariablen zuweisen, die *vom gleichen Typ* ist. Nehmen wir an, Sie benutzen zwei Variablen unseres Typs *Adresstyp*:

Zuweisung

```
Dim adr1 As Adresstyp, adr2 As Adresstyp
```

Nehmen wir an, im weiteren Verlauf Ihres Programms wird in *adr1* eine vollständige Adresse gespeichert, bestehend aus vier Komponenten. Wenn Sie diese Adresse auch in *adr2* speichern wollen, benutzen Sie folgende Zuweisung:

```
adr2 = adr1
```

Intern passiert dabei Folgendes: Der ersten Komponente *Vorname* von *adr2* wird der Inhalt dieser Komponente in *adr1* zugewiesen. Dann wird die zweite Komponente von *adr1* in *adr2* kopiert und so weiter.

Natürlich können Sie auch einzelne Komponenten angeben. Zum Beispiel, wenn Sie nur die in der ersten Adresse enthaltene Postleitzahl in die zweite Adressvariable kopieren wollen:

```
adr2.Plz = adr1.Plz
```

Bei der komponentenweisen Zuweisung müssen Sie darauf achten, dass immer die gleichen Komponenten verwendet werden. Die Zuweisung

```
adr2.Plz = adr1.Vorname
```

führt zu einer Fehlermeldung, denn im Datentyp *Adresstyp* ist die Komponente *Plz* vom Typ *Long*, *Vorname* jedoch vom Typ *String*. Verständlicherweise können Sie einer numerischen Variablen jedoch nicht den Inhalt einer Stringvariablen – eine Zeichenkette – zuweisen.

Eine Verbundvariable können Sie komplett an eine Prozedur/Funktion übergeben, allerdings nur als Referenz, nicht als Wert. Der Aufruf

Übergabe

```
Call Demo(adr)
```

übergibt der Prozedur *Demo* die Verbundvariable *adr*. Die Parameterliste von *Demo* muss eine Übernahmevariable vom gleichen Typ aufweisen:

```
Sub Demo(adresse As Adresstyp)
```

Ähnlich übergeben Sie eine einzelne Komponente. Sie führen einfach die Komponente in der Argumentliste auf:

```
Call Demo(adr.Vorname)
```

In der zugehörigen Prozedur-Parameterliste führen Sie wieder eine Übernahmevariable auf, die den gleichen Typ (*String * 20*) wie die übergebene Komponente besitzt:

```
Sub Demo (s As String * 20)
```

Alternativ können Sie übrigens auch eine Stringvariable variabler Länge verwenden:

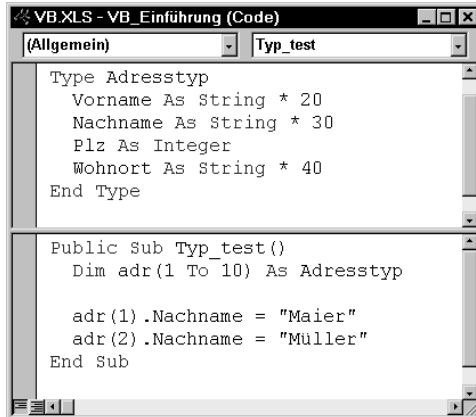
```
Sub Demo (s Als ZeichenF)
```

Hauptsache, die Übernahmevariable ist eine Stringvariable, da ja eine Zeichenkette und keine Zahl übergeben wird.

Sie können auch Arrays aus Verbundvariablen deklarieren, beispielsweise so (Abbildung 33.20).

Abbildung 33.20:

Vb.xls, Modul
»VB_Einführung«,
Deklaration von
»Adresstyp« und
Prozedur »Typ_test«



Das Array *adr* besteht aus den zehn Variablen *adr(1)* bis *adr(10)*, die jeweils vom Typ *Adresstyp* sind und denen wie in der Abbildung gezeigt nun zum Beispiel komponentenweise Daten zugewiesen werden können. Für den Zugriff gilt somit die Syntax

```
Datenfeldname(Index).Komponente
```

Eine außerordentlich wichtige Rolle spielen verschachtelte Verbunde. Jede Komponente eines Verbunds darf nicht nur aus einer einfachen Variablen bestehen, sondern selbst wieder aus einem Verbund. Ein Beispiel:

```
'Innerer Verbund
Type AdressOrt
  Plz As Long
  Ort As String * 40
End Type
```

```
'Äußerer Verbund
Type Adresstyp
  Vorname As String * 20
  Nachname As String * 30
  Wohnort As AdressOrt
End Type
```

```
'Verbundvariable
Dim adr As Adresstyp
```

Der Verbund *Adresstyp* besteht aus zwei einfachen Komponenten und zusätzlich aus einer Komponenten *Wohnort*, die selbst einen Verbund mit zwei Komponenten darstellt.

Die Frage ist, wie eine Komponente dieses inneren Verbunds angesprochen wird, beispielsweise die Postleitzahl:

```
Variablenname.KomponenteA.KomponenteI
```

KomponenteA ist eine Komponente des äußeren Verbunds und *KomponenteI* eine Komponente des inneren Verbunds. Entsprechend ist

```
adr.Wohnort.Plz
```

ein Bezug auf die in diesem inneren Verbund enthaltene Komponente *Plz*. Verbunde können beliebig tief ineinander verschachtelt sein. Bezüge auf tief im Innern enthaltene Komponenten besitzen folgende Form:

```
Variablenname.Komponente.Komponente.Komponente....
```

Listing 33.16:
Verschachtelte
Verbunde

34 Kontrollstrukturen

Ich erläutere nun die Kontrollstrukturen von VBA, die Befehle zur Steuerung des Programmablaufs.

Bisher liefen Ihre Programme rein sequentiell ab, das heißt eine Anweisung nach der anderen wurde ausgeführt. Umfangreiche und mit unterschiedlichsten Features versehene Anwendungen erfordern jedoch weitaus komplexere Abläufe.

Ich zeige Ihnen, wie Sie

- ➔ mithilfe von Bedingungen Entscheidungen treffen, nach dem Motto »führe die folgenden drei Anweisungen nur aus, wenn Bedingung A erfüllt ist«,
- ➔ auch komplexeste Bedingungen formulieren, bei denen zwischen einer ganzen Reihe unterschiedlicher Zustände unterschieden wird und je nachdem entweder Anweisung A oder Anweisung B oder Anweisung C ausgeführt wird,
- ➔ mit Schleifen Anweisungen wiederholt ausführen lassen; im einfachsten Fall, um zehnmal nacheinander »Hallo« auszugeben, in sinnvolleren Anwendungen beispielsweise, um den Benutzer mit nur drei oder vier Anweisungen immer wieder zur Eingabe von Werten aufzufordern und diese zum aktuellen Zwischenresultat zu addieren
- ➔ und welche Möglichkeiten VBA zur Behandlung von Laufzeitfehlern bietet.

Werden Ihre Programme anspruchsvoller, spielt Letzteres eine zunehmend wichtigere Rolle. Sie kennen bestimmt die berühmte Fehlermeldung »Abbrechen, Wiederholen, Ignorieren?«, wenn Sie auf ein Diskettenlaufwerk wechseln wollen, das keine Diskette enthält oder nicht verriegelt ist.

Der gleiche Fehler tritt auch in einem VBA-Programm auf. VBA stellt Ihnen jedoch vielseitige Möglichkeiten zur Behandlung derartiger Fehler zur Verfügung, auf die ich detailliert eingehen werde.

Ich erläutere, wie Ihr Programm Laufzeitfehler erkennen und darauf reagieren kann, beispielsweise durch Ausgabe einer eigenen Fehlermeldung, Wiederholung der fehlgeschlagenen Aktion oder (falls sinnvoll) Ignorieren des Fehlers und Fortsetzung mit der nächsten Anweisung.

Wie ich Ihnen zeige, ermöglicht VBA sogar, potentielle Fehler, die normalerweise kaum getestet werden können (beispielsweise den Defekt eines Diskettenlaufwerks), zu simulieren und zu prüfen, ob Ihre Fehlerbehandlungsroutinen auch darauf korrekt reagieren.

34.1 Verzweigungen

GoTo Die einfachste Kontrollstruktur ist die Verzweigung, der gezielte Sprung zu einer bestimmten Anweisung:

```
GoTo {Zeilenmarke | Zeilennummer}
```

Dem Schlüsselwort *GoTo* folgt eine Zeilenmarke oder eine Zeilennummer. Vergessen Sie bitte die Zeilennummer. Das Programmieren mit Zeilennummern gehört in die Zeit vor VBA. Ich werde *GoTo* ausschließlich mit Zeilenmarken einsetzen.

Eine Zeilenmarke (auch Label genannt) ist ein Name, der den gleichen Konventionen wie Variablennamen unterliegt, aber mit einem Doppelpunkt enden muss:

Test:

GoTo unterbricht den sequentiellen Programmfluss und verzweigt zur angegebenen Zeilenmarke, um die Programmausführung dort fortzusetzen (Abbildung 34.1).

Die Prozedurausführung beginnt wie üblich mit der ersten Anweisung, die im Direktfenster »Test1« ausgibt. *GoTo Marke* verzweigt zur Zeile mit dem Label »Marke:«. Die folgenden drei Anweisungen werden daher übersprungen und die nächste Anweisung, die ausgeführt wird, ist *Debug.Print »Test5«* und nicht etwa *Debug.Print »Test2«*.

**Bedingte
Verzweigung**

Interessanter als die unbedingte ist die bedingte Verzweigung. Dazu benötigen Sie eine Kombination von *GoTo* und *If*. Mit *If* wird eine Bedingung getestet und in Abhängigkeit davon mit *GoTo* ein Sprung ausgeführt oder auch nicht. Das Schema:

```
If Bedingung Then GoTo Label
```

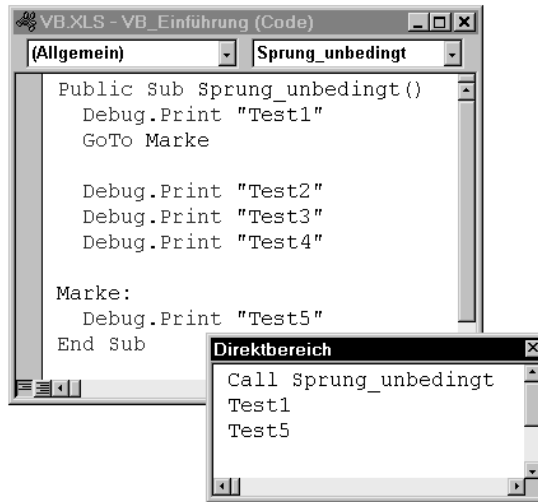


Abbildung 34.1:

Vb.xls, Modul
»VB_Einführung«,
Prozedur
»Sprung_unbedingt«

Ist *Bedingung* erfüllt, wird der Sprung *GoTo Label* ausgeführt, sonst nicht. Mit nach diesem Schema gebildeten bedingten Sprüngen lassen sich alle höheren Kontrollstrukturen von VBA nachbilden, zum Beispiel Zählschleifen.

Schleifen

Stellen Sie sich vor, eine bestimmte Aufgabe – zum Beispiel die Ausgabe des Textes »Hallo« – soll genau zehnmal ausgeführt, der zugehörige Programmblock also zehnmal durchlaufen werden (Abbildung 34.2).

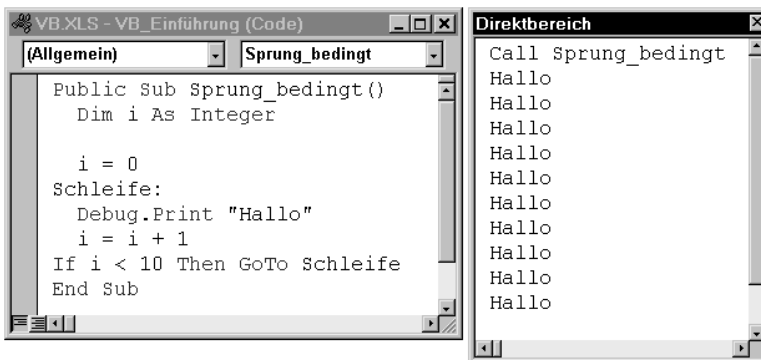


Abbildung 34.2:

Vb.xls, Modul
»VB_Einführung«,
Prozedur
»Sprung_bedingt«

Die *Integervariable* *i* wird benutzt, um zu erkennen, wie oft die Schleife bereits ausgeführt wurde und erhält zu Beginn entsprechend den Wert 0.

Nun folgen die Anweisungen, die zehnmal wiederholt werden sollen, der Aufruf der *Print*-Methode, um im Direktfenster »Hallo« auszugeben und das Erhöhen von *i* um 1, um anzuzeigen, dass die gewünschte Aktion soeben einmal mehr ausgeführt wurde.

Danach – am Schleifenende – wird geprüft, ob der Inhalt der Zählvariablen noch kleiner als zehn ist. Wenn ja, wird mit *GoTo Schleife* zum Schleifenanfang verzweigt und der nächste Durchgang beginnt.

Dieses Spielchen wiederholt sich genau zehnmal. Nach der zehnten Erhöhung von *i* um 1 besitzt diese Variable den Wert 10. Die Aussage *i < 10* ist falsch und die Sprunganweisung wird nicht ausgeführt.

34.1.1 Entscheidungen und Bedingungen

- If* Für Entscheidungen zwischen mehreren Alternativen wird vorwiegend *If* eingesetzt. VBA kennt unterschiedliche Formen von *If*-Strukturen, eine einzeilige und eine Blockstruktur. Die einfachere einzeilige Form kennen Sie bereits, abgesehen von einer optionalen Angabe, die ich bisher verschwieg:

```
If Aussage Then Anweisung1 [Else Anweisung2]
```

Bedingungen Ist *Aussage WAHR*, wird *Anweisung1* ausgeführt.

Statt *Aussage* werde ich im Folgenden den gebräuchlicheren Ausdruck *Bedingung* verwenden. Diese Bedingung kann aus einem halben Dutzend Einzelbedingungen bestehen, die mit logischen Operatoren verknüpft sind.

Die Anweisung wiederum kann eine einzelne Anweisung sein oder eine Kombination mehrerer durch Doppelpunkte getrennter Anweisungen:

```
If (a = 1 And b <= 20) Or (c >= 10) Then Debug.Print "Okay": GoTo Marke
```

Die Klammern sind eigentlich überflüssig und sollen nur die Bewertungsreihenfolge verdeutlichen. Übersetzt lautet der Ausdruck: »Wenn *a* gleich 1 und gleichzeitig *b* kleiner als 20 ist oder *c* größer oder gleich 10 ist, dann gebe »Okay« aus und verzweige zu »Marke"«.

- Else* Der optionale Teil *Else Anweisung2* ermöglicht die Ausführung einer anderen Anweisung, wenn die angegebene Bedingung nicht erfüllt ist.

```
If a = 1 Then Debug.Print "a enthält 1" Else Debug.Print "a enthält nicht 1"
```

Ist die Bedingung *a = 1* erfüllt, wird der *Then*-Zweig ausgeführt, sonst der *Else*-Zweig.

If-Then-Else entspricht somit dem umgangssprachlichen *entweder...oder*. Beide Zweige können Kombinationen von – durch Doppelpunkte getrennten – Anweisungen enthalten.

In der folgenden Abbildung wird die Funktion *InputBox* aufgerufen, um den Anwender zu fragen, ob zwei Zahlen addiert werden sollen (Abbildung 34.3).

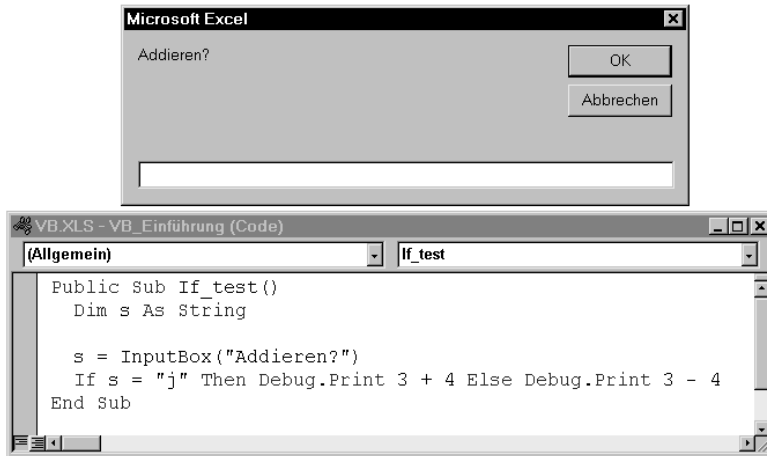


Abbildung 34.3:

Vb.xls, Modul
»VB_Einführung«,
Prozedur »If_test«

Antwortet er mit »j«, enthält die Stringvariable *s* anschließend dieses Zeichen und die Bedingung *If s = »j«* ist erfüllt. Die Anweisung *Debug.Print 3 + 4* wird ausgeführt, die beiden Zahlen somit addiert und das Ergebnis im Direktfenster ausgegeben.

Gab der Anwender etwas anderes als »j« ein, beispielsweise überhaupt nichts, ist die Bedingung nicht erfüllt und entsprechend wird stattdessen die Anweisung *Debug.Print 3 - 4* ausgeführt.

Oft sollen in einem *Then*- oder *Else*-Zweig *mehrere* Anweisungen ausgeführt werden, wenn die Bedingung erfüllt bzw. nicht erfüllt ist. Die einfachste Möglichkeit ist, im *If*- oder *Then*-Zweig mehrere durch Doppelpunkte getrennte Anweisungen auszuführen:

```
If Ausdruck Then Anweisung1: Anweisung2: ...: AnweisungN
Else Anweisung1: Anweisung2: ...: AnweisungN
```

Bei der Ausführung eines der beiden Zweige werden jeweils *alle* darin enthaltenen Anweisungen ausgeführt. Ein Beispiel:

```
If a = 1 Then Debug.Print "a enthält 1": Debug.Print "b vielleicht auch" Else
Debug.Print "a enthält nicht 1": Debug.Print "b vielleicht auch nicht"
```

Sowohl im *If*- als auch im *Else*-Zweig werden nun jeweils zwei durch Doppelpunkte getrennte Anweisungen ausgeführt, wenn die Bedingung erfüllt bzw. nicht erfüllt ist.

If-Blöcke

Wie Sie sehen, werden einzelzeilige *If*-Anweisungen schnell unübersichtlich, wenn ein Zweig mehrere Anweisungen enthält. In solchen Fällen benutzen Sie Blockstrukturen. Die einfachste Form ist diese:

Listing 34.1: If Ausdruck Then
If-Then-Blockstruktur
End If

Ist die Bedingung *Ausdruck* erfüllt, werden der Reihe nach alle Anweisungen ausgeführt, die sich in dem Block zwischen *Then* und *End If* befinden. Ein Beispiel dafür ist:

Listing 34.2: If x < 20 Then
Beispiel für einen Then-Block
Debug.Print "x ist nicht größer als 20"
Debug.Print "x ist auch nicht gleich 20"
Debug.Print "x ist größer als 20 "
End If

If-/Else-Blöcke Jede Anweisung kann somit in eine eigene Zeile geschrieben werden. Die Anweisungen zwischen *If* und *Then* bilden einen gemeinsam auszuführenden Anweisungsblock.

Ein solcher Block ist auch für den *Else*-Zweig möglich, wodurch wir die *If-Then-Else*-Blockstruktur erhalten:

Listing 34.3: If Ausdruck1 Then
If-Then-Else-Blockstruktur
[Block 1]
[Else
[Block 2]]
End If

Ein Beispiel:

Listing 34.4: If x < 20 Then
Beispiel für einen If...Then...Else-Block
Debug.Print "x ist nicht größer als 20"
Debug.Print "x ist auch nicht gleich 20"
Debug.Print "x ist kleiner als 20 !!!"
Else
Debug.Print "x ist nicht kleiner als 20"
Debug.Print "x ist entweder gleich 20"
Debug.Print "oder x ist größer als 20 !!!"
End If

Elseif Häufig hat man in Programmen nicht nur unter zwei, sondern unter drei, vier oder noch mehr Alternativen zu wählen. Dann benötigen Sie eine erweiterte Form von *If-Then-Else*:

Listing 34.5: If Ausdruck1 Then
If-Then-Elseif-Blockstruktur
[Block 1]
ElseIf Ausdruck2 Then
[Block 2]]
ElseIf Ausdruck3 Then
...
End If

...
End If

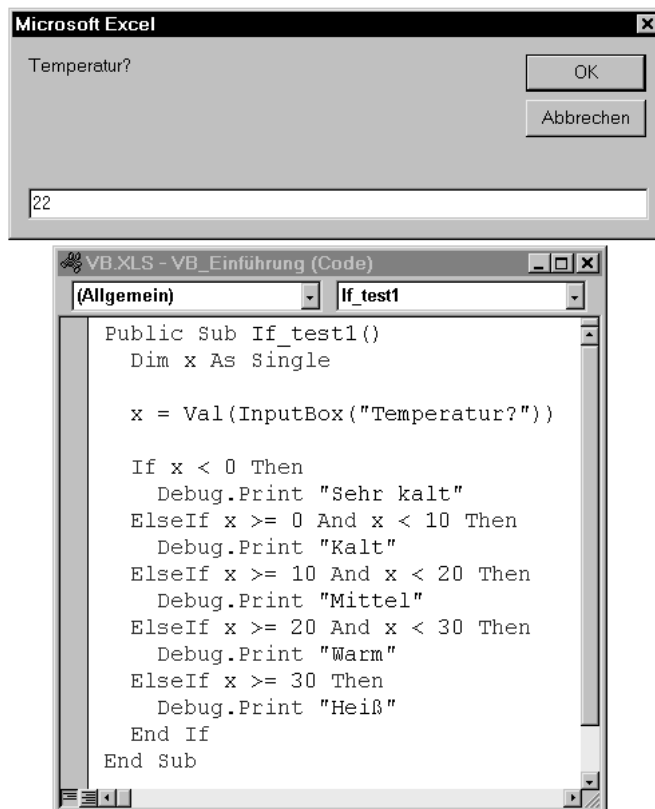
Diese Blöcke kann man so interpretieren: »Wenn Ausdruck1 *TRUE* ist, führe Block1 aus. Wenn Ausdruck1 nicht *TRUE* ist, aber Ausdruck2 *TRUE* ist, führe Block2 aus. Wenn weder Ausdruck1 noch Ausdruck2 *TRUE* sind, dafür jedoch Ausdruck3 *TRUE* ist, führe Block3 aus«.

Die Punkte deuten an, dass beliebig viele weitere *ElseIf*-Blöcke folgen dürfen. Jeder Unterblock ist an eine eigene Bedingung geknüpft und wird nur ausgeführt, wenn sie erfüllt ist *und die vorhergehenden Bedingungen nicht erfüllt waren*. Daher wird keinesfalls mehr als ein Block ausgeführt.

Ein Beispiel: Liegt ein bestimmter Temperaturwert unter 0 Grad Celsius, soll das Programm melden »Sehr kalt«. Liegt er zwischen 0 und 10 Grad, lautet die Meldung »Kalt«. Zwischen 10 und 20 Grad wird »Mittel« gemeldet und so weiter (Abbildung 34.4).

Listing 34.5:
If-Then-Elseif-
Blockstruktur
(Forts.)

Abbildung 34.4:
Vb.xls, Modul
»VB_Einführung«,
Prozedur »If_test1«



Der Anwender wird mit der *InputBox*-Funktion nach einem Temperaturwert gefragt. Die Funktion übergibt die Eingabe als Zeichenkette, als Datentyp *String*. Da der Wert jedoch einer numerischen Variable zugewiesen werden soll, wird zuvor die Funktion *Value* angewendet, die den numerischen Wert der Zeichenkette ermittelt, und dieser Wert wird x zugewiesen, im Beispiel also die Zahl 22.

Danach wird geprüft, ob x kleiner ist als 0. Wenn ja, wird der folgende Block und die einzige darin enthaltene Anweisung ausgeführt:

```
Debug.Print "Sehr kalt"
```

Ist die erste Bedingung nicht erfüllt, wird die nächste Bedingung geprüft. Ist x größer oder gleich 0 und kleiner als 10, ist sie erfüllt und der zugehörige zweite Block wird ausgeführt, also

```
Debug.Print "Kalt"
```

Im Beispiel ist jedoch erst die vierte Bedingung erfüllt und somit wird folgende Anweisung ausgeführt:

```
Debug.Print "Warm"
```

Beachten Sie, dass niemals mehr als ein Block ausgeführt wird, auch wenn mehr als eine Bedingung erfüllt ist! Ist eine Bedingung erfüllt, wird der zugehörige Anweisungsblock ausgeführt und *ohne Prüfung der restlichen Bedingungen* sofort zu *End If* verzweigt.

Vollständige Syntax

Mit *ElseIf*-Zweigen ist es sehr einfach, mehrere Alternativen zu testen.

Was aber, wenn die möglichen Alternativen unvollständig abgedeckt sind? Wenn möglicherweise keine der Alternativen in Frage kommt, da der Anwender eine Eingabe vornehmen kann, die keiner der mit *If* und *ElseIf* formulierten Bedingungen genügt? Wenn er zum Beispiel nach Autotypen gefragt wird, deren PS-Zahl er wissen will, aber einen dem Programm unbekannten Typ eingibt?

Dann benötigen wir die vollständige Syntax von *If-Then-Else*:

Listing 34.6:

If-Then-ElseIf-Else-
Blockstruktur

```
If Ausdruck1 Then
    [Block 1]
ElseIf Ausdruck2 Then
    [Block 2]
ElseIf Ausdruck3 Then
    [Block 3]
ElseIf Ausdruck4 Then
    [Block 4]
```

```
...  
...  
  
[Else  
    [Block N]]  
End If
```

Ist keine der mit *ElseIf* überprüften Bedingungen erfüllt, wird der abschließende *Else*-Zweig ausgeführt.

Ein Beispiel:

```
If auto$ = "BMW 525 i" Then  
    [Block 1]  
ElseIf auto$ = "Kadett 16V" Then  
    [Block 2]  
ElseIf "Porsche 911 Turbo" Then  
    [Block 3]  
...  
...  
Else  
    Print "Diesen Typ kenne ich nicht"  
End If
```

Gibt der Anwender »VW Käfer« ein, wird entsprechend »Diesen Typ kenne ich nicht« ausgegeben.

Eine spezielle Form der If-Anweisung wird zur bedingten Kompilierung benutzt, die so genannte #If-Compiler-Direktive:

```
#If ... Then ... #Else
```

Die Syntax ist mit den verschiedenen *If*-Blockstrukturen identisch (es gibt allerdings keine einzeilige Variante). Der einzige andere syntaktische Unterschied: die Schlüsselwörter *#If*, *#Else*, *#ElseIf* und *#End If* werden mit dem Zeichen »#« eingeleitet.

Nur wenn der angegebene Ausdruck *TRUE* ist, werden die dem Schlüsselwort *#If* folgenden Anweisungen überhaupt kompiliert, ansonsten werden sie von VBA ebenso ignoriert wie Kommentare.

VBA sieht praktisch entweder nur den *#If* bzw. nur den *#Else* folgenden Programmcode, abhängig davon, ob die Bedingung erfüllt ist oder nicht.

Verwenden Sie beispielsweise die Bedingung *#If Win32 = TRUE*, die nur erfüllt ist, wenn Ihr Programm in einer 32-Bit-Entwicklungsumgebung abläuft, so können Sie unterschiedliche Programmteile für unterschiedliche Entwicklungsumgebungen schreiben.

Listing 34.6:
If-Then-ElseIf-Else-Blockstruktur
(Forts.)

Listing 34.7:
Beispiel für die vollständige Syntax



34.1.2 Mehrfachtests

Select Wenn Sie nur eine einzige Variable testen wollen und daran interessiert sind, welchen von verschiedenen vorgegebenen Werten sie enthält oder in welchen von ebenfalls fest vorgegebenen Zahlenbereichen sie sich befindet, ist *Select* komplexen *If*-Blöcken vorzuziehen.

Listing 34.8: Select Case Testausdruck
 Select Case
 Case Ausdrucksliste 1
 [Anweisungsblock 1]
 Case Ausdrucksliste 2
 [Anweisungsblock 2]
 ...
 ...
 ...
 [Case Else
 [Anweisungsblock N]]
 End Select

Case *Testausdruck* ist normalerweise der Name einer zu testenden Variablen, zum Beispiel *x* (Beispiel: *Select Case x*). In den folgenden *Case*-Anweisungen geben Sie nur noch die zu testenden Werte an, nicht mehr den Variablennamen selbst.

Zum Beispiel prüft *Case 15*, ob die angegebene Variable den Wert 15 enthält. Wenn ja, wird der zugehörige Anweisungsblock ausgeführt. An diesem Beispiel sehen Sie, dass sich jedes *Case* automatisch auf die nur einmal in *Testausdruck* angegebene Variable bezieht.

Damit ergeben sich sehr einfache und übersichtliche Tests wie der folgende:

Listing 34.9: Variablentests mit Select Case
 Select Case zeichen\$
 Case "a"
 Debug.Print "Die Variable enthält das Zeichen 'a'"
 Case "b"
 Debug.Print "Die Variable enthält das Zeichen 'b'"
 Case "c"
 Debug.Print "Die Variable enthält das Zeichen 'c'"
 Case Else
 Debug.Print "Die Variable enthält weder 'a' noch 'b' noch 'c'"
 End Select

Da sich jeder *Case*-Zweig auf die am Anfang angegebene Variable bezieht, ist eine Anweisung wie

```
If x < 10 And y < 10 Then Debug.Print "Beide Zahlen sind kleiner als 10"
```

nicht mit *Select* übersetzbar. Für den gleichzeitigen Test mehrerer Variablen kommt nur *If* in Frage. Weitere Einschränkungen von *Select* betreffen logische Operatoren. In einer *Case*-Ausdrucksliste dürfen keine logischen Operatoren verwendet werden!

Mit *If* ist es dagegen einfach, zu prüfen, ob x eine der beiden Zahlen 5 oder 10 enthält:

```
If x = 5 Or x = 10 Then ...
```

Select bietet für das verbotene *Or* einen Ersatz, sodass es immerhin möglich ist, durch diese Struktur mehrere mit *Or* verknüpfte *If*-Anweisungen zu ersetzen. *Case*-Ausdruckslisten dürfen nämlich nicht nur einen zu testenden Wert enthalten, sondern beliebig viele (durch Kommata getrennte) Werte.

Case-Ausdrucks-
listen

```
Select Case name$
  Case "Bauer", "Baier", "Bertram"
    Print "Der in der Variablen enthaltene Name beginnt mit 'B'"
  Case "Werner", "Walter", "Wolters"
    Print "Der in der Variablen enthaltene Name beginnt mit 'W'"
  Case Else
    Print "Der in der Variablen enthaltene Name beginnt weder mit 'B' noch
    mit 'W'"
End Select
```

Listing 34.10:
Case-Ausdrucks-
listen

Wie dieses Listing zeigt, können Sie hinter *Case* ganze Wertelisten angeben. Ist einer der Werte mit dem Testausdruck identisch, wird der zugehörige Anweisungsblock ausgeführt.

In Verbindung mit dem Wort *Is* dürfen Sie in *Case*-Ausdruckslisten auch relationale Operatoren verwenden: *Is*

```
Case Is Relationaler_Operator Ausdruck
```

Zum Beispiel testet *Case Is < 5*, ob die zu prüfende Variable kleiner als 5 ist. Und *Case Is >= 23.12* testet, ob ihr Inhalt größer oder gleich 23.12 ist. Wenn ja, wird jeweils der zugehörige Anweisungsblock ausgeführt.

Mit *If* können Sie Bereiche prüfen, zum Beispiel, ob die Variable x einen Wert zwischen 5 und 10 enthält: *Bereichsprüfung*

```
If x > 5 And x < 10 Then ...
```

Mit *Select* funktioniert die Bereichsprüfung aufgrund der verbotenen logischen Operatoren zwar anders, aber sogar eher noch einfacher:

```
Case Bereichsanfang To Bereichsende
```

Case 3 To 7 testet, ob sich das getestete Objekt im Bereich zwischen 3 und 7 befindet (3 und 7 sind dabei eingeschlossen).

Die folgenden Beispiele sollten Sie sich näher anschauen. Sie demonstrieren, dass auch Variablen in Ausdruckslisten verwendbar sind und natürlich auch Strings.

Tabelle 34.1:
Beispiele für Case-
Ausdruckslisten

Ausdrucksliste	Testfunktion
Case 98	Testausdruck gleich 98?
Case 5, x, 30	Testausdruck gleich 5 oder gleich x oder gleich 30?
Case Is >= 5	Testausdruck größer oder gleich 5
Case x To y	Testausdruck größer gleich x und kleiner gleich y?
Case »Maier"	Testausdruck gleich »Maier«?
Case »a«, Name\$	Testausdruck gleich »a« oder gleich Name\$?
Case Is < »Otto"	Testausdruck (alphabetisch) kleiner als »Otto«?
Case »c« To »n"	Testausdruck größer gleich »c« und kleiner gleich »n«?

34.2 Schleifen

Bisher wurden all unsere Anweisungen höchstens einmal ausgeführt. Für die wiederholte Ausführung von Anweisungen benötigen wir eine Schleife, eine Möglichkeit, die Ausführung eines Anweisungsblocks gewissermaßen im Kreis laufen zu lassen: Am Ende des Anweisungsblocks wird von vorne begonnen, der gesamte Block also zum zweiten Mal ausgeführt und so weiter, bis die gewünschte Anzahl an Wiederholungen stattfand. Dann wird der Block verlassen und die folgenden Anweisungen werden ausgeführt.

Diese Anforderungen machen bereits klar, was Anweisungen zur Bildung von Programmschleifen bieten müssen:

- ➔ Schlüsselwörter zur Begrenzung des wiederholt auszuführenden Blocks
- ➔ Möglichkeiten zur Kontrolle der Wiederholungsanzahl

34.2.1 For...Next-Schleifen

VBA besitzt drei recht unterschiedliche Schleifenkonstruktionen. Die bei weitem am häufigsten verwendete ist die *For...Next*-Schleife. Der – etwas vereinfachte – Aufbau sieht so aus:


```
For Zähler = Start To Ende [Step Schrittweite]
    [Block]
Next Zähler
```

Listing 34.11:
For...Next-Schleife

Die Schlüsselwörter *For* und *Next* begrenzen den zu wiederholenden Anweisungsblock. *Zähler* ist eine beliebige numerische Variable, die Zählvariable oder Schleifenvariable genannt wird.

Schleifenvariable

Start ist ein numerischer Wert, der dieser Variablen zu Beginn der Schleife zugewiesen wird. *Ende* ist ein Wert, der die Anzahl der Wiederholungen bestimmt, der Endwert der Schleifenvariablen.

**Start- und
Endwert**

Schrittweite legt fest, um welchen Betrag *Zähler* nach jedem Durchgang erhöht wird (bzw. vermindert, wenn die Schrittweite eine negative Zahl ist). Entfällt dieser optionale Parameter, wird *Zähler* nach jedem Durchgang um den Wert 1 erhöht. Nehmen wir als Beispiel eine einfache Schleife ohne die optionale Schrittweite:

Schrittweite

```
For i = 1 To 10
    Debug.Print "Hallo"
Next i
```

Listing 34.12:
Beispiel für
For-Schleife

Am Schleifenanfang erhält die Variable *i* – unsere Zählvariable – den Wert 1. Als Endwert wurde 10 angegeben. Das heißt, die Schleife soll beendet werden, wenn *i* größer ist als dieser Endwert.

Ohne die optionale Schrittweite wird *i* nach jedem Durchgang um 1 erhöht. Nach genau zehn Durchgängen enthält *i* daher den Wert 11 und ist nun größer als der angegebene Endwert 10 – die Schleife ist beendet.

Der Anweisungsblock wird daher zehnmal durchlaufen; zehnmal wird die Anweisung *Debug.Print* »Hallo« ausgeführt und jeweils »Hallo« im Direktfenster ausgegeben.

Dass die Schleifenvariable nach jedem Durchgang um 1 erhöht wird, ist oft nützlich. Stellen Sie sich vor, Sie wollen aus irgendeinem Grund die Zahlenreihe 1,2,3,4,...,15 auf dem Bildschirm ausgeben:

```
For i = 1 To 15
    Debug.Print i
Next i
```

Listing 34.13:
Die Schleifen-
variable

Die Anweisungen *For i = 1 To 15* und *Next i* bewirken, dass *i* zuerst den Wert 1 besitzt und nach jedem Durchgang um 1 erhöht wird, also der Reihe nach die gewünschten Werte 1 bis 15 annimmt. Und mit *Debug.Print i* als zu wiederholender Anweisung wird in jedem Durchgang der gerade aktuelle Wert ausgegeben.

Start- und Endwert Durch Verändern von Start- bzw. Endwert können Sie beliebige andere Zahlenreihen ausgeben. Zum Beispiel gibt das Programm

Listing 34.14: For i = 20 To 33
Start- und Endwert Debug.Print i
Next i

der Reihe nach die Zahlen 20 bis 33 aus. Im ersten Durchgang hat *i* den Wert 20, den *Debug.Print i* nun ausgibt. Nach diesem Durchgang wird *i* um 1 erhöht und besitzt somit im zweiten Durchgang den Wert 21, im dritten den Wert 22 und so weiter, bis *i* im letzten Durchgang den Endwert 33 enthält.

Schrittweite Die optionale Schrittweite (Standardwert 1) bestimmt, um welchen Betrag die Schleifenvariable nach jedem beendeten Durchgang erhöht wird.

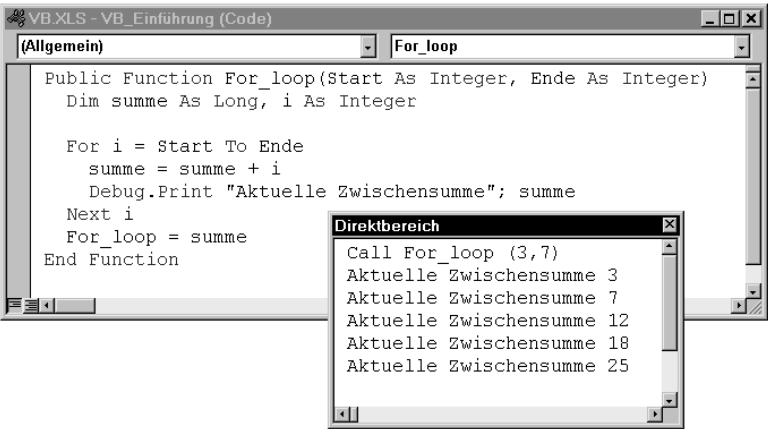
Listing 34.15: For i = 10 To 100 Step 5
Step Debug.Print i
Next i

In dieser Schleife erhält *i* den Startwert 10 und wird nach jedem Durchgang um 5 erhöht. In jedem Durchgang gibt *Debug.Print i* den gerade aktuellen Wert der Schleifenvariable aus, also der Reihe nach die Zahlen 10, 15, 20, 25, ...,100.

Die letzte ausgegebene Zahl ist 100, da *i* durch die anschließende Erhöhung den Wert 105 annimmt, nun größer als der Endwert ist und die Schleife daher beendet wird.

Außer der Ausgabe von Zahlenfolgen sind natürlich auch sinnvollere Anwendungen denkbar (Abbildung 34.5).

Abbildung 34.5:
Vb.xls, Modul
»VB_Einführung«,
Prozedur
»For_Loop«



Der Funktion *For_Loop* werden zwei *Integer*-Werte übergeben, Anfang und Ende eines Zahlenintervalls. *For_Loop* summiert die in diesem Intervall übergebenen Zahlen.

Rufen Sie *For_Loop* entsprechend der Abbildung mit *Print For_Loop(3,7)* auf, so addiert sie die Zahlen 3, 4, 5, 6 und 7 und übergibt die Summe 25 als Funktionswert.

Zur Arbeitsweise: Die Variable *Summe* ist aufgrund des benötigten Wertebereichs eine *Long*-Variable, denn bereits die Summe der Zahlen 100 bis 1000 ergibt 495550, was den Wertebereich des Typs *Integer* deutlich überschreitet.

Die Schleifenvariable *i* erhält als Startwert den von Ihnen übergebenen Bereichsanfang *Start*.

Die Anweisung *Summe = Summe + i* addiert zum aktuellen Inhalt von *Summe* (0 nach dem Aufruf der Funktion) den momentanen Wert der Schleifenvariable, also den Startwert *Start*, die erste Zahl des Sie interessierenden Bereichs.

Debug.Print gibt den Inhalt von *Summe* aus, die aktuelle Zwischensumme.

Der Anweisungsblock in der Schleife wurde zum ersten Mal ausgeführt. Nun wird die Schleifenvariable *i* erhöht und der zweite Durchgang beginnt.

Da *i* um 1 erhöht wurde, wird diesmal die zweite Zahl des Bereichs zur aktuellen Zwischensumme *Summe* addiert und danach diese neue Zwischensumme ausgegeben.

So geht es weiter, bis *i* irgendwann größer als der Endwert *Ende* ist. Die Schleife wird verlassen und mit *For_Loop = Summe* das Endergebnis als Funktionswert übergeben.

34.2.2 For Each...Next-Schleifen

Die *For Each...Next*-Schleife ist eine Sonderform der *For..Next*-Schleife und zum komfortablen Durchlaufen von Arrays und den später erläuterten Klassen gedacht.

Sie spricht automatisch der Reihe nach *alle* Elemente eines Arrays oder einer solchen Klasse an.

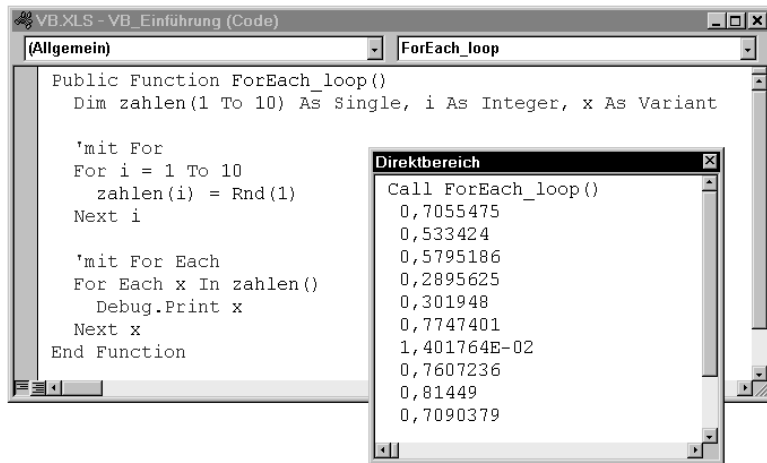
```
For Each Element In Gruppe  
    [Block]  
Next Element
```

Listing 34.16:
For Each...Next-
Schleife

Element ist wieder eine Schleifenvariable, die – bei Arrays – unbedingt vom Typ *Variant* sein muss. *Gruppe* ist das zu durchlaufende Array bzw. die Klasse. Ein praktisches Beispiel (Abbildung 34.6) sehen Sie hier:

Abbildung 34.6:

Vb.xls, Modul
»VB_Einführung«,
Prozedur
»ForEach_Loop«



Zahlen ist ein Array aus zehn *Single*-Variablen *zahlen(1)* bis *zahlen(10)*. Die *For*-Schleife spricht diese Variablen der Reihe nach mit *zahlen(i) = Rnd(1)* an. Die Funktion *Rnd(1)* erzeugt eine Zufallszahl zwischen 0 und 1, das heißt nach Verlassen der Schleife besitzt jede der zehn Array-Variablen einen Wert zwischen 0 und 1.

Um das zu belegen, werden die Inhalte der zehn Variablen anschließend im Direktfenster ausgegeben. Dazu können Sie erneut eine *For...Next*-Schleife verwenden, *For Each...Next* ist jedoch bequemer. Der Ausdruck

```
For Each x in zahlen()
```

bewirkt, dass automatisch der Reihe nach alle Elemente des Arrays *zahlen()* angesprochen werden. *x* repräsentiert dabei das jeweils aktuelle Element, sodass *Debug.Print x* den Inhalt dieses Elements im Direktfenster ausgibt.

Der Vorteil gegenüber *For...Next* ist, dass Sie sich nicht darum kümmern müssen, welchen Index das erste bzw. das letzte Element des Arrays besitzt, sondern Sie können davon ausgehen, dass in jedem Fall alle Elemente angesprochen werden, vom ersten bis zum letzten.

Das ist vor allem dann vorteilhaft, wenn einer Funktion ein Array übergeben wird und diese Funktion zunächst überhaupt nicht weiß, wie groß es ist. Die Funktion könnte die Array-Grenzen zwar mit den Funktionen *LBound* und *UBound* ermitteln, *For...Each* erspart Ihnen jedoch diesen zusätzlichen Aufwand.

34.2.3 While- und Do-Schleifen

For-Schleifen können Sie nur einsetzen, wenn Sie exakt wissen, wie oft eine Schleife zu durchlaufen ist. Sonst müssen Sie eine *While*-Schleife verwenden.

While-Schleife

```
While Aussage
    [Block]
Wend
```

Listing 34.17:
While-Schleife

Die Anweisungen zwischen *While* und *Wend* werden wiederholt, solange *Aussage TRUE* ist, eine beliebige Aussage, die wie $a < 1$ einfach oder wie $(a = 1 \text{ And } b = 10) \text{ Or } x\$ = \text{»Hallo«}$ komplexer sein kann.

**Schleifen-
bedingung**

```
Dim x As Integer
```

```
x = 1
While x < 11
    Debug.Print x
    x = x + 1
Wend
```

Listing 34.18:
Beispiel für eine
While-Schleife

Dieses Programm gibt der Reihe nach die Zahlen 1 bis 10 aus.

x erhält zu Beginn den Wert 1. Die Bedingung *While $x < 11$* (»solange x kleiner als 11 ist«) ist erfüllt und die Anweisungen in der Schleife werden ausgeführt: *Debug Print x* gibt den aktuellen Inhalt von x aus, also 1, und $x = x + 1$ erhöht x um 1.

Da die Bedingung »kleiner als 11« immer noch erfüllt ist, wird die Schleife erneut durchlaufen.

Dieses Spielchen wiederholt sich, bis x den Wert 10 besitzt. 10 wird ausgegeben, x um 1 erhöht (ergibt 11) und die Schleife verlassen, da die Bedingung nicht mehr erfüllt ist.

Do-Schleifen stellen eine Art Verallgemeinerung der *While*-Schleife dar und machen diese eigentlich überflüssig. Wie bei der *While*-Schleife gibt es eine Schleifenbedingung, die getestet wird. Die Ausführung der Anweisungen innerhalb der Schleife hängt vom Ausgang dieses Tests ab.

Do-Schleife

Im Gegensatz zu *While* können Sie jedoch wählen, ob am Anfang oder am Ende der Schleife getestet wird.

```
1. Test am Schleifenanfang: Do [{While | Until} Aussage]
                             [Block]
                             Loop
```

Listing 34.19:
Testen am
Schleifenanfang/
Schleifenende

Listing 34.19:
Testen am
Schleifenanfang/
Schleifenende
(Forts.)

2. Test am Schleifenende:

```
Do
    [Block]
Loop [{While | Until} Aussage]
```

While/Until

In beiden Fällen befindet sich vor der zu testenden Aussage wahlweise eines der beiden Schlüsselwörter *While* oder *Until*.

Mit *While* wird die Schleife ausgeführt, wenn die Aussage *TRUE* ist, und beendet, wenn sie *FALSE* ist. Mit *Until* gebildete Schleifen werden genau umgekehrt nur dann ausgeführt, wenn die Aussage *FALSE* ist und beendet, wenn sie *TRUE* ist.

Listing 34.20:
Do While und
Do Until

```
Dim x As Integer

x = 1
Do While x < 11
    Debug.Print x
    x = x + 1
Loop
```

```
Dim x As Integer

x = 1
Do Until x < 11
    Debug.Print x
    x = x + 1
Loop
```

Do While x < 11 könnte man mit »wiederhole, wenn *x* kleiner als 11 ist« übersetzen. Die Schleife wird zehnmal ausgeführt. Immer wieder wird *x* um 1 erhöht und der aktuelle Wert ausgegeben – eben solange, wie *x* kleiner als 11 ist.

Im Gegensatz dazu wird die Schleife rechts davon überhaupt nicht ausgeführt. Denn *Do Until x < 11* bedeutet »wiederhole, bis *x* kleiner als 11 ist« oder anders formuliert »wiederhole, außer wenn *x* kleiner als 11 ist«.

Wenn die Bedingung erfüllt und *x* kleiner als 11 ist, wird die Schleife nicht ausgeführt, und das ist ja bereits von vornherein der Fall – *x* ist bereits zu Beginn kleiner als 11, die Schleife wird daher sofort übergangen.



Ob Sie While oder Until verwenden, ist letztlich reine Geschmackssache. Nehmen wir an, die Zahlenfolge 1, 2, 3, 4, 5 etc. soll immer weiter addiert werden, solange die Summe noch kleiner als 400 ist.

Eine entsprechend mit *While* formulierte Schleifenbedingung der Art »wiederhole, solange *Summe* kleiner ist als 400« (*Do While Summe < 400*) können Sie jederzeit äquivalent mit *Until* formulieren, indem Sie einfach die Bedingung umkehren (negieren): »wiederhole, bis *Summe* größer oder gleich 400 ist« (*Do Until Summe >= 400*).

Das ist die *While*-Variante, die die Zahlenfolge addiert, solange die Summe kleiner als 400 ist:

```
Dim Summe As Integer, Betrag As Integer
```

```
Summe = 0
Betrag = 1
Do While Summe < 400
    Summe = Summe + Betrag
    Betrag = Betrag + 1
Loop
Debug.Print "Die Summe ist: "; Summe
```

Listing 34.21:

Do While

Hier sehen Sie die *Until*-Variante, die die Zahlenfolge addiert, bis die Summe größer oder gleich 400 ist:

```
Dim Summe As Integer, Betrag As Integer
```

```
Summe = 0
Betrag = 1
Do Until Summe >= 400
    Summe = Summe + Betrag
    Betrag = Betrag + 1
Loop
Debug.Print "Die Summe ist: "; Summe
```

Listing 34.22:

Do Until

Keine Geschmacksfrage ist jedoch die Wahl des Testortes. Meist ist es sinnvoll, die Bedingung bereits am Schleifenanfang zu testen.

Testort

Stellen Sie sich eine Textdatei vor, in der eine bestimmte Zeile gesucht wird, zum Beispiel »Dies ist ein Test«. Zeile für Zeile der Datei wird eingelesen und überprüft. Der Suchvorgang wird solange wiederholt, bis die Zeile gefunden – oder das Dateiende erreicht ist.

Denn möglicherweise ist die gesuchte Zeile überhaupt nicht in der Datei enthalten und unser Programm würde über das Dateiende hinaus suchen wollen. Was dann passiert, können wir nicht hundertprozentig voraussagen, höchstwahrscheinlich ist jedoch ein Programmabsturz die Folge. Die Schleifenbedingung lautet daher »suche, bis die Textzeile »Dies ist ein Test« gefunden oder aber das Dateiende erreicht ist«.

In einem Pseudocode formuliert sieht das so aus:

```
Do Until »"Dies ist ein Test" gefunden« Or »Dateiende erreicht«
    'nächste Zeile holen und prüfen,
    'ob es die gesuchte Zeile ist
Loop
```

Listing 34.23:

Test am Schleifenanfang

Die Suchanweisungen werden wiederholt, bis die Zeile »Dies ist ein Test« gefunden ist oder aber das Programm erkennt, dass das Ende der Datei erreicht ist.

In diesem Beispiel *muss* der Test am Schleifenanfang stattfinden, da folgender Fall denkbar ist: Die Datei ist leer und enthält nicht eine Textzeile.

Dann darf der Suchvorgang kein einziges Mal ausgeführt werden, denn das Dateiende ist ja bereits von vornherein erreicht und mit dem Dateianfang identisch! Und wie gesagt muss unbedingt verhindert werden, dass das Programm versucht, über das Dateiende hinauszugreifen.

Beim Test am Schleifenende wird das Programm im Spezialfall einer leeren Datei fehlerhaft arbeiten. Denn mit dieser Konstruktion wird die Schleife auf alle Fälle einmal durchlaufen und erst am Schleifenende die Bedingung getestet. Dann ist es jedoch möglicherweise bereits zu spät und das Programm über das Dateiende hinausgeschossen.

34.2.4 Abbruch einer Schleife oder Prozedur

Mit der Anweisung *Exit* können *For*- und *Do*-Schleifen ebenso wie Funktions- und Unterprozeduren vorzeitig beendet werden, bei Schleifen also noch bevor die eigentliche Abbruchbedingung erfüllt ist, zum Beispiel bevor der Schleifenzähler in einer *For*-Schleife den Endwert überschreitet:

`Exit {For | Do | Function | Sub}`

For- und *For Each*-Schleifen werden mit *Exit For*, *Do*-Schleifen mit *Exit Do*, Funktionsprozeduren mit *Exit Function* und Unterprozeduren mit *Exit Sub* abgebrochen. *Exit* darf an beliebiger Stelle innerhalb der Schleife/Prozedur auftauchen.

Listing 34.24:
Exit For

```
Dim i As Integer
For i = 1 To 10
    Debug.Print "Hallo"
    If i = 5 Then Exit For
Next i
```

Diese Schleife wird nicht zehnmal, sondern nur fünfmal durchlaufen. Wenn die Schleifenvariable *i* den Wert 5 annimmt, beendet *Exit For* die Schleife.

Das Beispiel ist zugegebenermaßen sehr konstruiert. Sinnvoller kann *Exit Do* in einer *Do*-Schleife beim erwähnten Suchen in einer Datei nach der Textzeile »Dies ist ein Test« angewendet werden. Normalerweise wird immer weiter gesucht, bis die Zeile gefunden wird. Allerdings darf es keinesfalls passieren, dass das Dateiende nicht erkannt und darüber hinaus gesucht wird.

Ein Fall für *Exit Do*:

```
Do
    If »Dateiende erreicht« Then Exit Do
    'Suchoperation
    'ausführen
Loop Until »Name ist gefunden«
```

Listing 34.25:
Exit Do

In einer Schleife wird in einer Textdatei gesucht, bis die gewünschte Zeile gefunden wird. Die erste Anweisung in der Schleife behandelt den Fall, dass das Dateiende erreicht wird. Dann muss die Suche vorzeitig abgebrochen werden, wofür sich *Exit Do* anbietet.

Analog dazu kann eine Funktionsprozedur mit *Exit Function* bzw. eine Sub-Prozedur mit *Exit Sub* vorzeitig abgebrochen werden, wenn eine bestimmte Bedingung erfüllt ist, zum Beispiel, wenn eine Zahl *x* durch eine Zahl *y* geteilt werden soll, die übergebene Zahl *y* jedoch 0 ist, wodurch eine Division bekanntlich nicht möglich ist.

```
Function Test (x As Single, y As Single)
    If y = 0 Then
        Debug.Print "Sorry, Division durch 0 ist nicht möglich"
        Exit Function
    End If
    Test = x / y
End Function
```

Listing 34.26:
Exit Function

Wird für *y* der Wert 0 übergeben, bricht *Exit Function* die Funktionsausführung nach der Ausgabe einer entsprechenden Meldung vorzeitig ab und die folgende Divisionsanweisung wird nicht ausgeführt.

34.3 Laufzeitfehler behandeln

In Programmen treten häufig Laufzeitfehler auf, die jedoch abgefangen werden können. Sie kennen bestimmt die berühmte Fehlermeldung »Abbrechen, Wiederholen, Ignorieren?«, wenn Sie auf ein Diskettenlaufwerk wechseln wollen, das keine Diskette enthält oder nicht verriegelt ist.

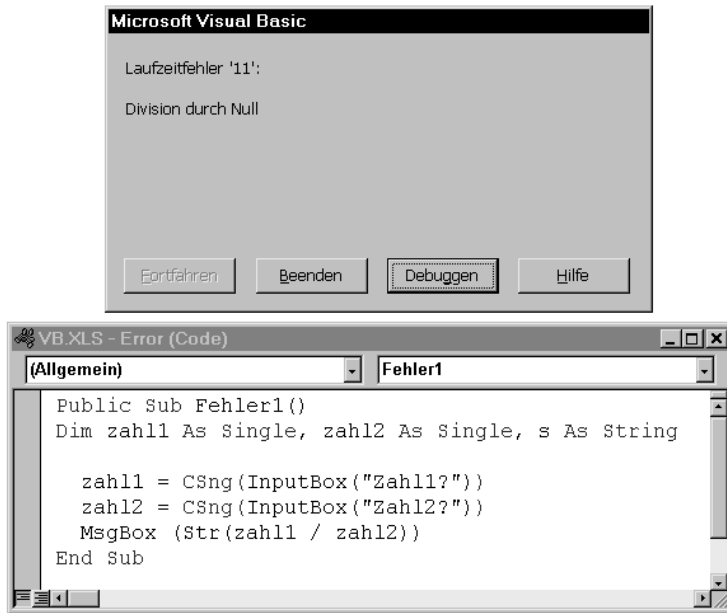
Der gleiche Fehler tritt auch in einem VBA-Programm auf, wenn diese Möglichkeit nicht behandelt wird. Nicht in Form der Original-DOS-Fehlermeldung. Den DOS-Fehler fängt seinerseits VBA ab und gibt in einem netten Dialogfeld eine entsprechende Fehlermeldung aus.

Diese VBA-Fehlermeldung können Sie jedoch wiederum abfangen und stattdessen mit der Wiederholung des Zugriffs (ziemlich sinnlos) oder auf eine beliebige andere Art und Weise darauf reagieren.

**Division durch
Null**

Es gibt nahezu beliebig viele weitere Fehlerquellen. Die folgende Abbildung zeigt die im Modul »Error« von VB.XLS enthaltene Prozedur »Fehler1«, die Sie zur Eingabe einer Zahl auffordert, danach zur Eingabe einer weiteren Zahl, und die anschließend die erste durch die zweite Zahl dividiert und mit *MsgBox* das Ergebnis dieser Division ausgibt (Abbildung 34.7).

Abbildung 34.7:
Vb.xls, Modul
»Error«, Prozedur
»Fehler1«



Der genaue Ablauf: Die Anweisung

```
zahl1 = CSng(InputBox("Zahl1?"))
```

erfragt mit *InputBox* eine Zahl und wandelt die von *InputBox* eingegebene Zeichenkette mit *CSng* in einen *Single*-Wert um, der anschließend der Variablen *zahl1* zugewiesen wird. Danach wird nach einer zweiten Zahl gefragt und diese *zahl2* zugewiesen.

Die folgende Anweisung

```
MsgBox (Str (zahl1 / zahl2))
```

teilt *zahl1* durch *zahl2*, wandelt das Ergebnis in eine Zeichenkette und gibt sie mit der *MsgBox*-Funktion aus.

Das Ganze funktioniert, solange Sie als zweite Zahl keine 0 eingeben. Sonst tritt bei dieser Division ein Laufzeitfehler auf, auf den VBA mit der gezeigten Fehlermeldung reagiert.

Klicken Sie auf *Debuggen*, so wird die Codezeile hervorgehoben, in der der Fehler auftrat, und Sie können mit dem Testen beginnen.

Alternativ zu dieser Standardfehlerbehandlung ermöglicht Ihnen VBA mit dem Error-Trapping, der Fehlerverfolgung, auftretende Fehler noch vor der Ausgabe einer Fehlermeldung in Ihrem Programm zu erkennen und in einer speziellen Fehlerroutine individuell zu behandeln.

Kern der Fehlerbehandlung ist die Anweisung *On Error*:

On Error

On Error GoTo Zeile

Zeile ist eine Zeilennummer oder besser eine Zeilenmarke, zu der beim Auftreten eines Fehlers verzweigt wird. Die betreffende Zeile muss sich in der gleichen Prozedur wie die *On Error*-Anweisung befinden.

On Error GoTo Zeile aktiviert die Fehlerverfolgung. Tritt während der Programmausführung ein Laufzeitfehler auf, so erscheint nicht die entsprechende VBA-Fehlermeldung, sondern die Programmausführung wird mit der angegebenen Zeile fortgesetzt.

On Error sollte möglichst am Prozeduranfang stehen – am besten als erste Programmzeile. Denn wenn ein Fehler auftritt, *bevor* die Fehlerverfolgung aktiviert wurde, wird auch nicht zur später angegebenen Zeile verzweigt (Abbildung 34.8).

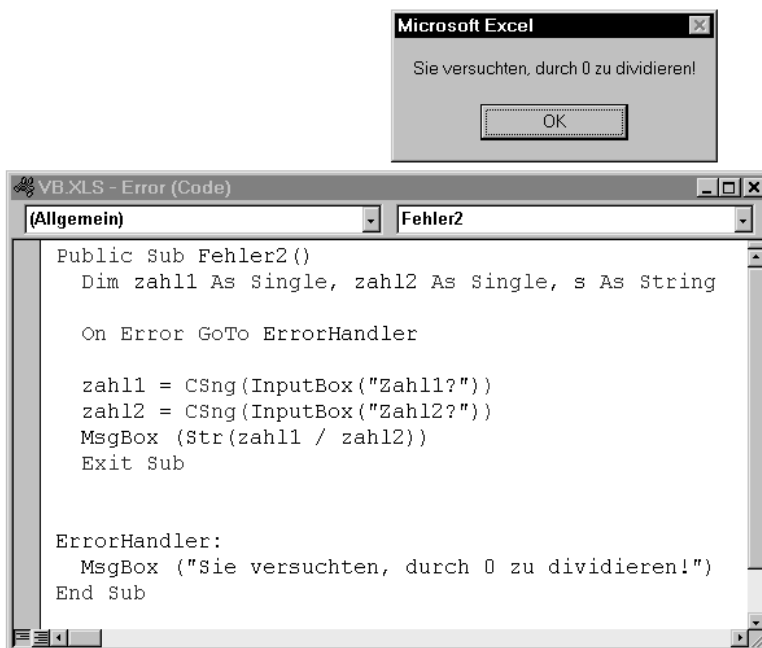


Abbildung 34.8:
Vb.xls, Modul
»Error«, Prozedur
»Fehler2«

On Error GoTo ErrorHandler aktiviert die Fehlerverfolgung und verzweigt beim Auftreten eines Laufzeitfehlers zum Label *ErrorHandler*, das sich in der gleichen Prozedur befindet (befinden *muss*).

Trat ein Laufzeitfehler auf, wird zur Fehlerroutine verzweigt und dort die Anweisung *MsgBox* ausgeführt, die in einem Dialogfeld eine individuelle Meldung ausgibt.



Die Anweisung Exit Sub ist außerordentlich wichtig, um zu verhindern, dass die Programmausführung in die Fehlerroutine hineinläuft und diese ausgeführt wird, auch wenn alles klappte und kein Laufzeitfehler auftrat.

Dass das Programm funktioniert, ist jedoch reiner Zufall: Normalerweise muss ein aufgetretener Fehler korrekt behandelt werden, bevor VBA bereit ist, weitere Fehler zu verfolgen. Die einzige Ausnahme bildet das hier verwendete vorzeitige Verlassen der Prozedur mit *Exit Sub*, an dem VBA erkennt, dass offenbar keine weitere Fehlerbehandlung stattfinden soll.

Im Normalfall muss die Fehlerroutine jedoch mit der Anweisung *Resume* in einer der folgenden Varianten abgeschlossen werden:

- ➔ *Resume* allein setzt die Programmausführung mit der Anweisung fort, durch die der Fehler verursacht wurde, im Beispiel also mit *MsgBox (Str([zahl1] / [zahl2]))*.
- ➔ *Resume Next* setzt die Ausführung mit der darauf folgenden Anweisung fort, im Beispiel also mit *Exit Sub*.
- ➔ *Resume Zeile* setzt das Programm ab der angegebenen Zeilenmarke fort.
- ➔ Die Anweisung *Error Err* erzeugt selbst den angegebenen Laufzeitfehler Nummer *Err* (Beispiele dazu am Ende dieses Kapitels). VBA versucht daraufhin, bei verschachtelten Prozeduren entlang des Aufrufpfades (in der Prozedur B, die die aktuelle Prozedur C aufrief, danach in der Prozedur A, die wiederum B aufrief etc.) eine weitere Fehlerroutine zu finden und führt diese aus, falls vorhanden.

Resume allein verzweigt zu jener Anweisung, die den Fehler verursachte. In einer etwas aufwändigeren Fehlerroutine würde eine Meldung ausgegeben, die den Benutzer auffordert, den Fehler zu beheben. Anschließend wird aus der Fehlerroutine mit *Resume* zu jener Anweisung zurückverzweigt, die den Fehler verursachte, und die nun – falls der Anwender den Fehler tatsächlich behoben hat – problemlos ausgeführt wird.



Resume allein ist jedoch ungeeignet, wenn es für den Benutzer eventuell unmöglich ist, einen aufgetretenen Fehler zu beheben. In solchen Fällen – wir geben dem Benutzer Gelegenheit zur Fehlerbeseitigung, er besitzt jedoch keine Möglichkeit dazu – führt Resume zu einer Endlosschleife. Eine vernünftige Weiterarbeit mit dem Programm ist nicht mehr möglich.

Resume Next verzweigt aus der Fehlerroutine wie erläutert zu jener Anweisung in die Prozedur zurück, die der Fehler verursachenden Anweisung folgt, im Beispiel zu *Exit Sub*. Höchst interessant ist, dass Sie *Resume Next* mit der Aktivierung der Fehlerverfolgung *On Error* kombinieren können, was oft eine besonders elegante Programmstruktur ergibt:

On Error Resume Next

Die Aktivierung der Fehlerbehandlung enthält in diesem Fall zugleich die Anweisung, beim Auftreten eines Fehlers zur folgenden Anweisung zu verzweigen, die Programmausführung also einfach fortzusetzen.

Das heißt jedoch, dass Sie in dieser folgenden Zeile unterschiedlich reagieren müssen, abhängig davon, ob in der vorhergehenden Zeile ein Fehler auftrat oder nicht.

Also benötigen Sie eine Möglichkeit, diesen Sachverhalt zu überprüfen. Dazu verwenden Sie das Fehler-Objekt *Err*.

Die vom Ausdruck *Err.Number* übergebene Eigenschaft *Number* dieses Objekts ist der Laufzeit-Fehlercode, eine Zahl, die auf jeden Fall ungleich Null ist und Auskunft über die Art des Fehlers gibt. *Err.Number* kann zusammen mit *On Error Resume Next* eingesetzt werden, um einen Fehler sofort nach dem Auftreten in der darauf folgenden Zeile zu behandeln (Abbildung 34.9).

Err.Number

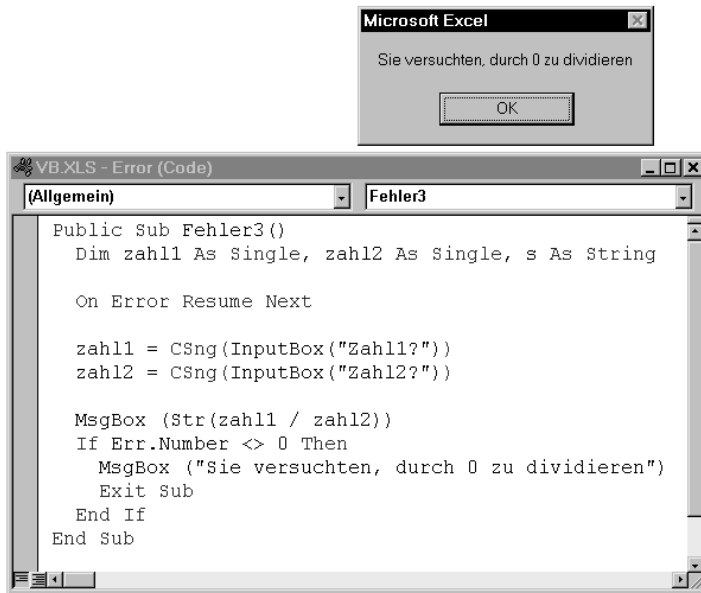
Tritt bei der Division kein Fehler auf, ist *Err.Number* 0. Die Bedingung *If Err.Number <> 0* ist nicht erfüllt, der *Then*-Zweig wird nicht ausgeführt und die Prozedur beendet.

Trat jedoch ein Fehler auf, verzweigt *On Error Resume Next* zur folgenden Zeile, also ebenfalls zu *If Err.Number <> 0 ...*. Diesmal ist *Err.Number* jedoch ungleich 0 und die Fehlerroutine im *Then*-Zweig wird ausgeführt. *MsgBox* gibt eine entsprechende Meldung aus, danach wird die Prozedur mit *Exit Sub* beendet (nur der Klarheit wegen, die folgende Anweisung *End Sub* würde sie ebenfalls beenden).h

Statt *If Err.Number <> 0* können Sie einfach *If Err.Number* schreiben.

Abbildung 34.9:

Vb.xls, Modul
»Error«, Prozedur
»Fehler3«



Err.Number ermöglicht sehr spezifische Reaktionen, abhängig von der Art des aufgetretenen Fehlers. Nehmen wir als Beispiel für die Anwendung von *Err.Number* das bereits vorgestellte Programm zum Dividieren zweier Zahlen. VBA erzeugt dabei, je nach Situation, zwei verschiedene Laufzeitfehler:

- ➔ Ist der Nenner 0, wird der Laufzeitfehler »Division durch Null« erzeugt, der den Fehlercode 11 besitzt.
- ➔ Sind Zähler und Nenner 0, wird ein Überlauffehler erzeugt, der den Fehlercode 6 besitzt.

Durch Überprüfen dieses Codes können Sie die möglichen Fehler individuell behandeln:

Listing 34.27:
Individuelle Fehler-
behandlung

```
...
If Err Then
    Select Case Err.Number
        Case 11
            MsgBox ("Sie versuchten, durch 0 zu dividieren!")
        Case 6
            MsgBox ("Sie versuchten, 0 durch 0 zu dividieren!")
        Case Else
            MsgBox ("Ein nicht näher bekannter Fehler trat auf!")
        End Select
    Else
        MsgBox ("Alles okay!")
    End If
...
```

Bei einer solchen spezifischen Fehlerbehandlung soll der Case Else-Zweig nicht vergessen werden. In einem komplexen Programm werden immer wieder Anwenderfehler auftreten, die der Programmierer nicht vorhersah. Diese Fehler behandelt der Case Else-Zweig, der für alles Übrige zuständig ist.



Die Funktion *Error* bietet eine weitere Möglichkeit, Sie selbst oder den Anwender über die Art des aufgetretenen Fehlers zu informieren. *Error* übergibt als Funktionswert die Fehlermeldung im Klartext, die dem übergebenen Fehlercode zugeordnet ist. Entsprechend informiert in einer Fehlerroutine der Ausdruck *MsgBox (Error(Err.Number))* im Klartext über den aufgetretenen Fehler.

Error

Nach der Behandlung eines Fehlers mit Resume oder der Ausführung einer Exit Sub- oder Exit Function-Anweisung in einer Fehlerroutine werden alle Fehlercodes auf 0 zurückgesetzt, sind also nicht mehr verfügbar. Das gleiche kann passieren, wenn eine Fehlerroutine eine andere Prozedur aufruft, sodass zuvor der Fehlercode einer Variablen zugewiesen werden sollte.



In der Praxis taucht bei der Erstellung von Fehlerrouninen ein Problem auf: Die Fehlerroutine fängt alle Fehler ab, wobei auf einige – häufiger auftretende – in besonderer Weise reagiert wird. Wie testen wir jedoch, ob unsere Fehlerroutine auch dann richtig arbeitet, wenn ein Fehler während der Testläufe nicht vorgetäuscht werden kann?

Fehler simulieren

Die meisten Fehler können Sie problemlos simulieren, indem Sie zum Beispiel den Drucker ausschalten oder einen Schreibschutz auf die Diskette kleben.

Einige Fehler, zum Beispiel ein defektes Disketten- bzw. Festplattenlaufwerk, können Sie jedoch nicht per Hand simulieren. Außer, Sie sind bereit, die Auswirkung eines Schlags mit dem Hammer an Ihrer Festplatte zu erproben – danach tritt der Fehler ganz bestimmt auf!

In derartigen Fällen wenden Sie zur Simulation des Fehlers mit dem Ausdruck *Err.Raise Fehlercode* die Methode *Raise* auf das *Err*-Objekt an (Abbildung 34.10).

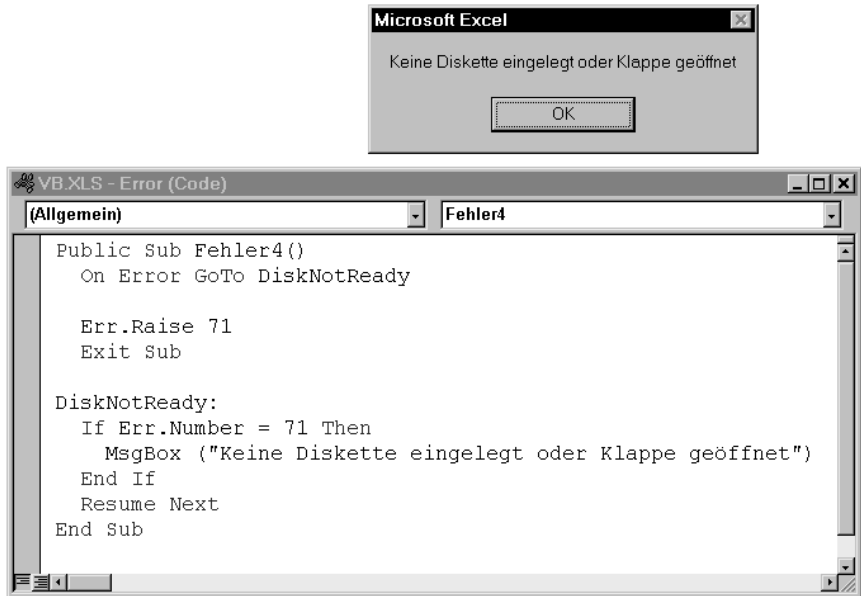
Err.Raise 71 simuliert das Auftreten des Fehlers »Diskette nicht bereit« mit der Nummer 71. VBA verzweigt zur aktivierten Fehlerroutine und dort wird – nach Überprüfung der Fehlernummer – mit einer entsprechenden Meldung reagiert.

Err.Raise können Sie außer zur Fehlersimulation auch zur Fehlerdefinition verwenden.

Fehlerdefinition

Abbildung 34.10:

Vb.xls, Modul
»Error«, Prozedur
»Fehler4«



Ein Beispiel: An mehreren Stellen in Ihrem Programm soll der Benutzer eine Postleitzahl eingeben. Postleitzahlen bewegen sich normalerweise im Bereich zwischen 10000 und 99999. Eingaben außerhalb dieses Zahlenbereichs sind fehlerhaft, werden aber selbstverständlich nicht als Fehler erkannt. Es sind Fehler im Sinne Ihres Programms, keine allgemeinen Basic-Fehler.

Solchen programmspezifischen Fehlern kann mit der *Err.Raise*-Anweisung ein eigener Fehlercode zugewiesen werden, theoretisch eine beliebige Zahl wie beispielsweise 65535 (Abbildung 34.11).

Laufzeitfehler5 fordert Sie mit der Anweisung *InputBox* auf, in einem Dialogfeld eine Postleitzahl einzugeben.

Liegt Ihre Eingabe nicht im Bereich zwischen 10000 und 99999, geben Sie beispielsweise 2300 ein, simuliert die Anweisung *Err.Raise 65535* das Auftreten eines Fehlers mit der Nummer 65535 und das Programm verzweigt zur Fehlerroutine.

Die Fehlerroutine prüft mit *Err.Number* den Fehlercode. Handelt es sich um den benutzerdefinierten Fehler 65535, gibt sie eine entsprechende Meldung aus und verzweigt anschließend mit *Resume Eingabe*: zum Label *Eingabe*, wiederholt also die Programmausführung ab der Eingabe der Postleitzahl.

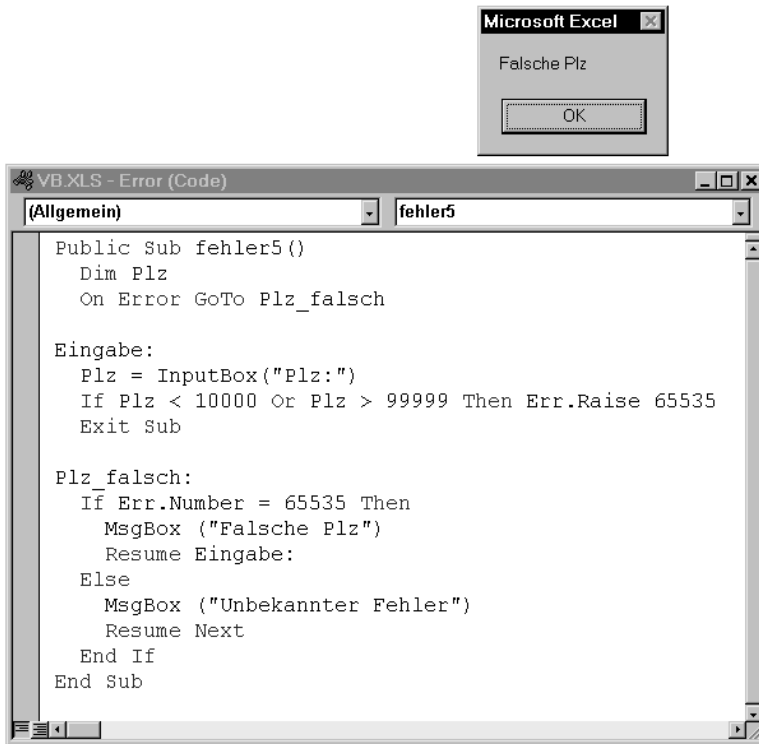


Abbildung 34.11:

Vb.xls, Modul
»Error«, Prozedur
»fehler5«

Um nicht mit VBA-internen Fehlercodes zu kollidieren, sollten Sie eigene Fehlercodes nicht wie im Beispiel einfach willkürlich wählen, sondern unbelegte Fehlercodes verwenden, am besten nur Werte, die größer sind als der höchste von VBA selbst verwendete Fehlercode!



Welcher Code das ist? Keine Ahnung, den genauen Wert müssen Sie aber gar nicht kennen. Wichtig ist nur, dass er sich in der vordefinierten VBA-Konstanten `vbObjectError` befindet. Also benutzen Sie für Ihren ersten selbstdefinierten Fehlercode statt 65535 oder ähnlich einfach den Ausdruck `vbObjectError+1`, für einen weiteren eigenen Fehlercode den Ausdruck `vbObjectError+2` und so weiter.

Der Fehleraktivierungsanweisung `On Error` kann direkt die Fehlerbehandlungsanweisung `Resume Next` folgen, ohne Angabe eines Labels. Dann ist der Fehler für VBA sofort nach dem Auftreten behandelt und die folgende Anweisung wird ausgeführt – die nun allerdings eine echte Fehlerbehandlung vornehmen und danach den Fehlercode mit der Anweisung `Err.Clear` löschen sollte.

Fehler löschen

35 Objekte

Die vorangegangenen Kapitel stellten eine Einführung in die *allgemeine* Programmierung mit VBA dar. Allgemein heißt, dass Sie nun die wichtigsten Elemente dieser Sprache kennen.

Um sinnvolle Excel-VBA-Programme zu schreiben, müssen Sie jedoch zusätzlich die verschiedenen Objekte von Excel kennen, beispielsweise Zellen, Bereiche und Tabellenblätter; erst das versetzt Sie in die Lage, diese Objekte beliebig per VBA-Programm zu manipulieren, beispielsweise um programmgesteuert Werte in Zellen einzutragen oder umgekehrt den in einer Zelle enthaltenen Wert auszulesen, neue Arbeitsblätter anzulegen oder zu speichern und so weiter.

Entsprechend geht es im folgenden Kapitel um das Rüstzeug, das zur Manipulation von Excel-Objekten notwendig ist. Ich erläutere die Objekthierarchie von Excel und zeige Ihnen, wie Sie gewünschte Objekte durch Objektbezüge referenzieren, das heißt ansprechen.

Ich zeige Ihnen, was unter den Eigenschaften eines Objekts zu verstehen ist und wie Objekte durch die Anwendung von Methoden manipuliert werden und wie all diese Manipulationen durch die Anwendung von Objektvariablen erheblich vereinfacht werden.

35.1 Objekttypen

Beim Programmieren in VBA haben Sie es mit Objekt-Bibliotheken zu tun, die Ihnen verschiedene Objekte zur Verfügung stellen, die Sie manipulieren können. Die Excel-Bibliothek stellt Ihnen beispielsweise Objekte wie *Workbooks* (Arbeitsmappen), *Worksheets* (Arbeitsblätter) und *Range* (Bereiche von Arbeitsblättern) zur Verfügung.

All diese Objekte besitzen verschiedene Eigenschaften und können mithilfe verschiedener Methoden manipuliert werden, beispielsweise, um in einer Tabelle einen Bereich zu selektieren oder um eine Tabelle zu löschen.

Nahezu alle Objekte sind Bestandteile eines umfassenderen Objekts, dieses ist wiederum Teil eines noch größeren Objekts und so weiter. Um sich auf ein Objekt zu beziehen, müssen Sie manchmal die Objekthierarchie einhal-

ten und von oben nach unten auf die einzelnen Objekte verweisen, bis das innerste Objekt, mit dem Sie sich beschäftigen wollen, eindeutig identifiziert ist.

Stellen Sie sich das Ganze wie jene netten russischen Puppen vor, die ineinander verschachtelt sind: Sie öffnen die Puppe und was kommt zum Vorschein? Eine darin enthaltene kleinere Puppe – in der sich erneut eine kleinere Puppe befindet und so weiter.

Ganz oben in der Excel-Bibliothek befindet sich das Objekt *Application*, das praktisch die aktuelle Anwendung darstellt, mit der Sie gerade arbeiten, also Excel selbst; und damit ist sie das Oberobjekt, in dem zwar viele andere Objekte enthalten sind, Formulare, Steuerelemente etc., das selbst jedoch einzigartig ist.



Diese Objektschachtelung ist sehr eng mit benutzerdefinierten Datentypen verwandt, bei denen ein selbstdefinierter Datentyp als Komponenten selbst wieder eigendefinierte Datentypen enthalten kann, die selbst wieder mehrere Komponenten enthalten (siehe Kapitel 33.9, »Eigendefinierte Datentypen (Verbunde)«).

Listing 35.1:
Analogie zu
Verbunden

```
'Innerer Verbund
Type AdressOrt
    Plz As Long
    Ort As String * 40
End Type
'Äußerer Verbund

Type Adresstyp
    Vorname As String * 20
    Nachname As String * 30
    Wohnort As AdressOrt
End Type

'Verbundvariable
Dim adr As Adresstyp
```

Die ganz unten deklarierte Variable *adr* ist vom Typ *Adresstyp*, der aus mehreren Komponenten besteht. Eine davon, *AdressOrt*, ist selbst ein selbstdefinierter Datentyp, der aus mehreren Komponenten besteht.

Erinnern Sie sich, wie bei derart verschachtelten Strukturen eine Komponente im Inneren einer anderen Struktur angesprochen wird, beispielsweise die Komponente *Plz* der Variablen *adr*, mit dem Ausdruck

```
adr.AdressOrt.Plz
```

oder allgemein ausgedrückt:

```
Variablenname.Komponente.Komponente.Komponente.....
```

Auf die gleiche Art und Weise wird bei VBA ein Objekt angesprochen, das in einem anderen Objekt enthalten ist, das sich womöglich wiederum in einem Objekt befindet und so weiter:

```
Objekt.Komponente.Komponente.Komponente.....
```

Dadurch können sich endlos lange Ausdrücke ergeben:

```
Application.Workbooks("VB.XLS").Worksheets("Kurse").Activate
```

Er aktiviert die Tabelle »Kurse« der Mappe VB.XLS (vorausgesetzt, diese Mappe ist momentan geöffnet!).

Für derartige Ausdrücke müssen Sie natürlich über die Objekthierarchie Bescheid wissen und Sie müssen wissen, welches Objekt welche Unterobjekte enthält. Ganz oben befindet sich wie erläutert das Objekt *Application*, das alle anderen Excel-Objekte enthält.

Vergessen Sie dieses Objekt vorläufig, da ein Bezug darauf normalerweise überflüssig ist und überflüssige Angaben können in Objektbezügen glücklicherweise entfallen, also auch die Angabe der momentan aktiven Anwendung Excel und somit des Objekts *Application*. Dadurch reduziert sich der gezeigte Ausdruck bereits zu

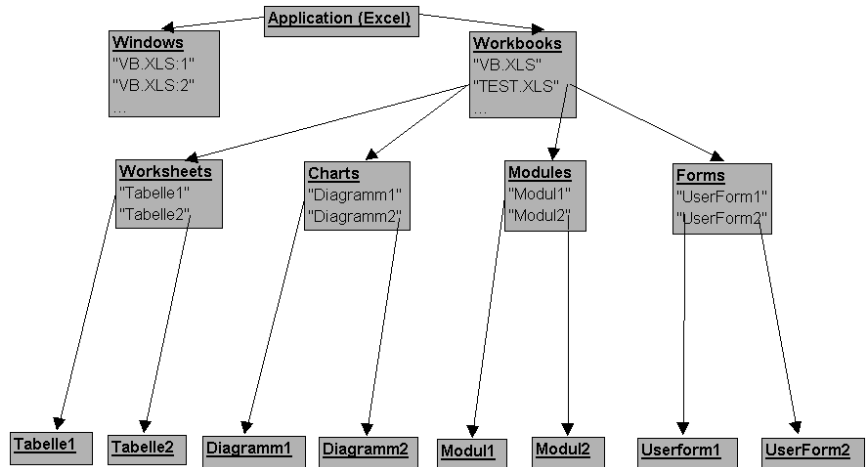
```
Workbooks("VB.XLS").Worksheets("Kurse").Activate
```

Für die Praxis wichtig sind dagegen die Unterobjekte von *Application*, vor allem die verschiedenen Objekt-Auflistungen. Jede Auflistung ist praktisch ein Behälter, der alle momentan verfügbaren Objekte eines bestimmten Typs enthält. Die folgende (vereinfachte) Abbildung soll das Prinzip verdeutlichen (Abbildung 35.1).

Beispielsweise enthält die Auflistung *Windows* eine Liste aller momentan geöffneten Fenster. Entsprechend enthält die Auflistung *Workbooks* eine Liste aller momentan geöffneten Excel-Mappen.

Oft enthält ein einzelnes Objekt einer Auflistung selbst wiederum derartige Behälter. Beispielsweise enthält jede Excel-Mappe, also jedes Objekt, das in der Auflistung *Workbooks* enthalten ist, je eine Auflistung *Worksheets*, die die Arbeitsblätter der betreffenden Mappe darstellt, und die Auflistungen *Modules* oder *Charts*, die die in der Mappe enthaltenen Modul- bzw. Diagrammblätter darstellen.

Abbildung 35.1:
Vb.xls, Tabelle
»Objekte«



Die einzelnen in diesen Auflistungen enthaltenen Objekte heißen entsprechend *Worksheet* (ein einzelnes Tabellenblatt), *Modul* (ein einzelnes Modul) und *Chart* (ein einzelnes Diagramm) – und enthalten selbst wiederum Objekte, nämlich die einzelnen Elemente, aus denen das betreffende Objekt besteht.

Beispielsweise enthält ein *Worksheet*-Objekt, also ein Arbeitsblatt, selbst wiederum Objekte wie Zellen, Bereiche etc. Und ein Diagramm enthält Elemente wie die Diagrammfläche, die Achsen etc.

!!
STOP

Beachten Sie, dass die Auflistung Charts keineswegs alle Diagramme enthält, sondern nur Diagrammblätter, also nur Diagramme, die auf eigenen Blättern dargestellt werden!

Nun verstehen Sie auch den Ausdruck

```
Application.Workbooks("VB.XLS").Worksheets("Kurse").Activate.
```

Activate besagt, dass irgend etwas aktiviert werden soll. Und zwar bezieht sich diese Anweisung auf den vorangehenden Teil des Ausdrucks, auf *Worksheets(»Kurse«)*. Dieser Teil besagt, dass es um das Objekt *Kurse* geht, das sich in der Auflistung *Worksheets* befindet.

Nun weiß VBA, dass ein Arbeitsblatt namens *Kurse* aktiviert werden soll, aber noch nicht, wo sich dieses Blatt befindet, in welcher Mappe.

Daher wird VBA zuvor mit dem Ausdruck *Workbooks(»VB.XLS«)* klar gemacht, dass sich das betreffende Blatt im Objekt *VB.XLS* befindet, das sich selbst wiederum in der Auflistung *Workbooks* befindet, also eine der momentan geöffneten Arbeitsmappen ist.

Und diese Mappe befindet sich wiederum im Oberobjekt *Application*, also in der aktuellen Anwendung (in diesem Fall: Excel).

Die folgende Tabelle soll Ihnen einen kleinen Überblick über die am häufigsten benutzten Objekte verschaffen:

Objekttyp	Bedeutung
Application	Die aktuelle Anwendung, praktisch Excel selbst
Windows	Eine Liste der verschiedenen momentan geöffneten Fenster
Window	Ein einzelnes Fenster
Sheets	Eine Liste aller momentan verfügbaren Tabellen-, Diagramm- und sonstiger Blätter
Workbooks	Eine Liste der momentan geöffneten Arbeitsmappen
Workbook	Eine einzelne Mappe
Worksheets	Eine Liste der in einer Mappe enthaltenen Tabellenblätter
Worksheet	Ein einzelnes Tabellenblatt
Modules	Eine Liste der in einer Mappe enthaltenen Modulblätter
Modul	Ein einzelnes Modulblatt
Charts	Eine Liste der in einer Mappe enthaltenen Diagramme
Chart	Ein einzelnes Diagrammblatt
Dialogs	Eine Liste der in einer Mappe enthaltenen Dialoge
Dialog	Ein einzelnes Dialogblatt
Name	Definierte Namen wie »Druckbereich«, »MWSt« etc.
Range	Bereiche wie eine einzelne Zelle, eine Zeile/Spalte, ein Mehrfachauswahl-Bereich, ein 3D-Bereich etc.
Cells	Eine einzelne Zelle

Tabelle 35.1:
Beispiele für
Objekttypen

Um ein bestimmtes Blatt einer Auflistung anzusprechen, verwenden Sie folgenden Ausdruck:

```
Auflistung("Name_des_Objekts")
```

Beispielsweise spricht der Ausdruck

```
Worksheets("Tabelle1")
```

das Tabellenblatt »Tabelle1« an.

Einen Ausdruck wie *Worksheets(»Tabelle1«)*, der einen Bezug auf ein Objekt darstellt, im Beispiel auf eine Tabelle, nennt man einen Objektbezug oder eine Objektreferenz. Entsprechend spricht man davon, ein bestimmtes Objekt zu referenzieren.

Intern enthält ein solcher Objektbezug eine Adresse und zwar jene Adresse, an der sich das betreffende Objekt im Speicher befindet. Eine spezielle Art von Zeiger ist der Null-Zeiger, der auf gar kein Objekt verweist, ähnlich wie bei Zeichenketten eine leere Zeichenkette »" "«.

35.2 Eigenschaften und Methoden

Auf Objekte können Methoden angewendet werden. Methoden sind nichts anderes als objektspezifische Prozeduren, die mit einem bestimmten Objekttyp etwas anstellen und oft wie Funktionen zusätzlich einen Wert zurückgeben.

Um eine Methode anzuwenden, wird ein Bezug auf das betreffende Objekt benötigt, gefolgt von einem Punkt und dem Namen der Methode:

`Objekt.Methode`

Beispielsweise kann auf alle Arbeitsmappen und auf alle Blätter die Methode *Activate* angewendet werden, um das betreffende Blatt zum aktiven Blatt zu machen. Da das *Workbooks*-Objekt eine Liste aller momentan geöffneten Mappen enthält, aktiviert daher ein Ausdruck der Art

`Workbooks(Mappenname).Activate`

die angegebene Mappe. Beispielsweise aktiviert

`Workbooks("Demo.xls").Activate`

die Mappe DEMO.XLS.

Befindet sich darin eine Tabelle namens »Tabelle1«, die nun aktiviert werden soll, verwenden Sie dazu den Ausdruck

`Worksheets("Tabelle1").Activate`

Das Objekt *Worksheets* enthält alle in der aktiven Mappe verfügbaren Tabellenblätter. *Worksheets(»Tabelle1«)* ist daher ein Bezug auf das Blatt »Tabelle1« dieser Mappe, auf das nun die *Activate*-Methode angewendet wird.

Jeder Objekttyp und damit auch alle einzelnen Objekte vom betreffenden Typ besitzen bestimmte Eigenschaften. Um an eine solche Eigenschaft heranzukommen, geben Sie einen Bezug auf das Objekt an, gefolgt von einem Punkt und dem Namen der Sie interessierenden Eigenschaft, benutzen also einen Ausdruck der Art

```
Objekt.Eigenschaft
```

Nehmen wir an, Sie interessieren sich für den Namen der Tabelle »Tabelle1« der aktiven Mappe. Der Objekttyp *Worksheet* besitzt ebenso wie die Typen *Chart* oder *Modul* die Eigenschaft *Name*, die den Namen des Blatts als Zeichenkette übergibt. Entsprechend übergibt der Ausdruck

```
Worksheets("Tabelle1").Name
```

den Namen »Tabelle1« – ziemlich sinnlos, da Sie den Namen dieser Tabelle unweigerlich kennen, wenn Sie ihn im Objektbezug angeben können. Interessanter ist bereits der Ausdruck

```
ActiveSheet.Name
```

ActiveSheet übergibt einen Bezug auf das momentan aktive Blatt. Auf dieses Objekt wird nun die Methode *Name* angewendet, die den Namen dieses Blatts übergibt, »Tabelle1«, »Tabelle2« oder ähnlich; beziehungsweise einen Null-Zeiger, falls gerade kein Tabellenblatt aktiv ist, sondern beispielsweise ein Modulblatt. Die Anweisung

```
Debug.Print ActiveSheet.Name
```

würde den Namen entsprechend im Direktfenster ausgeben.

Angenommen, Sie interessieren sich außer für den Namen des aktiven Tabellenblatts zusätzlich auch für den Namen des folgenden Blatts der Mappe. Alle Blattobjekte besitzen unter anderem die Eigenschaft *Next*, die einen Bezug auf das nächste Blatt der Mappe übergibt. Der Ausdruck

```
ActiveSheet.Next
```

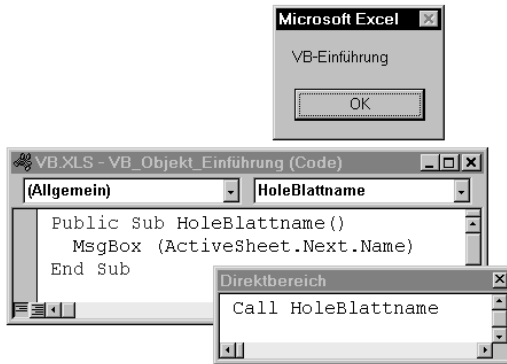
referenziert daher das nächste Blatt und kann somit stellvertretend für dieses Blatt verwendet werden. Also übergibt

```
ActiveSheet.Next.Name
```

den Namen dieses folgenden Blatts. Eine entsprechende Prozedur würde so aussehen (Abbildung 35.2):

Abbildung 35.2:

Vb.xls, Modul
»VB_Objekt_Einführung«, Prozedur
»HoleBlattname«



Lese- und Schreibzugriff

Übrigens ist das Gegenstück zu *Next* die Eigenschaft *Previous*, die den Vorgänger des aktiven Blatts referenziert.

Objekteigenschaften sind entweder echte Eigenschaften wie die Standardhöhe eines Tabellenblatts oder sein Name oder aber Bezüge auf ein im betreffenden Objekt enthaltenes Unterobjekt. Bei echten Eigenschaften können Sie keineswegs immer nur lesend auf die Eigenschaft zugreifen wie in den bisherigen Beispielen.

Viele Eigenschaften erlauben auch einen Schreibzugriff, zum Beispiel die Eigenschaft *Name* der verschiedenen Blattobjekte. Beispielsweise ändert der Ausdruck

```
ActiveSheet.Name = "Test"
```

den Namen des aktiven Blatts der momentan aktiven Mappe und gibt ihm den Namen »Test«.



Allgemein: Ein Ausdruck der Art

```
v = Objekt.Eigenschaft
```

liest den Wert einer Eigenschaft und weist ihn beispielsweise einer Variablen zu. Ein Ausdruck der Art

```
Objekt.Eigenschaft = Wert
```

weist genau umgekehrt einer Eigenschaft den angegebenen Wert zu.

Eine Methode kann einen Bezug übergeben. Beispielsweise übergibt die Methode *Cells*, die auf Tabellenblätter angewendet wird, einen Bezug auf ein Objekt des Typs *Range*, das heißt auf einen Bereich einer Excel-Tabelle:

```
Objekt.Cells(Zeilenindex, Spaltenindex)
```

Zeilenindex ist eine Zeilennummer und *Spaltenindex* nicht etwa ein Spaltenbuchstabe, sondern eine Spaltennummer, wobei Spalte A den Index 1 besitzt, Spalte B den Index 2 etc. Entsprechend übergibt der Ausdruck

```
ActiveSheet.Cells(1, 2)
```

ein aus genau einer Zelle (B1) bestehendes Objekt des Typs *Range*. Dieser Objekttyp besitzt verschiedene Eigenschaften, unter anderem die Eigenschaft *Value*. Der Ausdruck

```
ActiveSheet.Cells(1, 2).Value = 10
```

weist daher der Zelle B1 des aktiven Blatts den Wert 10 zu, so als hätten Sie die Zelle selektiert und diesen Wert per Hand eingetippt.

Falls Ihnen die objektorientierte Programmierung bisher fremd war, sind Sie inzwischen wahrscheinlich recht verwirrt und Ihnen fehlt der Überblick über die verschiedenen Objekte, ihre Eigenschaften und die auf sie anwendbaren Methoden.

Für eine detaillierte Erläuterung der Objekte, Methoden und Eigenschaften benötigen Sie die Online-Hilfe, in der Sie beispielsweise nach dem Stichwort »Worksheets« suchen können, um eine Erläuterung dieser Auflistung zu erhalten.

Darin finden Sie wiederum Verweise auf das Objekt *Worksheet*, also auf einen Text, der erläutert, was ein einzelnes Objekt dieses Typs ist, und zusätzliche Verweise auf die auf dieses Objekt anzuwendenden Eigenschaften und Methoden.

Zum Gewinnen eines groben Überblicks gibt es noch ein weitere Möglichkeit: Sie aktivieren ein Modul, drücken F2 und selektieren im Objektkatalog die Bibliothek *Excel*.

Das linke Listenfeld enthält nun alle Excel-Objekte. Selektieren Sie eines davon, werden im rechten Listenfeld alle Methoden aufgelistet, die auf das betreffende Objekt angewendet werden können, und alle Eigenschaften, die es besitzt (Abbildung 35.3).

Beispielsweise kann auf die Auflistung *Sheets*, also auf Blätter einer Arbeitsmappe, laut Abbildung die Methode *PrintPreview* angewendet werden, die für die angegebene Tabelle die Seitenansicht aktiviert.

Abbildung 35.3:
Excel-Objekte



35.3 Bezüge

*Bezugs-
eindeutigkeit*

Bezüge auf ein bestimmtes Objekt können äußerst unterschiedlich formuliert werden. Entscheidend ist einzig und allein, dass der Bezug eindeutig ist.

Nehmen Sie folgenden Ausdruck:

```
Application.ActiveWorkbook.ActiveSheet.Cells(1, 2).Value = 10
```

Application.ActiveWorkbook wendet auf das Oberobjekt *Application* die Methode *ActiveWorkbook* an, die einen Bezug auf die momentan aktive Mappe übergibt, ein *Workbook*-Objekt.

Auf dieses Objekt wird nun die Methode *ActiveSheet* angewendet, die das aktive Blatt dieser Mappe repräsentiert. Ist gerade ein Tabellenblatt aktiv, repräsentiert der Ausdruck

```
Application.ActiveWorkbook.ActiveSheet
```

somit ein Objekt vom Typ *Worksheet*, ein Tabellenblatt.

Darauf wird nun mit *Cells(1, 2)* die Methode *Cells* angewendet, die ein Objekt vom Typ *Range* übergibt, nämlich einen aus dieser einen Zelle bestehenden Bereich.

Dieses *Range*-Objekt besitzt wiederum die Eigenschaft *Value*, nämlich den Wert der Zelle, der nun durch den Ausdruck *Value = 10* verändert wird.

Den gesamten Ausdruck können Sie durch

```
ActiveSheet.Cells(1, 2).Value = 10
```

ersetzen, da bereits die Angabe *ActiveSheet* absolut eindeutig ist. Es kann nur ein Blatt geben, das aktiv ist, daher müssen Sie nicht noch zusätzlich angeben, in welcher aktiven Mappe es sich befindet.

Sie können sogar auf die Angabe *ActiveSheet* verzichten:

```
Cells(1, 2).Value = 10
```

Dann wird automatisch das gerade aktive Blatt benutzt; dennoch ziehe ich persönlich *ActiveSheet.Cells(1, 2)* vor, da daraus klar hervorgeht, dass es um das aktive Blatt geht, Sie sich beim Ausdruck *Cells(1, 2)* jedoch erst überlegen müssen, ob VBA diesen Ausdruck wohl so interpretieren wird.

Allgemein: Immer dann, wenn ein Teilbezug bereits eindeutig ist, ist es überflüssig, die Vorgänger des betreffenden Objekts anzugeben.

Die folgenden Ausdrücke sind daher äquivalent und übergeben alle einen Bezug auf Zelle B1, ein Objekt vom Typ *Bereich*:

```
AktivesBlatt.Zellen(1; 2)
AktiveArbeitsmappe.AktivesBlatt.Zellen(1; 2)
Anwendung.AktiveArbeitsmappe.AktivesBlatt.Zellen(1; 2)
```

Eine nette Falle besteht darin, dass ein Bezug sehr wohl für VBA eindeutig sein kann, aber dennoch nicht das gewünschte Objekt anspricht. Nehmen Sie folgendes Beispiel:

```
Workbooks("MAPPE1.XLS").Worksheets("Test").Cells(1, 2).Value = 10
```

Dieser Ausdruck weist jener Zelle B1 den Wert 10 zu, die sich im Tabellenblatt »Test« der Mappe MAPPE1.XLS befindet.

Enthält nur eine der momentan geöffneten Mappen eine Tabelle namens »Test«, können Sie auf die Angabe des Namens MAPPE1.XLS verzichten:

```
Worksheets("Test").Cells(1, 2).Value = 10
```

Enthalten jedoch mehrere Mappen eine Tabelle dieses Namens, könnte es passieren, dass die Anweisung daneben geht. In diesem Fall bezieht VBA die Anweisung nämlich einfach auf die momentan aktive Mappe, auch wenn Sie den Wert 10 eventuell in der Tabelle »Test« einer anderen Mappe speichern wollen.



Die With-Anweisung

Zur Vermeidung von Bandwurmausdrücken hält VBA eine spezielle Abkürzungstechnik bereit, die Anweisung *With*:

Listing 35.2: With Objekt
 Die With-Anweisung

```
With Objekt
    [Anweisung1]
    ...
    ...
    [AnweisungN]
End With
```

Sollen mehrere Anweisungen mit einem Objekt ausgeführt werden, erspart Ihnen diese Anweisung die immer wiederkehrende Angabe des gleichen Objekts. Stattdessen wird das Objekt nur ein einziges Mal angegeben, hinter dem Schlüsselwort *With*.

Die Anweisungen

Listing 35.3: ActiveSheet.Cells(1, 2).Value = 8
 Umständlich ActiveSheet.Cells(1, 2).Font.Size = 14
 ActiveSheet.Cells(1, 2).Font.Italic = True

können Sie wesentlich einfacher formulieren:

Listing 35.4: With ActiveSheet.Cells(1, 2)
 Einfacher .Value = 8
 .Font.Size = 14
 .Font.Italic = True
 End With

35.4 Objektvariablen

Eine weitere Möglichkeit zur Vermeidung von unverständlichen und langatmigen Bandwurmausdrücken ist die Verwendung von Objektvariablen. Angenommen, eine Prozedur soll den Namen des aktiven und des folgenden Blatts ausgeben, so benötigen Sie dazu normalerweise die beiden Ausdrücke

```
ActiveSheet.Name
ActiveSheet.Next.
```

Variant-Datentyp

Variablen vereinfachen derartige immer wiederkehrende Ausdrücke. Sie können nämlich Objektbezüge Variablen des Typs *Variant* zuweisen, der bekanntlich alle Arten von Daten aufnimmt, inklusive Objektbezüge. Dazu müssen Sie die Zuweisung jedoch mit *Set* einleiten:

```
Set Variable = Objektbezug
```

Nach der Zuweisung enthält die betreffende Variable den Objektbezug, sie repräsentiert gewissermaßen das Objekt, sodass Sie die Variable nun wie das Objekt selbst behandeln können, das sie vertritt:

```
Sub HoleBlattname()  
    Dim x As Variant  
  
    Set x = ActiveSheet  
    MsgBox x.Name  
    MsgBox x.Next.Name  
End Sub
```

Listing 35.5:
Objektbezüge
speichern

Nach der Zuweisung *x = ActiveSheet* enthält *x* einen Bezug auf das aktive Blatt der aktiven Mappe, repräsentiert es praktisch, und *x.Name* ist ein Bezug auf die Eigenschaft *Name* des Objekts.

Mit Objektvariablen können Sie wie gewohnt operieren. Sie können den in *x* enthaltenen Objektbezug beispielsweise mit *Set y = x* einer anderen *Variant*-Variablen zuweisen und ihn auf diese Weise kopieren, dürfen dabei jedoch ebenfalls das Schlüsselwort *Set* nicht vergessen!

Außer dem Datentyp *Variant* können Sie für Objektvariablen den Datentyp *Object* verwenden:

Object-Datentyp

```
Dim x As Object
```

Mit dem Schlüsselwort *Object* deklariert, kann *x* nur ein Objektbezug zugewiesen werden. *Object* ist ein ebenso spezieller Datentyp wie *Integer* oder *String*.

Noch spezieller sind Objektvariablen: Zu jedem Objekttyp gehört ein gleichnamiger Variablentyp, der nur Bezüge auf ein Objekt des betreffenden Typs aufnehmen kann. Deklarieren Sie mit

Objektdatentypen

```
Dim x As Sheet
```

x als Variable vom Typ *Sheet*, so kann *x* nur Bezüge auf *Sheet*-Objekte aufnehmen. Angewandt auf das Beispiel sieht das so aus:

```
Sub Obj()  
    Dim x As Sheet  
  
    Set x = ActiveSheet  
    MsgBox x.Name  
    MsgBox x.Next.Name  
End Sub
```

Listing 35.6:
Objektbezüge
speichern

Wissen Sie von vornherein, welche Objekttypen einer Variablen zugewiesen werden, sollten Sie Objektvariablen des betreffenden Typs verwenden. Nicht nur, dass die Ausführungsgeschwindigkeit höher ist als bei Verwendung allgemeinerer Variablen – diese Technik ist vor allem sicherer! Der Ausdruck

```
Set x = ActiveSheet
```

weist *x* einen Bezug auf ein Objekt vom Typ *Sheet* zu (vorausgesetzt, das momentan aktive Blatt ist eine Tabelle). Vertun Sie sich jedoch und schreiben Sie

```
Set x = ActiveWorkbook
```

so wird *x* stattdessen ein Bezug auf ein Objekt vom Typ *Workbook* zugewiesen, nämlich auf die momentan aktive Mappe, und Ihr Programm wird nicht funktionieren.

Ist das Programm sehr umfangreich, sehen Sie bestimmt nicht auf Anhieb, wo der Fehler steckt. Wurde *x* jedoch als Datentyp *Sheet* deklariert, ist es gar nicht erst möglich, *x* versehentlich einen Bezug auf einen anderen Objekttyp zuzuweisen: Bei der Ausführung der Zuweisung meldet VBA »Datentypen unverträglich« und Sie sehen dank dieser Fehlermeldung, dass der verwendete Ausdruck *ActiveWorkbook* fälschlicherweise ein *Workbook*-Objekt referenziert statt des gewünschten *Worksheet*-Objekts.

Ein sehr spezieller Zuweisungswert ist *Nothing*. *Nothing* hebt den in einer Objektvariablen gespeicherten Bezug wieder auf. Die Anweisungsfolge

```
Set x = ActiveSheet  
Set x = Nothing
```

weist *x* somit zunächst einen Bezug auf ein *Worksheet*-Objekt zu, nämlich auf das momentan aktive Blatt, und hebt diese Zuweisung anschließend wieder auf, sodass *x* danach weder auf dieses noch auf ein anderes Blatt weist.

35.5 Property-Prozeduren

Wie Sie sicher bemerkten, verbergen sich hinter Eigenschaften und Methoden letztlich ganz einfach Prozeduren, die auf ganz bestimmte Objekte angewendet werden. Derartige Prozeduren können Sie auch selbst erstellen. Und zwar drei Typen von *Property*-Prozeduren, die

- ➔ einen Bezug auf ein Objekt übergeben,
- ➔ einer Eigenschaft einen Wert zuweisen,
- ➔ den Wert einer Eigenschaft übergeben.

Die Syntax ist in allen drei Fällen sehr ähnlich:

PROPERTY SET-ANWEISUNG:

```
=====
[Private|Public] [Static] Property Set Prozedurname (Argumentliste)
  [Anweisung1]
  ...
  [Exit Property]
  ...
  [Anweisung N]
End Property
PROPERTY LET-ANWEISUNG:
```

```
=====
[Private|Public] [Static] Property Let Prozedurname (Argumentliste)
  [Anweisung 1]
  ...
  [Exit Property]
  ...
  [Anweisung N]
End Property
PROPERTY GET-ANWEISUNG:
```

```
=====
[Private|Public] [Static] Property Get Prozedurname [(Argumentliste)]
  [Anweisung 1]
  [Prozedurname = Ausdruck]
  ...
  [Exit Property]
  ...
  [Prozedurname = Ausdruck]
  [Anweisung N]
End Property
```

Abgesehen von den unterschiedlichen Schlüsselwörtern im Prozedurkopf und -rumpf und der leicht abweichenden *Exit*-Anweisung entspricht die Syntax der einer Sub-Prozedur (im Falle von *Property Set* und *Property Let*) bzw. einer Funktions-Prozedur (*Property Get*).

Sogar die Geltungsbereichsregeln sind identisch: Auf *Property*-Prozeduren können alle Prozeduren aller Module zugreifen. Mit dem Schlüsselwort *Public* kann das explizit ausgedrückt werden:

```
Public Property Set Demo()
```

Listing 35.7:
Eigenschaft-
Prozeduren

Private deklariert dagegen *Property*-Prozeduren, auf die nur die im gleichen Modul enthaltenen Prozeduren zugreifen können:

```
Private Property Set Demo()
```

Und mit dem Schlüsselwort *Static*, das Sie sowohl mit *Public* als auch mit *Private* kombinieren können, erreichen Sie, dass alle innerhalb der Prozedur deklarierten Variablen bis zum nächsten Aufruf erhalten bleiben:

```
Static Property Set Demo()
```



Argumentliste ist bei Property Set- und Property Let-Prozeduren nicht optional! Zumindest ein Argument muss übergeben werden: bei Property Set-Prozeduren der Objektbezug und bei Property Let-Prozeduren der Wert, der der Eigenschaft zugewiesen werden soll. Werden zusätzliche Argumente übergeben, muss dieser Bezug das letzte Argument sein.

36 Tabellen und Diagramme

Nach der vorangegangenen Einführung in die objektorientierte Programmierung wird es Zeit, Ihnen zu zeigen, was Sie damit anfangen können.

Prinzipiell wissen Sie natürlich: Sie können alle möglichen Excel-Objekte manipulieren. Allerdings ist das recht abstrakt. Daher werde ich Ihnen nun in diesem und den folgenden Kapiteln einige Anwendungsbeispiele zu den wichtigsten Themen geben, die gleichzeitig den Umgang mit häufig benötigten Objekten erläutern.

Allerdings muss ich Sie darauf hinweisen, dass aufgrund der Unmengen an Objekten, die Excel zur Verfügung stellt, ihrer Eigenschaften und der Vielzahl der auf sie anwendbaren Methoden jedes Kapitel zwangsläufig nur eine unvollständige Einführung in das betreffende Gebiet darstellt, die nur ein Ziel besitzt: Sie in die Lage zu versetzen, sich das betreffende Gebiet anschließend selbstständig erarbeiten zu können.



Diese Einführung erfolgt vorwiegend anhand von Anwendungsbeispielen, die Ihnen zeigen sollen, wie die verschiedenen Excel-Objekte in der Praxis eingesetzt werden. Ich erläutere beispielsweise, wie man

- ➔ herausbekommt, wie viele Arbeitsmappen gerade geöffnet sind und wie viele Tabellenblätter sie jeweils enthalten,
- ➔ ein bestimmtes Fenster öffnet oder ein Tabellenblatt aktiviert oder löscht,
- ➔ Gitternetzlinien in einem Tabellenfenster aus- bzw. wieder einblendet,
- ➔ prüft, ob Fensterausschnitte fixiert sind oder nicht und den aktuellen Zustand ändert,
- ➔ feststellt, welchen Inhalt eine bestimmte Zelle enthält oder genau umgekehrt einen Wert darin speichert
- ➔ und einen bestimmten Tabellenbereich durchsucht.

All das erfolgt anhand winziger Beispiele, kleiner Ein-, Zwei- oder auch Dreizeiler.

Danach stelle ich Ihnen zwei wesentlich komplexere Demoprogramme vor, die bereits recht anspruchsvoll sind: ein Programm, das die komfortable Eingabe von Aktienkursen in eine Tabelle mit Spaltenüberschriften wie »BASF«, »Daimler« etc. ermöglicht, indem es

- ➔ nach dem einzutragenden Datum fragt und dabei das aktuelle Datum automatisch vorgibt,
- ➔ in Spalte A die erste leere Zelle sucht und dieses Datum dort einträgt,
- ➔ danach nach dem Namen der Aktie und dem zugehörigen Kurs fragt,
- ➔ die Spaltenüberschrift sucht, deren Name dem eingegebenen Aktienamen entspricht
- ➔ und den eingegebenen Kurs in der ersten leeren Zelle der betreffenden Spalte einträgt.

Das zweite Demoprogramm stellt eine Erweiterung des ersten dar. Es geht ebenfalls von einer Aktientabelle aus, zusätzlich gibt es in diesem Fall jedoch noch ein separates Diagrammblatt, auf dem diese Kurse grafisch dargestellt werden.

Das Programm erleichtert die regelmäßige Aktualisierung der Kurse und des zugehörigen Diagramms, indem es den Anwender ebenfalls nach dem aktuellen Datum und den zugehörigen Kursen fragt und beides selbstständig an passender Stelle in die Aktientabelle einträgt. Anschließend erweitert es jedoch zusätzlich auch noch das bestehende Diagramm um die neu eingegebenen Kurse.

Entscheidend bei beiden Demoprogrammen ist, dass sie komplex genug sind, um das wirklich praxisorientierte Lernen der Excel-VBA-Programmierung zu ermöglichen, da zwangsläufig alle möglichen Arten von Excel-Objekten auf unterschiedlichste Art und Weise manipuliert werden müssen, um zum gewünschten Ergebnis zu kommen.

36.1 Referenzieren und Manipulieren

*Bezugs-
eindeutigkeit*

Das wichtigste Thema ist zweifellos der programmgesteuerte Zugriff auf Excel-Tabellen und – in Maßen – auf Diagramme. Dazu müssen Sie zunächst wissen, wie man einzelne Arbeitsmappen und Tabellen aktiviert und in diesen wiederum Bereiche und Zellen selektiert.

Einiges darüber wissen Sie bereits, beispielsweise, dass die verschiedenen Auflistungen alle Objekte eines bestimmten Typs enthalten, die momentan verfügbar sind, und dass Sie durch Übergabe des Objektnamens ein bestimmtes Objekt einer solchen Auflistung referenzieren können:

Auflistung(Objectname)

Auf alle Objekte können Sie jedoch auch indiziert zugreifen und statt des Objekt Namens seinen Index verwenden, eine Zahl:

Auflistung(Index)

Das erste Objekt jeder Auflistung besitzt übrigens den Index 1.

Zusätzlich besitzt jede Auflistung die Eigenschaft *Count*, die die Anzahl der darin enthaltenen Objekte übergibt:

x = Auflistung.Count

Damit können Sie problemlos in einer *For*-Schleife alle in einer solchen Liste enthaltenen Objekte nacheinander ansprechen:

```
For i = 1 To Auflistung.Count  
    Ausdruck = Auflistung(i)  
Next i
```

Listing 36.1:
Objekte der Reihe
nach referenzieren

Beginnen wir mit Fenstern: Die Auflistung *Windows* enthält alle momentan geöffneten Fenster, also alle Objekte vom Typ *Window*.

Fenster

Kennen Sie den Namen eines Excel-Fensters (genauer: seine Aufschrift, den Text, der sich in der Titelleiste befindet), so können Sie es in dieser Liste direkt ansprechen. Beispielsweise steht der Ausdruck

Windows("Vbaliste.xls")

stellvertretend für das Fenster »Vbaliste.xls«.

Kennen Sie zwar nicht den Namen, dafür jedoch den Index eines Fensters, so können Sie es stattdessen indiziert ansprechen. Wissen Sie, dass das betreffende Fenster als drittes geöffnet wurde und daher das dritte Fensterobjekt im Objekt *Windows* ist, referenzieren Sie es entsprechend mit

Windows(3)

Der Ausdruck

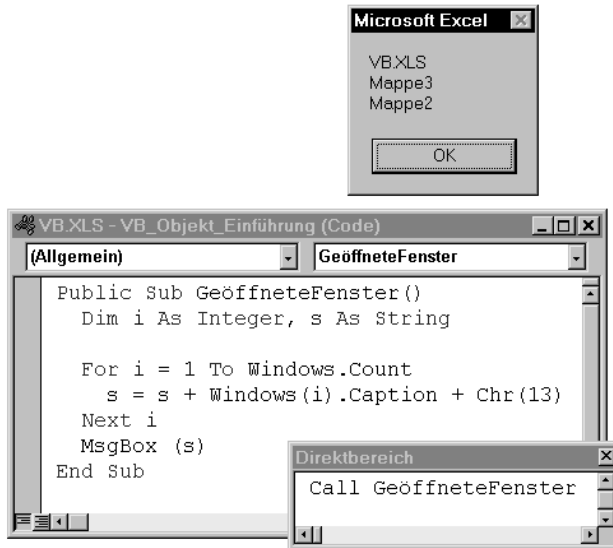
Windows.Count

übergibt die Anzahl aller momentan geöffneten Fenster.

Fenster besitzen im Gegensatz zu vielen anderen Objekten zwar nicht die Eigenschaft *Name*, die den Namen eines Objekts wiedergibt, dafür jedoch die Eigenschaft *Caption*, bei der es sich um den Text in der Titelleiste des Fensters handelt.

Das folgende Programmchen nutzt diese Eigenschaft, um der Reihe nach die Titeltexthe aller momentan geöffneten Fenster auszugeben (Abbildung 36.1).

Abbildung 36.1:
Vb.xls, Modul
»VB_Objekt_Einführung«, Prozedur
»GeöffneteFenster«



Im Beispiel sind drei Fenster vorhanden, das heißt drei Arbeitsmappen geöffnet.

Die *For*-Schleife beginnt mit dem Startwert 1. Den Endwert legt der Ausdruck *Windows.Count* fest. Da die *Count*-Methode, angewandt auf das Objekt *Windows*, die Anzahl der momentan geöffneten Fenster übergibt, besitzt die Schleife im Beispiel den Endwert 3.

Da *i* also der Reihe nach die Werte 1 bis 3 annimmt, übergibt der Ausdruck *Windows(i).Caption* somit der Reihe nach die Aufschrift des ersten, zweiten und zuletzt die des dritten Fensters.

Jeder dieser Texte wird mit *s = s + ...* an den momentan bereits in der *String*-Variablen *s* enthaltenen Text angehängt, inklusive eines abschließenden ASCII-Codes 13 (wegen *... + Zn(13)*), der bei der Ausgabe des Resultats mit *MsgBox* den Zeilenvorschub nach jeder Fensteraufschrift bewirkt.

Ein weiteres Beispiel: Fenster besitzen unter anderem die Eigenschaft *DisplayGridlines*. Sie ist *True*, wenn im betreffenden Fenster Gitternetzlinien angezeigt werden und *FALSE*, wenn keine angezeigt werden.

Auf diese Eigenschaft ist ein Schreibzugriff möglich, sodass Sie Gitternetzlinien in einem Tabellenfenster jederzeit aus- bzw. wieder einblenden können (Abbildung 36.2).

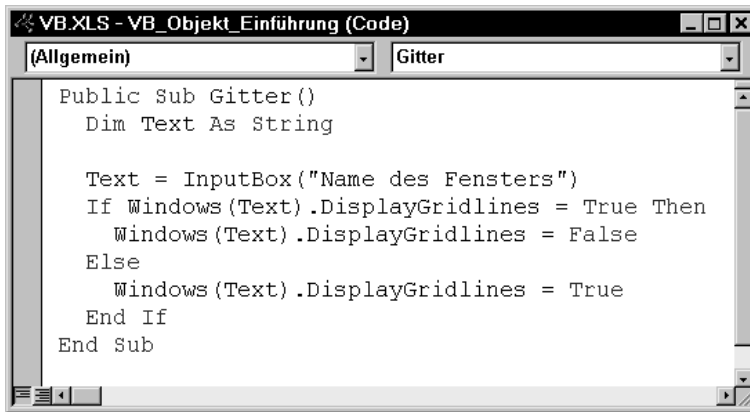


Abbildung 36.2:

Vb.xls,
»VB_Objekt_Einführung«, Prozedur
Gitter

Voraussetzung zur Ausführung dieser Prozedur ist, dass in Excel mindestens ein Fenster – beispielsweise »Test.xls« – geöffnet und darin momentan ein Tabellenblatt aktiv ist.

Rufen Sie die Prozedur *Gitter* auf, dann werden Sie nach dem Namen eines Fensters gefragt. Sie geben in diesem Beispiel »Test.xls« ein. Diese Zeichenkette wird *text* zugewiesen und danach mit

```
If Windows(text).DisplayGridlines = True Then
```

geprüft, ob für das Fenster *Windows(text)*, also für das Fenster mit der Aufschrift »Test.xls«, die Eigenschaft *DisplayGridlines* *TRUE* ist, also Gitternetzlinien zu sehen sind. Falls ja, wird die Gitternetzlinienanzeige auf *FALSE* gesetzt und damit deaktiviert, sonst genau umgekehrt aktiviert.

Zusätzlich können Sie Fenster mit der Methode *Close* zuklappen, mit der Eigenschaft *FreezePanes* prüfen, ob Fensterausschnitte fixiert sind oder nicht und den aktuellen Zustand ändern und so weiter.

Das Objekt *Workbooks* enthält eine Liste aller momentan geöffneten Arbeitsmappen und der Ausdruck

```
Workbooks("Aktien.xls")
```

referenziert entsprechend die Mappe *AKTIEN.XLS* – vorausgesetzt, sie ist verfügbar, also geöffnet.

Auch dieses Objekt besitzt die Eigenschaft *Count*, sodass es *Workbooks.Count* der momentan geöffneten Mappe übergibt.

Arbeitsmappen

Arbeitsmappen besitzen zwar nicht wie Fenster die Eigenschaft *Caption*, dafür jedoch die Eigenschaft *Name*, die den Namen der Mappe übergibt. Entsprechend übergibt

```
Workbooks(3)
```

den Namen der dritten Mappe in dieser Liste, beispielsweise »Aktien.xls«.

Auf Arbeitsmappen können Sie unter anderem die Methoden *Close*, *Activate*, *PrintPreview*, *Save* und *Open* anwenden, die die angegebene Mappe schließen, aktivieren, speichern oder öffnen. Beispielsweise aktiviert

```
Workbooks("Aktien.xls").PrintPreview
```

die Seitenvorschau für die Mappe AKTIEN.XLS, genauer für das momentan darin aktive Blatt, und

```
Workbooks("Aktien.xls").Save
```

speichert die Mappe, ist also äquivalent zum Befehl DATEI|SPEICHERN.

Blätter Die verschiedenen *Sheets*-Objekte enthalten Listen aller Blätter des betreffenden Typs, die in einer Arbeitsmappe enthalten sind. Beziehen Sie sich nicht auf eine ganz bestimmte Mappe, so wird immer die zurzeit aktive Mappe behandelt. Der Ausdruck

```
Worksheets("Tabelle1")
```

referenziert daher die Tabelle »Tabelle1« der gegenwärtig aktiven Mappe.

Wollen Sie sich jedoch auf Tabellenblätter beziehen, die sich nicht in der aktiven Mappe, sondern in einer anderen (geöffneten) Mappe befinden, müssen Sie diese folgendermaßen angeben:

```
Workbooks(Mappenname).Worksheets(Blattname)
```

Beispielsweise referenziert der Ausdruck

```
Workbooks("aktien.xls").Worksheets("Tabelle1")
```

die Tabelle »Tabelle1« der Mappe AKTIEN.XLS, auch wenn diese Mappe gerade nicht aktiv ist.

Nun ein praktisches Beispiel: Unter anderem kann auf nahezu alle Objekte die Methode *Delete* angewendet werden, die das betreffende Objekt löscht:


```
Sub Test
  For i = 1 To Worksheets.Count
    Worksheets(i).Delete
  Next i
End Sub
```

Listing 36.2:
Indizierter Zugriff
auf Listen

Die Eigenschaft *Count* übergibt die Anzahl der in der aktiven Mappe enthaltenen Tabellenblätter. Der Schleifenindex *i* nimmt der Reihe nach die Werte 1, 2, 3 etc. an, sodass der Ausdruck *Worksheets(i)* im ersten Schleifendurchgang einen Bezug auf das erste Tabellenblatt der Mappe darstellt, im zweiten einen Bezug auf das zweite Tabellenblatt und so weiter.

Die Methode *Delete*, die das angegebene Objekt löscht, wird somit der Reihe nach auf alle Blätter der aktiven Mappe angewendet, die dadurch nacheinander gelöscht werden.

Auf Blätter können Sie unter anderem die Methoden *Activate*, *Close*, *Save* und *Delete* anwenden. Der Name der betreffenden Methode ist sicher jeweils Erläuterung genug.

Blätter besitzen unter anderem die Eigenschaft *Name*, die den in der Registerleiste angezeigten Namen übergibt. Da auf diese Eigenschaft ein Schreibzugriff möglich ist, würde beispielsweise der Ausdruck

```
Workbooks("aktien.xls").Sheets(1).Name = "Tabelle1"
```

dem ersten Blatt der Mappe AKTIEN.XLS den Namen »Tabelle1« geben. Dabei ist übrigens vollkommen egal, ob es sich um ein Tabellen- oder um ein Diagrammblatt handelt. Im Gegensatz dazu würde

```
Workbooks("aktien.xls").Worksheets(1).Name = "Tabelle1"
```

das erste Tabellenblatt der Mappe so nennen und

```
Workbooks("aktien.xls").Charts(1).Name = "Tabelle1"
```

das erste Diagrammblatt.

Die verschiedensten Objekte (Arbeitsmappen, Diagramme, Dialogblätter, Menüleisten, Module, OLE-Objekte, Ausschnitte, Bereiche, Fenster, Arbeitsmappen und Tabellen) können Sie mit der Methode *Activate* zum aktiven Objekt machen, analog zu einem Mausklick auf ein gerade inaktives Fenster.

Objekte aktivieren

Statt sich auf eine nicht aktive Mappe zu beziehen, können Sie sie alternativ dazu mit der *Activate*-Methode aktivieren und sich danach einfach auf die aktive Mappe beziehen.

Zum Beispiel aktiviert die folgende Prozedur zunächst die Mappe AKTIEN.XLS und danach die darin enthaltene Tabelle »Tabelle1«:

Listing 36.3: Sub Ein_Test ()
Objekte aktivieren Worksheets("Aktien.xls").Activate
 Worksheets("Tabelle1").Activate
End Sub

Bezüge auf aktive Objekte Häufig wollen Sie mit der aktiven Mappe oder dem darin gegenwärtig aktiven Tabellen- oder Diagrammblatt arbeiten, kennen jedoch den Namen des Blattes nicht.

Dann verwenden Sie die Eigenschaft *Active...*, die einen Bezug auf das aktive Objekt des betreffenden Typs übergibt:

- ➔ Die Eigenschaft *ActiveSheet* übergibt einen Bezug auf das aktive Blatt, das beispielsweise eine Tabellen- oder ein Diagrammblatt sein kann.
- ➔ Die Eigenschaft *ActiveWorksheet* übergibt dagegen einen Bezug auf das aktive Tabellenblatt.
- ➔ Die Eigenschaft *ActiveChart* übergibt dagegen einen Bezug auf das aktive Diagramm.
- ➔ Die Eigenschaft *ActiveWindow* übergibt einen Bezug auf das aktive Fenster.
- ➔ Die Eigenschaft *ActiveWorkbook* übergibt einen Bezug auf die aktive Mappe.
- ➔ Die Eigenschaft *ActiveCell* übergibt einen Bezug auf die aktive Zelle, ein Objekt vom Typ *Range*.

Ist zurzeit die Tabelle »Tabelle1« aktiv, sind daher die beiden folgenden Bezüge auf diese Tabelle äquivalent:

```
Sheets("Tabelle1")  
ActiveSheet
```

Befindet sich diese Tabelle nicht in der aktiven Mappe, sondern in der – geöffneten – Mappe AKTIEN.XLS, so sind folgende Bezüge darauf äquivalent:

```
Workbooks("aktien.xls").Sheets("Tabelle1")  
Workbooks("aktien.xls").ActiveSheet
```

Vorauszusetzen ist dabei, dass »Tabelle1« die aktive Tabelle jener Mappe ist, also jenes Blatt, das zu sehen ist, wenn Sie die Mappe durch Anklicken aktivieren.

Ist gerade kein Objekt des betreffenden Typs aktiviert, geben die *Active...*-Eigenschaften den Bezug *Nothing* zurück, beispielsweise wenn Sie auf die aktive Mappe die Eigenschaft *ActiveChart* anwenden, darin jedoch kein Diagrammblatt aktiv ist.

Auf einzelne Zellen beziehen Sie sich mit der Methode *Cells*. Sie kann auf die Objekttypen *Application*, *Range* und *Worksheet* angewendet werden und übergibt eine einzelne Zelle als Objekt des Typs *Range*. Der Ausdruck

Zellen

```
Cells(Zeilenindex, Spaltenindex)
```

übergibt einen Bezug auf die Zelle, die sich in der Zeile *Zeilenindex* und der Spalte *Spaltenindex* befindet. Beispielsweise übergibt

```
Cells(3, 1)
```

einen Bezug auf die Zelle A3 der aktiven Tabelle.

Ist jedoch gerade keine Tabelle aktiv, sondern beispielsweise das Modulblatt, in dem sich die Prozedur befindet, die diesen Ausdruck enthält, so müssen Sie zusätzlich das gewünschte Tabellenobjekt angeben. Der Ausdruck

```
Sheets("Tabelle1").Cells(3, 1)
```

bezieht sich entsprechend auf Zelle A3 der Tabelle »Tabelle1« der aktiven Mappe.

Der Ausdruck

```
Workbooks("aktien.xls").Sheets("Tabelle1").Cells(3, 1)
```

bezieht sich auf Zelle A3 der Tabelle »Tabelle1«, die sich in der Mappe AKTIEN.XLS befindet.

Statt die Zeilen- und Spaltenindizes anzugeben, können Sie auch die alternative Syntax

```
Cells(Index)
```

verwenden: *Index* ist eine Zahl, die bei Zeile 1 beginnend der Reihe nach alle Spalten (maximal 256) der betreffenden Zeile abläuft. Für A1 ist dieser Index 1, für B1 eine 2, für die letzte Zelle IV1 von Zeile 1 ist der Index 256, für B1 257, für C2 258 etc.

Die Methode *Range* übergibt wie die *Cells*-Methode ein Objekt vom Typ *Range*, bei dem es sich allerdings um mehr als eine einzige Zelle handeln kann. Der Ausdruck

Bereiche

`Range(Bereichsangabe)`

übergibt einen Bezug auf den angegebenen Bereich. Der Ausdruck

`Range("A1:C3")`

übergibt entsprechend einen Bezug auf den Bereich A1:C3. Entsprechend übergibt

`Range("A:A")`

einen Bezug auf Spalte A und

`Range("1:3")`

einen Bezug auf die Zeilen 1 bis 3.

Alternativ dazu können Sie auch die *Cells*-Methode zur Bereichsangabe verwenden, was oft flexibler ist. Der Ausdruck

`Range(Cells(1, 1), Cells(3, 3))`

übergibt einen Bezug auf den Bereich A1 (*Cells(1, 1)*) bis C3 (*Cells(3, 3)*).

**Bereiche
auswählen**

Ein Bezug auf einen Bereich bedeutet noch nicht, dass dieser automatisch selektiert wird. Dazu verwenden Sie die Methode *Select*. Beispielsweise selektiert der Ausdruck

`Range("A1:C3").Select`

in der aktiven Tabelle den Bereich A1:C3, so als hätten Sie die Maus bei gedrückter Maustaste darüber gezogen. Allerdings muss auch tatsächlich eine Tabelle aktiv sein, wofür Sie zuvor mit der *Activate*-Methode sorgen können:

Listing 36.4:
Bereichs-
manipulation

```
Sub Test2()  
    Worksheets(1).Activate  
    Range("a1:c3").Select  
End Sub
```

Ebenso gut können Sie den Bereich mit der Methode *Delete* entfernen oder mit *Clear* die Inhalte seiner Zellen löschen.

Um einen Zellbereich auszufüllen, verwenden Sie die Eigenschaft *Value*. Sie übergibt bzw. setzt den Wert eines Objekts, angewandt auf den Inhalt der zugehörigen Zellen. Beispielsweise füllt

`Range("A1:C3").Value = "Hallo"`

den Bereich A1:C3 mit der Zeichenkette »Hallo« aus, speichert diese Zeichenkette also in jeder Zelle des Bereichs. Entsprechend speichert

```
Range("A1").Value = "Hallo"
```

nur in A1 diese Zeichenkette, ebenso wie die Anweisung

```
Cells(1, 1).Value = "Hallo"
```

Die Eigenschaft Value ist optional. Das heißt, um in einer Zelle oder einem Bereich Werte zu speichern, können Sie ebensogut schreiben:

```
Range("A1") = "Hallo"
```

bzw.

```
Cells(1, 1) = "Hallo"
```

Statt Werte können Sie auch Formeln speichern. Dazu verwenden Sie die Eigenschaft *Formel*. Die betreffende Formel muss als Zeichenkette in Anführungszeichen eingeschlossen werden. Beispielsweise speichert

```
Range("A1").Formula = "=SUM(B2:B5)"
```

ebenso wie

```
Cells(1, 1).Formula = "=SUM(B2:B5)"
```

in A1 die Formel `=SUMME(B2:B5)`.

Wie immer beim Programmieren mit VBA müssen Sie die englische Bezeichnung verwenden, also SUM statt SUMME! Die Umwandlung in die deutsche Formelbezeichnung SUMME erfolgt automatisch beim Speichern der Formel in der Zelle.

Um Zellen zu formatieren, verwenden Sie unter anderem die Bereicheigenschaft *Font*. Sie übergibt ein gleichnamiges Objekt, das auch die Eigenschaften *Name* (Name der Schriftart) und *Size* (Schriftgröße) besitzt, auf die sowohl der Lese- als auch der Schreibzugriff möglich ist:

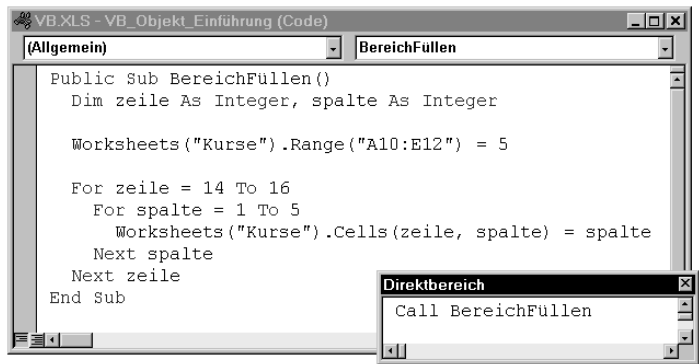
```
Sub Test3()  
    Worksheets(1).Activate  
    Range("A1:C3").Font.Name = "Times New Roman"  
    Range("A1:C3").Font.Size = "12"  
End Sub
```

Listing 36.5:
Schriftmanipulation



Noch ein Beispiel für die Anwendung von *Range* und von *Cells* (Abbildung 36.3):

Abbildung 36.3:
Vb.xls,
»VB_Objekt_Einführung
ung«, Prozedur
»BereichFüllen«



Dieses Programm füllt zunächst alle Zellen des Bereichs A10:E12 der Tabelle »Kurse« mit dem Wert 5. Dafür ist eine einzige Anweisung zuständig:

`Worksheets("Kurse").Range("A10:E12") = 5`

Sie referenziert zunächst das Blatt »Kurse« der aktiven Mappe und darin wiederum den Bereich A10:E12, dem sie anschließend den Wert 5 zuweist (Abbildung 36.4).

Abbildung 36.4:
Auswirkungen

	A	B	C	D	E
9					
10	5,00	5	5	5	5
11	5	5	5	5	5
12	5	5	5	5	5
13					
14	1	2	3	4	5
15	1	2	3	4	5
16	1	2	3	4	5
17					

Weitaus interessanter sind die beiden folgenden ineinander verschachtelten Schleifen. Sie zeigen, wie man alle Zellen eines bestimmten Bereichs der Reihe nach ansprechen kann.

Die Schleifenvariable *zeile* besitzt zunächst den Wert 14 und *spalte* den Wert 1. Entsprechend referenziert *ZelleListe(zeile; spalte)* die Zelle in der ersten Spalte von Zeile 14, also A14. Die Anweisung

`Worksheets("Kurse").Cells(zeile, spalte) = spalte`

weist daher der Zelle A14 der Tabelle »Kurse« den aktuellen Wert der Variablen *spalte* zu, also 1.

Im nächsten Durchgang der inneren Schleife besitzt *spalte* den Wert 2, sodass der Ausdruck *Cells(zeile, spalte)* die Zelle B14 referenziert und dieser der Wert 2 zugewiesen wird.

Das geht so weiter, bis im letzten Durchgang der inneren Schleife der Zelle E14 der Wert 5 zugewiesen wird.

Nun beginnt der zweite Durchgang der äußeren Schleife. *zeile* besitzt den Wert 2 und *spalte* wird wieder der Startwert 1 zugewiesen. *Cells(zeile, spalte)* referenziert somit die Zelle A15 und dieser wird der Wert 1 zugewiesen.

Anschließend werden die Zellen B15, C15, D15 und E15 durchlaufen und im letzten Durchgang der äußeren Schleife analog dazu die Zellen A16, B16, C16, D16 und E16.

Sehr häufig taucht in Excel-VBA-Programmen folgendes Problem auf: In einer Tabelle muss ein Wert eingetragen werden, den der Benutzer eingab. Alle Einträge erfolgen in einer Spalte untereinander und jeder neue Wert soll in der ersten freien Zelle der betreffenden Spalte eingetragen werden.

*Suche nach
Leerzellen*

Das Problem lautet also: »Suche die erste Zelle einer Spalte, die nichts enthält, also leer ist.«

Dazu können Sie eine Prozedur schreiben, die wie zuvor gezeigt der Reihe nach alle Zellen eines Bereichs abläuft und jeweils prüft, ob die aktuelle Zelle einen Inhalt besitzt. Wesentlich einfacher ist jedoch folgende Methode (Abbildung 36.5).

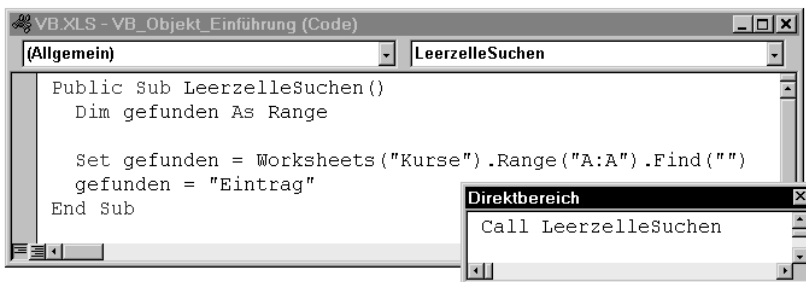


Abbildung 36.5:
Vb.xls,
»VB_Objekt_Einführung«, »Prozedur
LeierzelleSuchen«

Der Ausdruck *Worksheet(»Kurse«).Range(»A:A«)* referenziert Spalte A der Tabelle »Kurse«. Auf diesen Bereich wird die Methode *Find* losgelassen, um nach »"« zu suchen, also nach »Nichts«. *Find* übergibt als Resultat einen Bezug auf die erste leere Zelle von Spalte A, die von oben nach unten durchsucht wird.

Dieser Zellbezug ist eine Bereichsangabe und wird entsprechend einer Variablen vom Typ *Bereich* zugewiesen, *gefunden*. Anschließend weist

gefunden = "Eintrag"

der betreffenden Zelle die Zeichenkette »Eintrag« zu, im Beispiel der Zelle A7 (Abbildung 36.6).

Abbildung 36.6:
Nach der
Programm-
ausführung

	A	B	C	D	E
1	Datum	BASF	DAIMLER	DEUTSCHE	HÖCHST
2	01.01.09	204,00	503,00	783,00	273,00
3	02.01.09	205,50	505,50	782,00	275,00
4	03.01.09	207,00	508,00	779,00	285,00
5	04.01.09	206,00	512,00	785,00	296,00
6	05.01.09	206,80	509,00	788,50	284,00
7	Eintrag				
8					

36.2 Tabellenfunktionen in VBA

Sie können in VBA-Programmen selbstverständlich Tabellenfunktionen aufrufen. Allerdings sind dabei einige Besonderheiten zu beachten.

Nehmen wir an, die Tabelle »PKWs« enthält einen benannten Bereich A2:B8 mit dem Namen »Suchbereich« (Abbildung 36.7).

Abbildung 36.7:
Vb.xls, Tabelle
»PKWs«

Suchbereich fx Audi

	A	B
1	Pkw	Durchschnittspreis
2	Audi	15.000,00 DM
3	BMW	70.000,00 DM
4	Mercedes	50.000,00 DM
5	Opel	23.000,00 DM
6	Porsche	123.000,00 DM
7	Toyota	65.000,00 DM
8	VW	12.000,00 DM
9		

Die erste Spalte enthält alphabetisch sortierte Bezeichnungen von Automarken und die zweite Spalte die Durchschnittspreise der jeweiligen Hersteller.

Angenommen, Sie wollen Spalte A nach einem Hersteller durchsuchen und den zugehörigen Preis in Spalte B herausbekommen. Dazu verwenden Sie am einfachsten die Tabellenfunktion *VLookup* (die englische Bezeichnung für *Sverweis*) Wahrscheinlich würden Sie *VLookup* etwa so aufrufen:

```
Autopreis = VLookup(Hersteller, Suchbereich, 2)
```

Diese Programmzeile würde gleich zwei Fehler enthalten: Zunächst einmal können Sie in VBA-Programmen Excel-Tabellenfunktionen nicht in der gewohnten Art

```
x = Funktion()
```

anwenden.

Stattdessen muss die Tabellenfunktion immer auf das Objekt Application, also auf Excel, angewendet werden, als eine Methode dieses Objekts!

Statt

```
Autopreis = VLookup(Hersteller, Suchbereich, 2)
```

müssen Sie daher schreiben:

```
Autopreis = Application.VLookup(Hersteller, Suchbereich, 2)
```

Der zweite Fehler: Sie können im Aufruf der Tabellenfunktion aus einem VBA-Programm heraus als Suchmatrix leider nicht den Namen »Suchbereich« des Bereichs angeben, ebenso wenig wie die entsprechenden Zellreferenzen, beispielsweise »A2:B8«. Stattdessen müssen Sie eine Objektvariable vom Typ Range übergeben und den Suchbereich zuvor einer solchen Variablen zuweisen.

Korrekt sieht das Programm wie folgt aus (Abbildung 36.8):

Zunächst werden Sie nach dem Hersteller gefragt. Geben Sie »Opel« ein, wird diese Zeichenkette der Variablen *Hersteller* zugewiesen.

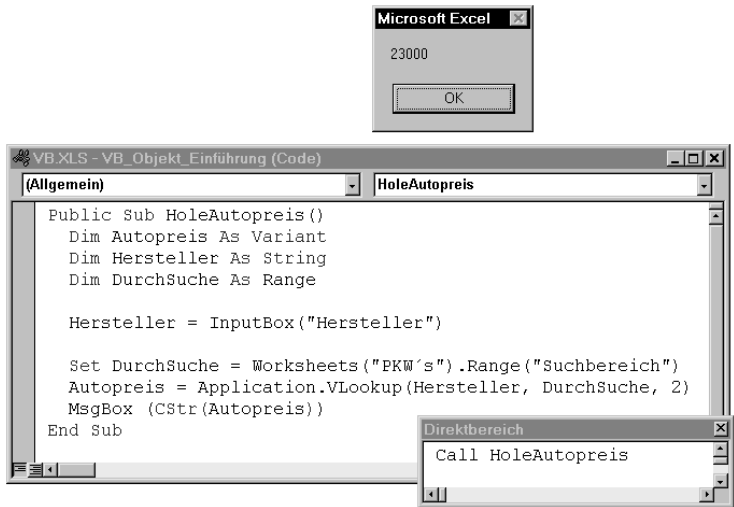
Danach wird der Range-Variablen *DurchSuche* ein Bezug auf den benannten Bereich »Suchbereich« der Tabelle »PKWs« zugewiesen. Die folgende Zeile durchsucht diesen Bereich mit *VLookup* nach dem Herstellernamen und übergibt den zugehörigen Wert in die Spalte der gleichen Zeile, im Beispiel 23000.

Dieser Wert wird *Autopreis* zugewiesen und – mit *CStr* in eine Zeichenkette umgewandelt – ausgegeben.

!!
STOP

!!
STOP

Abbildung 36.8:
Vb.xls, Modul
»VB_Objekt_Einführung«, Prozedur
»HoleAutopreis«



36.3 Demo »Aktienkursaktualisierung«



Alle für das folgende Demo benötigten Blätter befinden sich in der Mappe VB.XLS auf der Begleit-CD.

Ich gehe von einer Tabelle aus, in die Aktienkurse eingegeben werden und die fortlaufend aktualisiert wird (Abbildung 36.9).

Abbildung 36.9:
Vb.xls, Tabelle
»Kurse«

The screenshot shows a table titled 'Vb.xls' with the following data:

	A	B	C	D	E
1	Datum	BASF	DAIMLER	DEUTSCHE	HÖCHST
2	01.01.09	204,00	503,00	783,00	273,00
3	02.01.09	205,50	505,50	782,00	275,00
4	03.01.09	207,00	508,00	779,00	285,00
5	04.01.09	206,00	512,00	785,00	296,00
6	05.01.09	206,80	509,00	788,50	284,00
7					

Selbstverständlich könnten Sie beim Eintragen eines neuen Kurses diesen und das zugehörige Datum manuell eingeben. Komfortabler und vor allem sicherer ist das Ganze jedoch programmgesteuert. Der geplante Ablauf:

1. Das Programm fragt Sie nach dem einzutragenden Datum und gibt dabei das aktuelle Datum (laut Rechneruhr) vor.
2. Es sucht in Spalte A die erste leere Zelle und trägt dieses Datum dort ein.

3. Danach fragt es nach dem Namen der Aktie und dem zugehörigen Kurs.
4. In Zeile 1 wird eine Zeilenüberschrift gesucht, deren Name dem eingegebenen Aktiennamen entspricht.
5. Danach wird die Spalte, in der der Name gefunden wurde, nach der ersten leeren Zelle durchsucht und dort der eingegebene Kurs eingetragen.

Das zugehörige Programm, die Prozedur »Kurseingabe« im Modul »Neuer-Kurs« von VB.XLS, erläutere ich in Abschnitten:

```
Sub Kurseingabe()  
    Dim heute As Date, b As Range  
    Dim aktie As String, kurs As Single
```

Listing 36.6:
Variablen-
deklaration

Zu Beginn werden die benötigten Variablen deklariert: *b* wird verschiedene Bezüge auf *Range*-Objekte enthalten, *heute* das eingegebene Datum, *aktie* den Namen der Aktie und *kurs* den zugehörigen aktuellen Kurs. Der zweite Abschnitt sieht so aus:

```
'Datum erfragen und eintragen  
heute = CDate(InputBox("Datum?", , CStr(Date)))  
Workbooks("vb.xls").Activate  
Worksheets("Kurse").Activate  
Set b = Range("A:A")  
Set b = b.Find("")  
b.Value = heute
```

Listing 36.7:
Datum behandeln

Die Anweisung

```
heute = CDate(InputBox("Datum?", , CStr(Date)))
```

wandelt mit der *CStr*-Funktion das von der Funktion *Date* übergebene aktuelle Datum in eine Zeichenkette um. Diese Zeichenkette wird der Funktion *InputBox* als drittes Argument übergeben, was bewirkt, dass diese Zeichenkette und somit das aktuelle Datum im resultierenden Dialogfeld vorgegeben wird (Abbildung 36.10).

Die nach dem Schließen des Dialogfelds von *InputBox* übergebene Zeichenkette wird danach genau umgekehrt mit der *CDate*-Funktion in ein Datum umgewandelt und der Datumsvariablen *heute* zugewiesen.

Abbildung 36.10:
Datumseingabe



Um die folgenden Programmteile, die auf die Tabelle »Kurse« der Mappe VB.XLS zugreifen, möglichst einfach zu halten, Sorge ich nun dafür, dass diese Tabelle die aktive Tabelle ist. Für den Fall, dass außer VB.XLS zurzeit noch andere Mappen geöffnet sind, wird VB.XLS zunächst mit

```
Workbooks("vb.xls").Activate
```

zur aktiven Mappe gemacht. Danach aktiviert die Anweisung

```
Worksheets("Kurse").Activate
```

die Tabelle »Kurse« dieser Mappe.



Im Beispiel befindet sich die Prozedur in der gleichen Mappe wie die Aktientabelle. Wäre das nicht der Fall, griffen Sie mit der Prozedur also auf eine Tabelle zu, die sich in einer anderen Mappe befände, müssten Sie den Fall berücksichtigen, dass die betreffende Mappe überhaupt nicht geöffnet wäre, dieses also zunächst prüfen und die Mappe mit der Aktientabelle gegebenenfalls erst noch öffnen.

In der Zuweisung

```
Setze b = Range("A:A")
```

Suchen-Methode

übergibt der Ausdruck `Range(»A:A«)` einen Bezug auf die komplette Spalte A als `Range`-Objekt, der anschließend der `Range`-Variablen `b` zugewiesen wird. Dieser Bereichsbezug wird im Ausdruck

```
Set b = b.Find("")
```

der `Find`-Methode übergeben, die den angegebenen Bereich durchsucht, im Beispiel also die Spalte A der aktiven Tabelle »Kurse«.

Das erste zu übergebende Argument ist ein Suchkriterium wie »Maier« oder »Müller«. Wird wie hier eine leere Zeichenkette `""` übergeben, sucht `Find` die erste Zelle, die nichts enthält und findet im Beispiel die Zelle A7.

Diese Zelle wird nun nicht etwa selektiert oder aktiviert, dafür übergibt *Find* jedoch selbst einen Bereichsbezug und zwar einen Bezug auf die gefundene Zelle. Er wird *b* zugewiesen, sodass *b* nun auf Zelle A7 verweist.

Die folgende Anweisung wendet auf *b*, also auf das *Range*-Objekt, das die Zelle A7 repräsentiert, die Methode *Value* an, um der Zelle das in *heute* gespeicherte Datum als neuen Wert zuzuweisen:

```
b.Value = heute
```

A7 enthält nun wie gewünscht das eingegebene Datum (Abbildung 36.11).

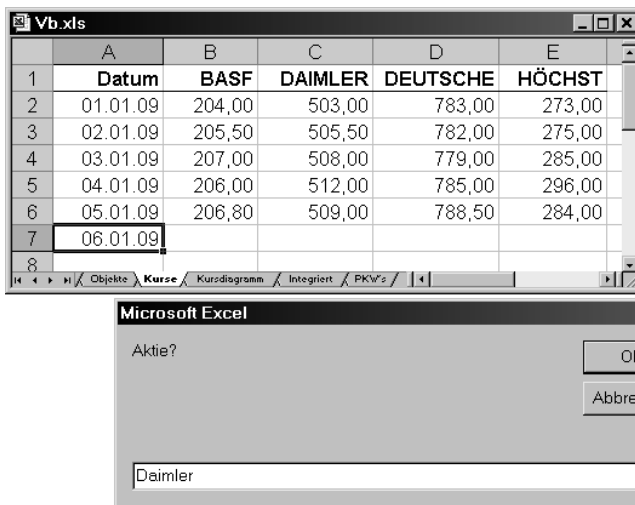


Abbildung 36.11:
Datumseintragung

Wie die Abbildung zeigt, wird der Anwender unmittelbar nach dem Eintragen des Datums in A7 nach dem Namen der Aktie gefragt, deren Kurs er eingeben will. Dafür ist die erste Anweisung des folgenden Prozedurabschnitts verantwortlich:

```
'Kurs erfragen und eintragen
aktie = InputBox("Akte?")
kurs = CSng(InputBox("Kurs?"))
Set b = Range("1:1")
Set b = b.Find(aktie)
Set b = b.EntireColumn
Set b = b.Find("")
b.Value = kurs
End Sub
```

Listing 36.8:
Aktienname
und Aktienkurs
behandeln

Die Anweisung

```
aktie = InputBox("Akte?")
```

fragt nach einem Aktiennamen, der Anwender gibt beispielsweise »Daimler« ein und diese Zeichenkette wird der Variablen *aktie* zugewiesen.

Die folgende Anweisung

```
kurs = CSng(InputBox("Kurs?"))
```

öffnet erneut ein Dialogfeld. In diesem wird nach dem Aktienkurs gefragt (Abbildung 36.12).

Abbildung 36.12:
Kurseingabe



Die von *InputBox* übergebene Zeichenkette, beispielsweise »523,3«, wird mit der *CSng*-Funktion in einen *Single*-Wert gewandelt und der Variablen *kurs* zugewiesen, die entsprechend vom Typ *Single* sein muss.

Die folgende Anweisung

```
Set b = Range("1:1")
```

weist *b* einen Bezug auf die komplette Zeile 1 zu. Anschließend wird auf diesen Bezug die *Find*-Methode losgelassen, die Zeile 1 nach der in *aktie* enthaltenen Zeichenkette durchsuchen soll, beispielsweise nach »Daimler«:

```
Set b = b.Find(aktie)
```

Ohne weitere Argumente unterscheidet *Find* nicht zwischen Groß-/Kleinschreibung und findet daher in Zelle C1 den Text »DAIMLER«. Der übergebene Bereichsbezug auf diese Zelle wird *b* zugewiesen. Nun muss die Spalte, in der sich dieser Aktienname befindet, nach der ersten leeren Zelle durchsucht werden. Die Anweisung

```
Set b = b.EntireColumn
```

**EntireColumn-
Methode**

benutzt die *EntireColumn*-Methode, die auf Bereichsobjekte angewandt wird und selbst wieder einen Bereichsbezug übergibt und zwar einen Bezug auf die komplette Spalte, auf die der Bezug verweist.

Im Beispiel ist das ein Bezug auf die Spalte C, da ein Bereichsbezug auf C1 übergeben wurde und sich diese Zelle nun mal in Spalte C befindet.

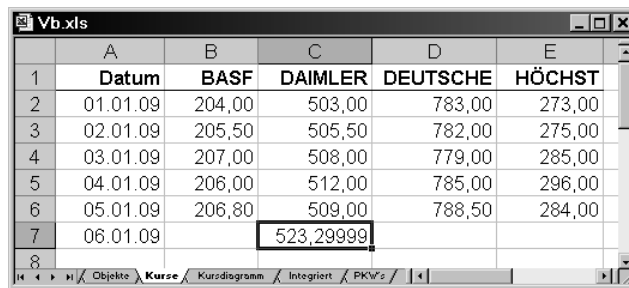
Erneut wird *Find* angewendet. Der Bereichsbezug *b*, der nun auf Spalte C verweist, wird nach der ersten leeren Zelle durchsucht:

```
Set b = b.Find("")
```

Im Beispiel findet *Find* die Zelle C7 und übergibt einen Bereichsbezug auf diese Zelle, der wieder *b* zugewiesen wird. Die folgende Anweisung

```
b.Value = kurs
```

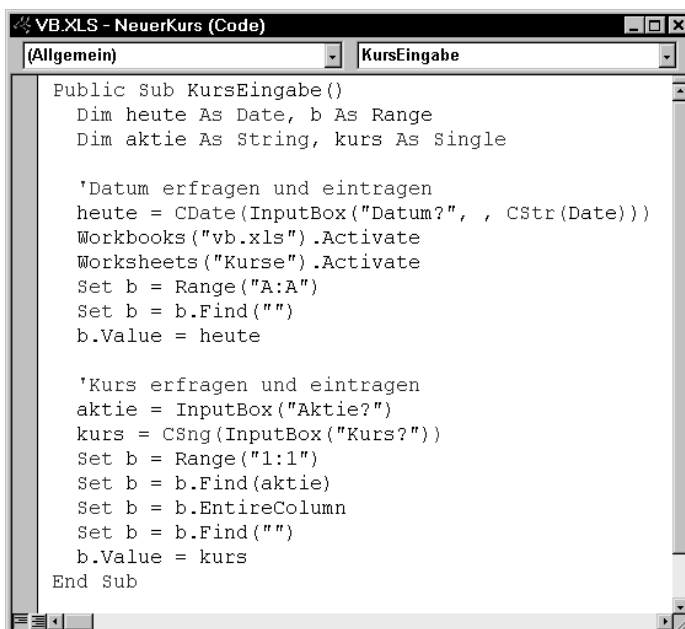
speichert den Inhalt von *kurs* in dieser Zelle, also den von Ihnen eingegebenen Aktienkurs (Abbildung 36.13).



	A	B	C	D	E
1	Datum	BASF	DAIMLER	DEUTSCHE	HÖCHST
2	01.01.09	204,00	503,00	783,00	273,00
3	02.01.09	205,50	505,50	782,00	275,00
4	03.01.09	207,00	508,00	779,00	285,00
5	04.01.09	206,00	512,00	785,00	296,00
6	05.01.09	206,80	509,00	788,50	284,00
7	06.01.09		523,29999		
8					

Abbildung 36.13:
Kurs Speicherung

Hier sehen Sie das Gesamtprogramm im Überblick (Abbildung 36.14):



```

VB.XLS - NeuerKurs (Code)
[Allgemein] | KursEingabe

Public Sub KursEingabe()
    Dim heute As Date, b As Range
    Dim aktie As String, kurs As Single

    'Datum erfragen und eintragen
    heute = CDate(InputBox("Datum?", , CStr(Date)))
    Workbooks("vb.xls").Activate
    Worksheets("Kurse").Activate
    Set b = Range("A:A")
    Set b = b.Find("")
    b.Value = heute

    'Kurs erfragen und eintragen
    aktie = InputBox("Aktie?")
    kurs = CSng(InputBox("Kurs?"))
    Set b = Range("1:1")
    Set b = b.Find(aktie)
    Set b = b.EntireColumn
    Set b = b.Find("")
    b.Value = kurs
End Sub

```

Abbildung 36.14:
Vb.xls, Modul
»NeuerKurs«,
Prozedur »Kursein-
gabe«

Verschachtelungen

Sind Sie ein Freund knapper Formulierungen, so können Sie das Programm wesentlich kürzer fassen. Das Prinzip wird klar, wenn Sie sich zwei Anweisungen der ersten Programmversion anschauen:

```
Set b = Range("A:A")
Set b = b.Find("")
```

In der ersten Anweisung übergibt die *Range*-Methode einen Bereichsbezug, der *b* zugewiesen und auf den danach die *Find*-Methode angewendet wird.

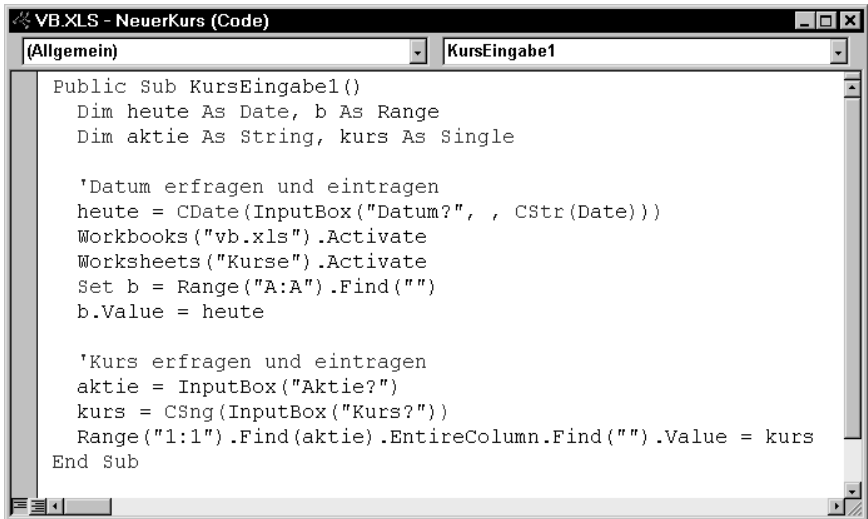
Ebenso gut wäre es möglich, auf diese Zwischenspeicherung zu verzichten und die *Find*-Methode gleich auf den von *Range* übergebenen Bereichsbezug anzuwenden:

```
Set b = Range("A:A").Find("")
```

Damit ergibt sich eine erheblich kürzere Programmvariante (Abbildung 36.15):

Abbildung 36.15:

Vb.xls, Modul
»NeuerKurs«,
Prozedur
»Kurseingabe1«



Denn das zuvor erläuterte Prinzip wird auch in dem folgenden komplexen Ausdruck angewandt:

```
Range("1:1").Find(aktie).EntireColumn.Find("").Value = kurs
```

Range übergibt einen Bezug auf einen Bereich, genauer auf Zeile 1. Auf diesen Bezug wird die *Find*-Methode angewandt, um in dem Bereich den Aktiennamen zu finden. Die *Find*-Methode übergibt selbst wieder einen Bereichsbezug und zwar auf die gefundenen Zelle.

Auf diesen Bezug wird nun die Methode *EntireColumn* angewandt, die die gesamte zugehörige Spalte als Bereichsbezug übergibt, und auf diesen wird erneut *Find* angewandt, diesmal, um die erste leere Zelle der Spalte zu finden.

Find übergibt den entsprechenden Bereichsbezug und mit der Eigenschaft *Value* wird der Zelle der in *kurs* enthaltene Wert zugewiesen.

Das Ganze ist nichts anderes als die Verschachtelung von Funktionsaufrufen in der Art

```
x = Wurzel(Wert(TeilZnF("Hallo", 3)))
```

mit dem Unterschied, dass Funktionsaufrufe von innen nach außen gelesen werden, Objektausdrücke jedoch von links nach rechts.

Sie müssen natürlich darauf achten, welchen Objekttyp eine Eigenschaft/Methode übergibt, um zu wissen, welche Eigenschaft/Methode darauf wiederum angewandt werden kann – allerdings ist das bei Funktionen schließlich auch nicht anders.

Am Demoprogramm ist übrigens manches verbesserungswürdig. Beispielsweise gibt man üblicherweise zwar nur ein Datum, dann aber gleich mehrere Kurse verschiedener Aktien ein.

Um das zu erreichen, muss der letzte Programmabschnitt in einer Schleife immer wieder ausgeführt werden, bis der Anwender keinen Aktiennamen mehr eingibt oder aber sich bei der Kurseingabe überlegt, dass er keine Lust mehr hat und überhaupt keinen Kurs eingibt (Abbildung 36.16).

Der Ausdruck

Endlosschleifen

```
Do ... Loop While True
```

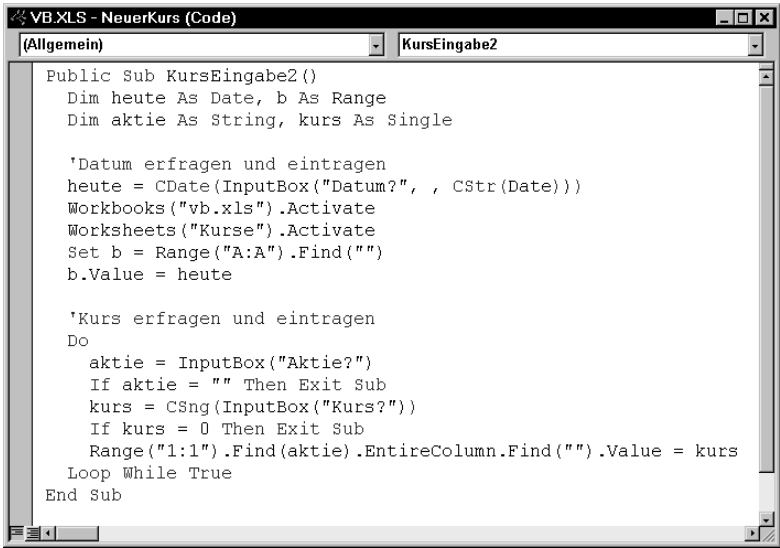
ist eine einfache Technik zur Bildung von Endlosschleifen. Diese Schleife läuft ewig, also müssen wir in der Schleife dafür sorgen, dass sie gegebenenfalls abgebrochen wird. Der dazu verwendete Ausdruck

```
If aktie = "" Then Exit Sub
```

prüft, ob der Anwender bei der Eingabe eines Aktiennamens gar nichts eingab, *InputBox* also eine leere Zeichenkette zurückgab. Falls ja, wird die Prozedur verlassen. Die Anweisung

```
If kurs = 0 Then Exit Sub
```

Abbildung 36.16:
Vb.xls, Modul
»NeuerKurs«,
Prozedur
»Kurseingabe2«



bewirkt das Gleiche, wenn er als Aktienkurs eine leere Zeichenkette eingab und `ZCSng` diese Zeichenkette in die Zahl 0 konvertierte, die anschließend der Variablen `kurs` zugewiesen wird.

Solange nicht eine dieser beiden Bedingungen erfüllt ist, können Sie nach einmaliger Eingabe eines Datums immer wieder einen Aktiennamen und einen zugehörigen Kurs eingeben und damit etwa folgende programmgesteuerten Aktualisierungen erzielen (Abbildung 36.17).

Abbildung 36.17:
Programm-
gesteuerte
Eingaben

	A	B	C	D	E
1	Datum	BASF	DAIMLER	DEUTSCHE	HÖCHST
2	01.01.09	204,00	503,00	783,00	273,00
3	02.01.09	205,50	505,50	782,00	275,00
4	03.01.09	207,00	508,00	779,00	285,00
5	04.01.09	206,00	512,00	785,00	296,00
6	05.01.09	206,80	509,00	788,50	284,00
7	06.01.09	208,5	511	792,3	283,2
8					

Formatierungs-
Eigenschaften

Wenn Sie wollen, können Sie die Einträge auch gleich noch programmgesteuert formatieren, indem Sie die Eigenschaft *Schriftart* von *Range*-Objekten und ihre Eigenschaften benutzen. Beispielsweise formatiert der Ausdruck

```
Range("A3:B4").Font.Bold = True
```

den Bereich A3:B4 fett: *Range(»A3:B4«)* übergibt einen Bezug auf diesen Bereich, also ein *Range*-Objekt. Dieser Objekttyp besitzt unter anderem die Eigenschaft *Font*, die ein Objekt vom gleichnamigen Typ *Font* übergibt. Objekte dieses Typs besitzen wiederum die Eigenschaften *Fett*, *Kursiv*, *Größe* etc., denen Sie beliebige Werte zuweisen können.

Kommen wir jedoch stattdessen zu einer weitaus wichtigeren Programmerweiterung. Eine Aktien-Tabelle wird sich sicherlich nicht allein in einer Mappe befinden, sondern – falls der Anwender auf Optik Wert legt – zusammen mit einem Diagrammblatt, das die in der Tabelle enthaltenen Kurse grafisch darstellt.

36.4 Demo »Diagrammaktualisierung«

Gehen wir davon aus, dass Sie die Tabelle erstellt haben, manuell ein paar Werte eingetragen und anschließend auf der Basis dieser Werte ein separates Diagrammblatt namens »Kursdiagramm« erzeugt haben (Abbildung 36.18).

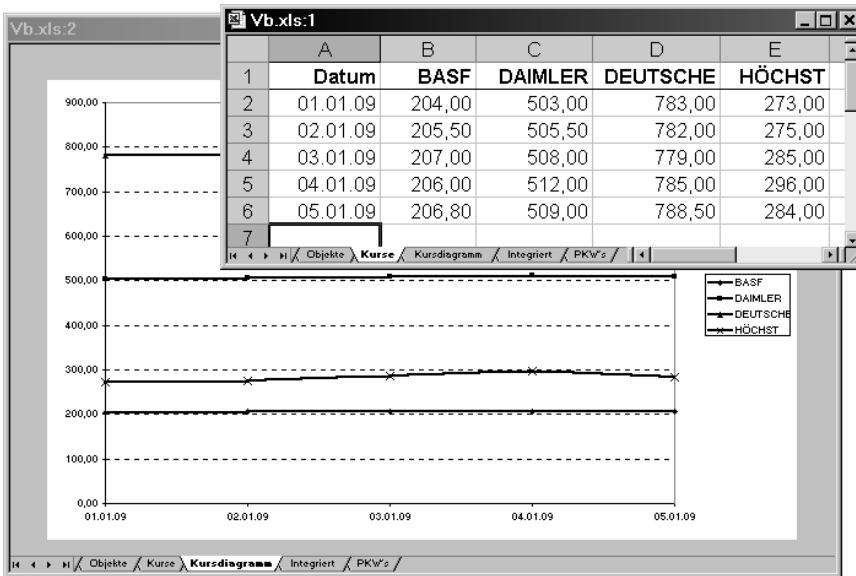


Abbildung 36.18: Vb.xls, Diagrammblatt »Kursdiagramm«

Alles Weitere soll das Programm automatisch erledigen. Es soll den Anwender nach dem Aufruf wie zuvor nach dem aktuellen Datum und den zugehörigen Kursen fragen. Anschließend soll es jedoch zusätzlich auch noch das bestehende Diagramm um die neu eingegebenen Kurse erweitern.

Diese Aufgabe klingt kompliziert, ist jedoch lösbar – und verschafft Ihnen einiges an zusätzlichem Wissen, beispielsweise über den effizienten Umgang mit Tabellenbereichen.

Paste-Methode Ich bin überzeugt, dass jeder Programmierer die gestellte Aufgabe auf andere Art und Weise lösen würde. Mein Ansatzpunkt ist die Methode *Paste*. Sie kann auf unterschiedlichste Objekte angewendet werden und fügt in das betreffende Objekt den aktuellen Inhalt der Zwischenablage ein.

Enthält die Zwischenablage die soeben eingegebenen Daten und wird die Methode auf ein *Chart*-Objekt angewandt, wird das Diagramm entsprechend um die betreffenden Datenpunkte erweitert. Im Beispiel wird dazu der Ausdruck

```
Charts("Kursdiagramm").Paste
```

benötigt.

Das *Charts*-Objekt ist analog zu *Worksheets* ein Objekt, das eine Liste aller Diagramme der aktiven Mappe enthält (eine Liste *aller* Diagramme, nicht nur der Diagrammblätter, sondern auch der integrierten Diagramme). Übergeben Sie diesem Objekt den Namen eines Diagramms, so übergibt es einen Bezug darauf, den Sie stellvertretend für das Diagramm selbst verwenden können.

Charts(»Kursdiagramm«) übergibt somit einen Bezug auf das *Chart*-Objekt mit dem Namen »Kursdiagramm«; auf dieses Diagramm wird anschließend die *Paste*-Methode angewendet, um den Inhalt der Zwischenablage ins Diagramm einzufügen.

Copy-Methode Nun stellt sich die Frage, wie wir den zusätzlichen Tabellenbereich in die Zwischenablage bekommen: mit der Methode *Copy*. Sie kopiert das Objekt, auf das sie angewendet wird, in die Zwischenablage.

Copy kann unter anderem auf das Objekt *Range* angewendet werden. Wir benötigen also einen Bezug auf den Bereich, der die neuen Daten enthält, wenden die *Copy*-Methode auf ihn an, um ihn in die Zwischenablage zu befördern, und fügen den Bereich anschließend mit der *Paste*-Methode in das Diagramm ein.

Schauen Sie sich bitte noch einmal die letzte Abbildung an. In diesem Beispiel wäre der benötigte Bezug ein Bezug auf den Bereich A7:E7. Am folgenden Tag wäre es der Bereich A8:E8 und so weiter.

Allgemein ausgedrückt: Der Bereich ist zeilenorientiert, beginnt mit jener Zelle, in der das Datum eingetragen wurde, und endet mit der letzten nicht

leeren Zelle der Zeile – vorausgesetzt, der Anwender gibt wie in dieser Abbildung auch tatsächlich alle Kurse ein und es sind daher keine Lücken zwischen den einzelnen Kursen.

Als Erstes benötigen wir einen Bezug auf das linke Bereichsende, auf jene Zelle, in der soeben das aktuelle Datum eingetragen wurde. Dazu wird der Programmteil »Datum erfragen und eintragen« gegenüber den vorhergehenden Versionen minimal geändert:

```
'Datum erfragen und eintragen
heute = CDate(InputBox("Datum?", , CStr(Date)))
Workbooks("vb.xls").Activate
Worksheets("Kurse").Activate
Set b = Range("A:A")
Set sstart = b.Find("")
sstart.Value = heute
```

Listing 36.9:
Zelle mit aktuellem
Datum merken

Der Unterschied betrifft eigentlich nur die beiden letzten Zeilen. Statt

```
Set b = b.Find("")
b.Value = heute
```

werden nun die beiden folgenden Zeilen verwendet:

```
Set sstart = b.Find("")
sstart.Value = heute
```

sstart ist eine Variable vom Typ *Range*. Statt wie zuvor *b* wird ihr nun der von der *Find*-Methode übergebene Bereichsbezug auf die erste leere Zelle in Spalte A zugewiesen.

Danach wird mit *sstart.Value = heute* im betreffenden Bereich, das heißt in dieser Zelle, wieder das Datum gespeichert. Alles verläuft wie zuvor, nur dass sich nun in der *Range*-Variablen *sstart* ein Bereichsbezug auf diese Zelle befindet.

Der folgende Programmteil »Kurs erfragen und eintragen« muss ebenfalls nur minimal verändert werden:

```
'Kurs erfragen und eintragen
Do
    aktie = InputBox("Aktie?")
    If aktie = "" Then Exit Do
    kurs = CSng(InputBox("Kurs?"))
    If kurs = 0 Then Exit Do
    Set b = Range("1:1")
    Set b = b.Find(aktie)
    Set b = b.EntireColumn
```

Listing 36.10:
Schleife statt
Prozedur verlassen

Listing 36.10:
Schleife statt
Prozedur verlassen
(Forts.)

```
b.Find("").Value = kurs
Loop While True
```

In der vorhergehenden Programmversion wurde die Prozedur mit *Exit Sub* verlassen, wenn der Anwender keinen Aktiennamen oder keinen Kurs mehr eingab. Nun wird stattdessen nur mit *Exit Do* die Schleife verlassen, nicht aber die Prozedur selbst.

Denn an deren Ende befindet sich nun folgender zusätzlicher Programmteil, der nach der beendeten Kurseingabe ausgeführt werden muss:

Listing 36.11:
Diagramm
aktualisieren

```
'Diagramm aktualisieren
Set sende = sstart.End(xlToRight)
Range(sstart, sende).Copy
Charts("Kursdiagramm").Paste
End Sub
```

Ende-Eigenschaft

sstart enthält einen Bereichsbezug auf die Zelle mit dem eingegebenen Datum. Die Anweisung

```
Set sende = sstart.End(xlToRight)
```

bezieht sich auf die Eigenschaft *End* von *Range*-Objekten. Diese Eigenschaft simuliert die Betätigung einer der Tastenkombinationen **Ende**+**→**, **Ende**+**←**, **Ende**+**↓** oder **Ende**+*Pfeil*; also sie stimuliert einen blockweisen Sprung in die betreffende Richtung, ausgehend vom übergebenen Bereichsbezug, im Beispiel also ausgehend von der Datumzelle.

Die *End*-Eigenschaft besitzt die Syntax

```
Objekt.End(Richtung)
```

Richtung ist eine der in Excel vordefinierten Konstanten *xlToRight*, *xlToLeft*, *xlUp* und *xlDown*. Abhängig von der gewünschten Sprungrichtung setzen Sie die zugehörige Konstante ein. Im Beispiel simuliert der Aufruf

```
sstart.End(xlToRight)
```

einen blockweisen Sprung nach rechts, ausgehend von der Datumzelle. Der Sprung führt somit zur letzten nicht leeren Zelle der Zeile, also zum letzten Aktienkurs.

Die *End*-Eigenschaft übergibt nach dem Sprung einen Bereichsbezug auf die Zelle, zu der der Sprung führte. Dieser Bezug wird im Programm der *Range*-Variablen *sende* zugewiesen. *sstart* enthält somit einen Bereichsbezug auf den Anfang und *sende* einen Bereichsbezug auf das Ende des zu kopierenden Bereichs.

Auf den von *sstart* bis *sende* reichenden Bereich wird die *Copy*-Methode angewandt, um seinen Inhalt in die Zwischenablage zu befördern. Für einen solchen mehr als eine Zelle umfassenden Bereichsbezug können Sie wie bereits früher gezeigt den Ausdruck *Range(Ecke_oben_links:Ecke_unten_rechts)* verwenden. Sie können beispielsweise mit

```
Range("A1:C3").Copy
```

der *Copy*-Methode einen Bezug auf den Bereich A1:C3 übergeben. Angewandt auf das Demoprogramm übergibt entsprechend der Ausdruck

```
Range(sstart, sende).Copy
```

den benötigten Bereichsbezug. Die darin enthaltenen Daten werden in die Zwischenablage kopiert und mit

```
Charts("Kursdiagramm").Paste
```

ins Diagramm eingefügt (Abbildung 36.19).

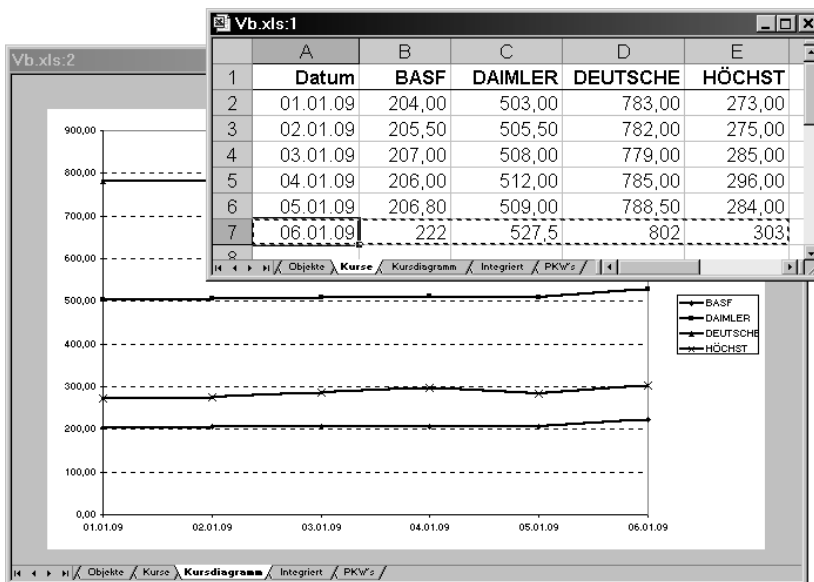


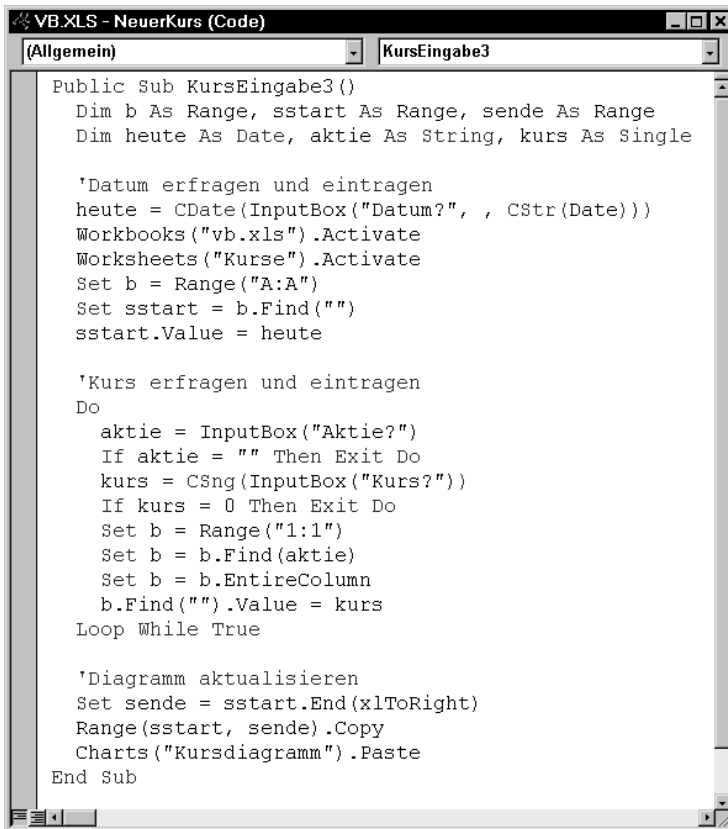
Abbildung 36.19:
Diagramm-
aktualisierung

Das Diagramm enthält nun auch die Kurse vom 6.1.2009, die erheblich höher sind als die Kurse vom Vortag.

Zum Abschluss zeige ich Ihnen noch das vollständige Programm, das zusammen mit den vorhergehenden Varianten im Modul »NeuerKurs« der Mappe VB.XLS enthalten ist, die auch das Diagramm und die Kurs-Tabelle enthält (Abbildung 36.20).

Abbildung 36.20:

Vb.xls, Modul
»NeuerKurs«,
Prozedur
»Kurseingabe3«



Das war's. Ich denke, dieses Kapitel stellt eine ausreichende Einführung in das Manipulieren von Tabellen dar. Mehr kann es nicht leisten, da Excel viel zu viele Objekte, zugehörige Eigenschaften und Methoden zum Umgang mit Tabellen und Diagrammen besitzt, um auf alle eingehen zu können – zumindest im Rahmen eines Buchs, das sich eigentlich nur am Rande mit der Programmierung in Excel beschäftigt.

37 Formulare und Steuerelemente

Im folgenden Kapitel geht es um den professionellen Touch Ihrer VBA-Anwendungen, darum, eigendefinierte Dialogfelder einzubauen, die die üblichen Steuerelemente wie Befehlsschaltflächen, Optionsfelder etc. enthalten.

Ebenso wie Tabellenblätter selbst reagieren all diese Steuerelemente auf verschiedenste Ereignisse. Sie können an diese Ereignisse Prozeduren binden, die beim Eintreten des betreffenden Ereignisses automatisch von VBA aufgerufen werden.

Das ermöglicht Ihnen beispielsweise, in eine Tabelle eine Befehlsschaltfläche mit der Aufschrift »Speichern« einzubauen. Klickt der Anwender darauf, so wird eine von Ihnen erstellte Prozedur aufgerufen, die die Tabelle speichert.

Ebenso gut können Sie dem Anwender einer Excel-Buchführungstabelle beispielsweise ein Menü BEARBEITEN zur Verfügung stellen, das unter anderem den Befehl EINGABE enthält. Wählt er ihn, öffnet die zugehörige Prozedur ein selbstdefiniertes Dialogfeld, in dem er in einem Listefeld eine vorgegebene Sparte wie »Auto« oder »Post« selektiert und in einem Textfeld den zugehörigen Wert einträgt.

Schließt er das Dialogfeld, trägt die Prozedur den betreffenden Wert in die erste freie Zelle der Spalte »Auto« bzw. »Post« ein.

Bei diesem programmgesteuerten Umgang werden dem Anwender praktisch alle Möglichkeiten genommen, überflüssige Fehler zu begehen (Wert in falsche Spalte eintragen, bereits vorhandenen Wert überschreiben etc.) und die Buchführungs-Tabelle ist soweit wie möglich vor dem Anwender geschützt.

Ich erläutere zunächst, wie Steuerelemente wie beispielsweise Befehlsschaltflächen in Tabellen oder Formulare eingefügt werden und wie Sie die Eigenschaften des betreffenden Steuerelements einstellen, u.a. seine Aufschrift und seine Farbe festlegen.

Danach zeige ich Ihnen, wie Ereignisprozeduren erstellt werden, das heißt Prozeduren, die automatisch aufgerufen werden, wenn mit dem betreffenden Steuerelement ein bestimmtes Ereignis eintritt, beispielsweise auf eine Schaltfläche geklickt wird oder durch Anklicken der Pfeile neben einer Drehen-Schaltfläche der zugehörige Wert geändert wird.

Statt in Tabellen können Sie Steuerelemente auch in Formulare einfügen, also in selbstdefinierte Dialogfelder, die sich genauso verhalten wie die Dialogfelder von Excel oder beliebigen anderen Windows-Anwendungen.

Nachdem Sie wissen, wie derartige Formulare erzeugt und gestaltet werden, gehe ich zum Abschluss im Detail auf die verschiedenen Steuerelementarten und ihre Eigenschaften ein.

Das dadurch vermittelte Wissen wird zum Abschluss in eine größere Anwendung umgesetzt, die es Ihnen ermöglicht, eine Aktien-Tabelle täglich per Formular zu aktualisieren: Statt in die Tabelle selbst tragen Sie die neuen Tageskurse und das Datum in das Formular ein, klicken darin auf *Hinzufügen* und das zugehörige VBA-Programm trägt die eingegebenen Werte daraufhin an passender Stelle in die zu Grunde liegende Tabelle ein, sodass es wesentlich weniger Fehlerquellen als beim manuellen Eintragen in die Tabelle gibt.



Die in diesem Kapitel erläuterten Formulare und Prozeduren befinden sich zum größten Teil in der Mappe VB.XLS auf der Begleit-CD.

37.1 Steuerelemente in Tabellen

Die Erstellung individueller Excel-Anwendungen basiert darauf, in Tabellen oder Formulare Steuerelemente einzufügen. Anschließend stellen Sie mit Hilfe des Eigenschaftenfensters die Eigenschaften des betreffenden Steuerelements wie gewünscht ein und definieren damit beispielsweise seinen Namen, seine Farbe etc.

Im letzten Schritt binden Sie mithilfe des Klassenmoduls jener Tabelle bzw. jenes Formulars, in dem sich das Steuerelement befindet, Ereignisprozeduren an das Steuerelement, die spezielle Aufgaben erledigen sollen.

Diese Ereignisprozeduren werden von VBA automatisch ausgeführt, wenn das betreffende Ereignis mit dem Steuerelement eintritt, wenn beispielsweise auf eine Befehlsschaltfläche geklickt wird (Eintreten des Ereignisses *Click* für die Schaltfläche).

Die einfachste Technik, Steuerelemente einzusetzen, besteht darin, sie in ganz normale Excel-Tabellen einzufügen (Abbildung 37.1).

In diesem Beispiel wurde rechts neben der Zelle C2 eine Drehen-Schaltfläche eingefügt und mit der Zelle C2 verknüpft.

Mit Hilfe der Schaltfläche kann nun an dem in C2 enthaltenen Wert gedreht werden: Klicken Sie auf den aufwärts gerichteten Pfeil, so wird der in der Zelle enthaltene Wert um eins erhöht, also aus der Vier eine Fünf gemacht; klicken Sie auf den abwärts gerichteten Pfeil, wird der Zellinhalt um eins vermindert.



Abbildung 37.1:

Vb.xls, Tabelle
»Integriert«

37.2 Der Entwurfsmodus und das Eigenschaftfenster

Um diese Schaltfläche in die Tabelle einzufügen, müssen Sie die abgebildete Symbolleiste Steuerelement-Toolbox einblenden. Darin klicken Sie auf das gewünschte Steuerelement und danach auf jene Stelle der Tabelle, wo es eingefügt werden soll.

Durch das Einfügen des Steuerelements wird automatisch das abgebildete Symbol *Entwurfsmodus* hervorgehoben: der so genannte Entwurfsmodus ist aktiviert.



In diesem Modus können Sie die Eigenschaften des Steuerelements manipulieren. Zunächst einmal können Sie damit die üblichen Operationen durchführen, die Sie bereits von Grafikobjekten her kennen:

- ➔ Auf dem Steuerelement verändert der Cursor seine Form und wird zu einem Fadenkreuz. Sie können das Steuerelement nun anpacken und verschieben.
- ➔ Klicken Sie auf das Objekt, so werden Anfasser sichtbar, mit denen Sie die Objektgröße und die Objektform verändern können.
- ➔ FORMAT|STEUERELEMENT... öffnet ein Dialogfeld mit den bekannten Registern *Größe*, *Schutz*, *Eigenschaften* und *Web*.

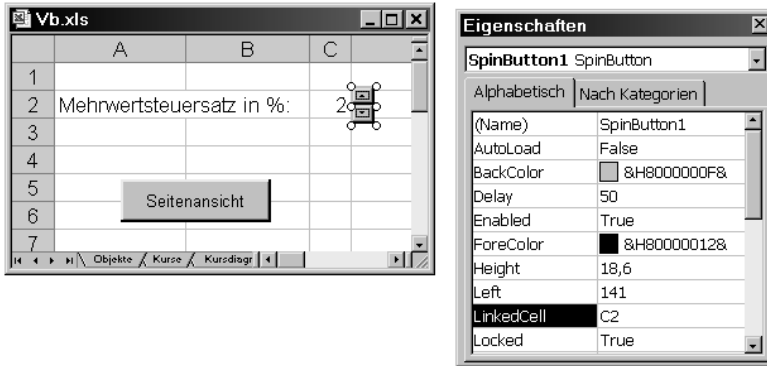
Vor allem aber können Sie nur im Entwurfsmodus alle Eigenschaften des Objekts mithilfe des bereits in Kapitel 32.1, »VBA-Projekte – ein Überblick«, erwähnten Eigenschaftfensters manipulieren.



Um es einzublenden, benutzen Sie das abgebildete Symbol, das sich ebenfalls in der Leiste *Steuerelement-Toolbox* befindet (Abbildung 37.2).

Ist der Entwurfsmodus aktiviert, enthält das Listenfeld dieses Fensters das aktive Objekt, beispielsweise die aktive Tabelle »Integriert« und alle wiederum darin enthaltenen Objekte, im Beispiel also die Drehen-Schaltfläche.

Abbildung 37.2:
Entwurfsmodus und
Steuerelement-
eigenschaften



Selektieren Sie eines dieser Objekte im Listenfeld oder klicken Sie einfach auf das Objekt selbst, so werden in der linken Spalte die Eigenschaften des Objekts angezeigt und in der rechten die aktuellen Einstellungen dieser Eigenschaften.

Die momentan wichtigste Eigenschaft ist *LinkedCell*: Sie bestimmt, mit welcher Zelle das Steuerelement verknüpft ist. Nach dem Einfügen des Steuerelements ist diese Einstellung zunächst leer. Das Steuerelement ist ungebunden und beim Klicken auf einen der beiden Pfeile passiert überhaupt nichts.

Um die Drehen-Schaltfläche mit dem Inhalt der Zelle C2 zu verknüpfen, trug ich in der rechten Spalte einfach »C2« ein. Daraufhin übte diese Schaltfläche die erläuterte Funktion aus und ermöglichte die Manipulation des Inhalts der Zelle C2.



Im Entwurfsmodus ist es nicht möglich, Änderungen der Eigenschaftseinstellungen auch auszuprobieren. Dazu muss der Entwurfsmodus zunächst durch erneutes Anklicken des entsprechenden Symbols wieder deaktiviert werden!

Name-Eigenschaft

Die wichtigste Eigenschaft jedes Steuerelements ist *(Name)*, der Name des Objekts, unter dem es in VBA-Programmen angesprochen werden kann.

Nach dem Einfügen eines Steuerelements erhält es zunächst einen Standardnamen wie »SpinButton1«, falls es sich um eine Drehen-Schaltfläche handelt, oder »CommandButton1«, wenn sich um eine Befehls-Schaltfläche handelt. Weitere Steuerelemente dieses Typs erhalten entsprechend den Namen »SpinButton2« bzw. »CommandButton2«.

Sie können diese Namen unverändert lassen. Allerdings ist es nicht sonderlich angenehm, sich anschließend in VBA-Programmen mit Ausdrücken wie

```
ActiveSheet.SpinButton1.LinkedCell = "C2"
```

auf ein solches Steuerelement zu beziehen. Enthält die Tabelle nur eine Drehen-Schaltfläche, ist das noch unproblematisch. Enthält sie jedoch zehn oder mehr Drehen-Schaltflächen, wird es schon problematischer. Daher empfehle ich Ihnen, Ihren Steuerelementen aussagekräftige Namen zu geben.

Im Beispiel, der Tabelle »Integriert«, würde es sich anbieten, die Drehen-Schaltfläche, die ja laut dem in A1 enthaltenen Text einen Umsatzsteuersatz dreht, den Namen »USt« zu geben.

Verwechseln Sie keinesfalls die Eigenschaft (Name) mit der Eigenschaft Caption vieler Steuerelemente! (Name) ist wie gesagt der interne Name, den Sie in Bezügen auf das Steuerelement verwenden müssen und er ist ungeheuer wichtig, da bei Verwendung des falschen Namens Ihr Programm nicht funktionieren kann.



!!
STOP

Caption ist dagegen die Aufschrift des Steuerelements, beispielsweise die Aufschrift »OK« oder »Abbrechen« einer Befehlsschaltfläche – und im Gegensatz zu (*Name*) somit ein im Grunde völlig bedeutungsloser Text.

Beides ist vollkommen unabhängig voneinander. Nehmen Sie als Beispiel dafür das zweite in der Tabelle »Integriert« von VB.XLS enthaltene Steuerelement, eine Befehls-Schaltfläche.

Sie wird auf die gleiche Weise eingefügt wie die Drehen-Schaltfläche: In der Symbolleiste *Steuerelement-Toolbox* müssen Sie auf das entsprechende Symbol klicken und danach auf jene Stelle der Tabelle, an der die Schaltfläche eingefügt werden soll.

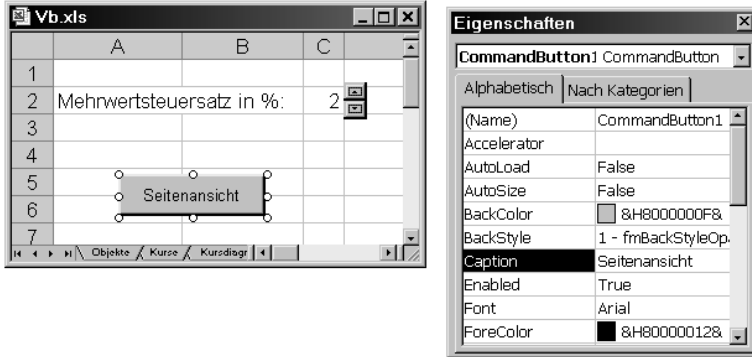
Danach ist erneut der Entwurfsmodus aktiviert und die Eigenschaften der Befehls-Schaltfläche können verändert werden. Sie unterscheiden sich von denen der Drehen-Schaltfläche, beispielsweise gibt es keine Eigenschaft *LinkedCell*, Befehls-Schaltflächen können also nicht an irgendwelche Tabellenzellen gebunden werden.

Dennoch ist auch diesmal das Eigenschaftenfenster nützlich, beispielsweise um wie hier für die Eigenschaft *Caption* den Text »Seitenansicht« einzutragen und der Schaltfläche damit diese Aufschrift zu verleihen (Abbildung 37.3).

Die Aufschrift der Befehls-Schaltfläche lautet somit »Seitenansicht«, ihr Name, den Sie in Bezügen darauf angeben müssen, ist jedoch weiterhin »CommandButton1«. Wie gesagt, beide Eigenschaften sind vollkommen unabhängig voneinander und dürfen keinesfalls verwechselt werden!

Abbildung 37.3:

»(Name)«-
und »Caption«-
Eigenschaften



37.3 Ereignisprozeduren

Befehls-Schaltflächen werden üblicherweise verwendet, um irgendeine Aktion auszuführen, wenn der Anwender auf die Schaltfläche klickt. Um diese Aktion zu definieren, benötigen wir eine Ereignisprozedur, eine Prozedur, die automatisch aufgerufen wird, wenn dieses Ereignis eintritt, wenn also auf die Schaltfläche geklickt wird.



Wie ich in Kapitel 32.1.3, »Ereignisprozeduren«, erläuterte, gehört zu jeder Excel-Tabelle ein Klassenmodul, in dem sich alle Ereignisprozeduren für die Tabelle selbst und für alle darin eingefügten Steuerelemente befinden.

Um es zu öffnen, können Sie den Visual Basic Editor aktivieren und im Projekt-Explorer auf die betreffende Tabelle doppelklicken, beispielsweise auf »Integriert«.

Einfacher: doppelklicken Sie bei aktiviertem Entwurfsmodus auf das Steuerelement selbst. Dadurch wird ebenfalls Visual Basic aktiviert und das Klassenmodul der Tabelle geöffnet, zusätzlich jedoch wird im *Objekt*-Listefeld automatisch das betreffende Steuerelement selektiert und im *Prozedur*-Listefeld die zu diesem Steuerelement passende Ereignisprozedur (Abbildung 37.4).

»Passend« ist von Steuerelement zu Steuerelement unterschiedlich. Bei einer Drehen-Schaltfläche ist häufig das *Change*-Ereignis interessant, das eintritt, wenn durch Anklicken eines der beiden Pfeile der zugehörige Wert geändert wird.

Entsprechend würde VBA die Ereignisprozedur *SpinButton1_Change* öffnen, wenn der Name der Schaltfläche »SpinButton1« lautet (also die Einstellung der Eigenschaft (*Name*)).

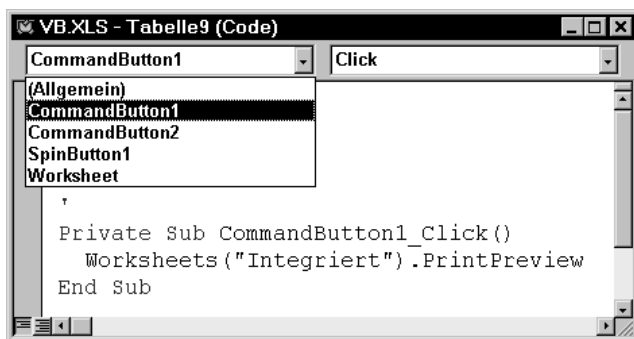


Abbildung 37.4:
Vorgabe der
passenden
Ereignisprozedur

Bei einer Befehls-Schaltfläche ist dagegen das beim Klicken darauf eintretende Ereignis *Click* interessanter und VBA öffnet somit entsprechend der Abbildung die Ereignisprozedur *CommandButton1_Click*, da die Schaltfläche den Namen »CommandButton1« besitzt.

Diese Prozedur enthält genau eine von mir eingegebene Anweisung:

```
Worksheets("Integriert").PrintPreview
```

Sie wendet auf die Tabelle »Integriert« die Methode *PrintPreview* an, aktiviert also die Seitenansicht der Tabelle.

Das Resultat: Wird in der Tabelle »Integriert« auf die Schaltfläche geklickt, tritt für dieses Objekt namens »CommandButton1« das Ereignis *Click* ein, die zugehörige Ereignisprozedur *CommandButton1_Click* wird ausgeführt und dadurch wird die Seitenansicht aktiviert.

Vergessen Sie nicht, nach Änderungen von Ereignisprozeduren den Entwurfsmodus zum Ausprobieren zu deaktivieren. Nur im Normalmodus werden Ereignisse ausgelöst und die Ereignisprozeduren aufgerufen, die diesen Ereignissen zugeordnet sind.

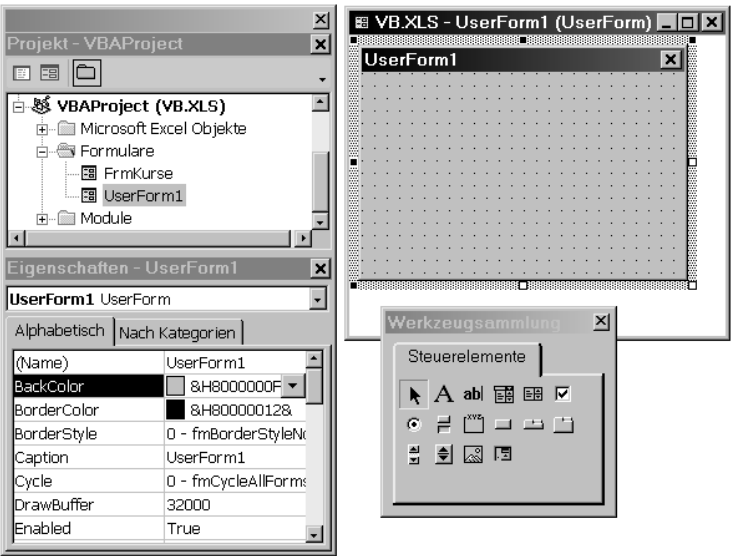


37.4 Formulare

Statt in Tabellen können Sie Steuerelemente auch in Formulare einfügen. Formulare sind praktisch selbstdefinierte Dialogfelder, die sich genauso verhalten wie die Dialogfelder von Excel oder beliebigen anderen Windows-Anwendungen. Um ein solches selbstdefiniertes Dialogfeld zu erstellen, fügen Sie mit EINFÜGEN|USERFORM ein leeres Formular in Ihr Projekt ein (Abbildung 37.5).

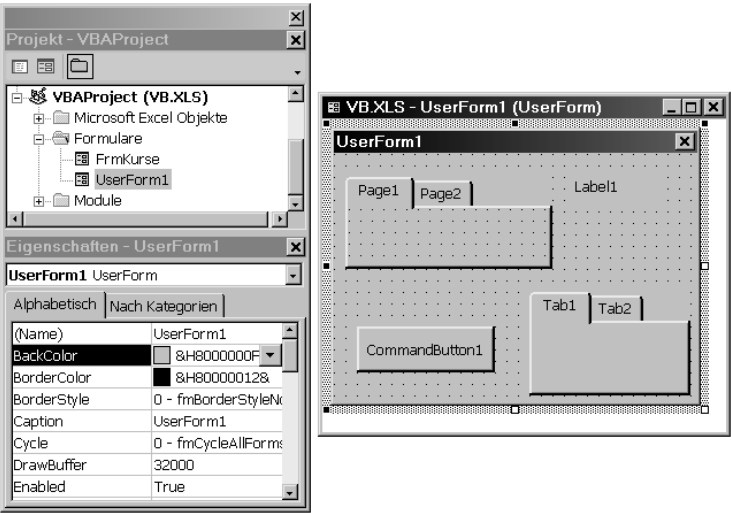
Sie können das Formular wie ein Grafikobjekt verschieben oder durch Ziehen/Schieben an einem der Anfasser seine Größe und Form verändern.

Abbildung 37.5:
Leeres Formular
einfügen



Mit der automatisch eingeblendeten Werkzeugsammlung können Sie unterschiedlichste Steuerelemente einfügen. Sie klicken auf das betreffende Symbol und danach dort auf das Formular, wo es eingefügt werden soll (Abbildung 37.6).

Abbildung 37.6:
Steuerelemente
einfügen



Um auszuprobieren, wie das resultierende Dialogfeld aussieht, wählen Sie AUSFÜHREN|SUB/USERFORM AUSFÜHREN oder klicken auf das abgebildete Symbol. Daraufhin wird Excel aktiviert und Ihr Formular erscheint, so als hätten Sie in Excel einen Befehl wie DATEI|SEITE EINRICHTEN... gewählt, der ein Dialogfeld öffnet (Abbildung 37.7).

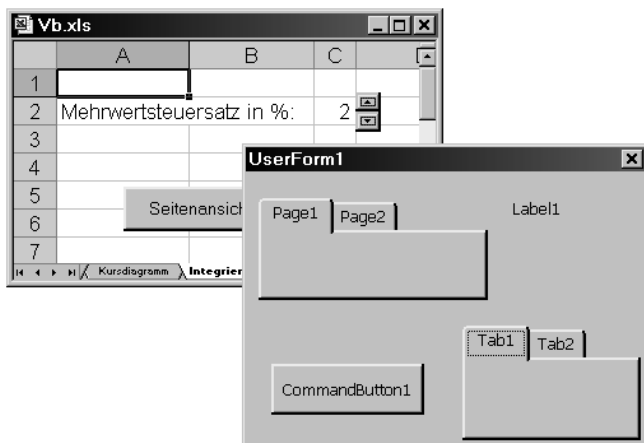


Abbildung 37.7:
Dialogfeld
ausführen

Ebenso wie bei eingebauten Excel-Dialogfeldern kann weder ein Symbol noch ein Excel-Befehl aktiviert werden, solange das Formular geöffnet ist. Da das Formular keine der beiden dazu üblichen Schaltflächen *OK* oder *Abbrechen* enthält, müssen Sie es per Anklicken des *Schließen*-Symbols in der oberen rechten Fensterecke schließen.

Alternativ dazu können Sie auch zu Visual Basic wechseln und dort **AUSFÜHREN|ENTWURFSMODUS** wählen oder auf das zugehörige Symbol klicken.

In beiden Fällen wird der vorhergehende Zustand wiederhergestellt: Sie befinden sich wieder in Visual Basic und der Entwurfsmodus ist aktiviert, in dem Sie das Formular verändern können.

Sie können zum Beispiel den nach dem Einfügen von VBA vorgegebenen Titel »UserForm1« durch einen netteren Text wie »Testformular« ersetzen. Dazu öffnen Sie das Eigenschaftfenster und klicken auf irgendeine Stelle im Formular, an der sich kein Steuerelement befindet, was den gleichen Zweck erfüllt. Anschließend ersetzen Sie unter *Caption* den Text »UserForm1« durch »Testformular«.

Viele Eigenschaften von Steuerelementen wie (*Name*) können Sie nur mit Hilfe des Eigenschaftfenster verändern. Andere Eigenschaften wie die Aufschrift einer Befehls-Schaltfläche, die Breite und Höhe eines Steuerelements können Sie dagegen entweder im Eigenschaftfenster oder aber interaktiv am Steuerelement selbst ändern.

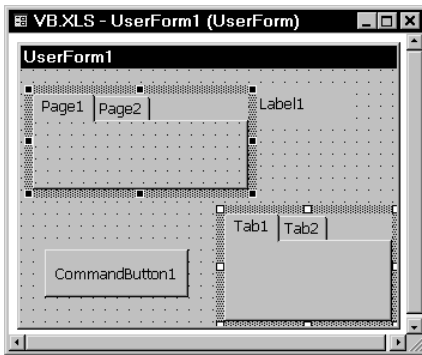
In beiden Fällen selektieren Sie zunächst das manipulierende Steuerelement durch Anklicken. Fügten Sie zuvor ein Steuerelement ein, so klicken Sie davor noch auf das abgebildete Symbol, um dem Cursor wieder seine normale Zeigeform zu verleihen.



Bei den meisten Steuerelementen, die eine Aufschrift besitzen, aktiviert nach der Selektion ein weiterer Klick (mit Pause, um keinen Doppelklick auszulösen) den Editiermodus: der Cursor wird auf das angeklickte Zeichen gesetzt und Sie können nun die Beschriftung des Steuerelements editieren.

Sie können mehrere Objekte gleichzeitig selektieren, indem Sie das erste Objekt selektieren und danach das zweite, dritte etc. Objekt anklicken, dabei jedoch die **[Strg]**-Taste gedrückt halten (Abbildung 37.8).

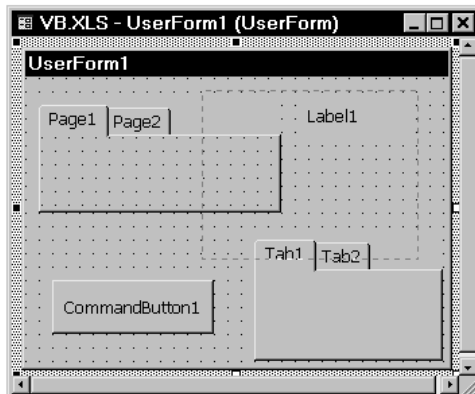
Abbildung 37.8:
Mehrere Objekte
selektieren



Anschließend können Sie alle selektierten Objekte gleichzeitig verschieben oder ihre Form/Größe verändern.

Eine oft einfachere Möglichkeit besteht darin, einen Rahmen um die zu selektierenden Objekte herumzulegen. Dazu klicken Sie die gedachte linke obere Rahmenecke an und ziehen die Maus bei gedrückter linker Taste zur rechten unteren Rahmenecke (Abbildung 37.9).

Abbildung 37.9:
Alternativtechnik



Lassen Sie die Maustaste los, so sind alle Elemente selektiert, die dieser Rahmen zumindest berührt.

Mit **Entf** können Sie alle gegenwärtig selektierten Objekte löschen.

Löschen

Stellen Sie beim Ausführen des Formulars fest, dass die Aktivierreihenfolge der Objekte bei mehrfacher Betätigung der -Taste nicht Ihren Vorstellungen entspricht, wählen Sie **ANSICHT|AKTIVIERREIHENFOLGE** bzw. klicken auf das zugehörige Symbol (Abbildung 37.10).

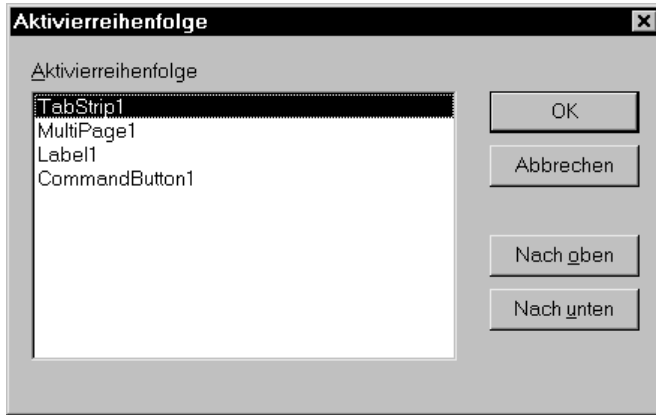


Abbildung 37.10:
Aktivierungsfolge

Die Reihenfolge der Steuerelementnamen entspricht der Reihenfolge, in der diese Steuerelemente bei mehrfacher Betätigung von  aktiviert werden.

Um die Reihenfolge zu verändern, klicken Sie auf ein Element der Liste und wählen *Nach oben*, um es nach oben zu verschieben, bzw. verschieben es mit *Nach unten* weiter nach unten, sodass es erst später aktiviert wird.

Sie können übrigens mit den üblichen Techniken mehrere Listenelemente selektieren und gleichzeitig verschieben.

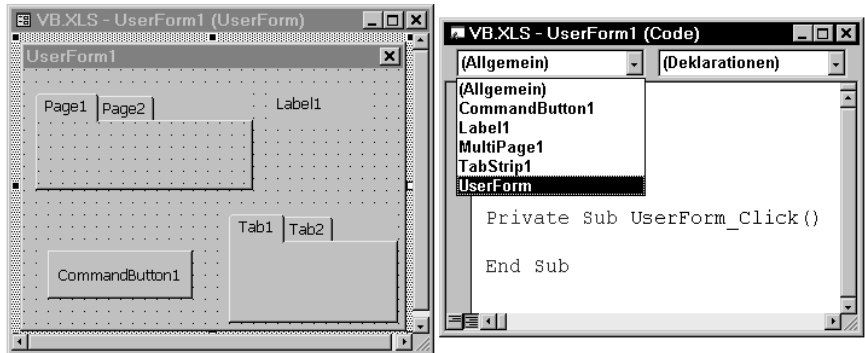
Ebenso wie eine Tabelle besitzt auch ein Formular ein Klassenmodul, in dem sich alle Ereignisprozeduren für das Formular selbst und für alle darin eingefügten Steuerelemente befinden. Um das Klassenmodul eines Formulars zu öffnen, doppelklicken Sie im Entwurfsmodus auf das Formular selbst oder auf irgendein darin enthaltenes Steuerelement (Abbildung 37.11).

Klassenmodul

VBA öffnet das Klassenmodul und gibt eine an das Formularereignis *Click* gebundene Ereignisprozedur vor; heißt das Formular »UserForm«, heißt die Ereignisprozedur also entsprechend *UserForm_Click*.

Sie wird aufgerufen, wenn der Anwender auf irgendeine Stelle im Formular selbst klickt, also auf einen Punkt, an dem sich kein Steuerelement befindet (sonst würde stattdessen das Ereignis *Click* für das betreffende Steuerelement ausgelöst).

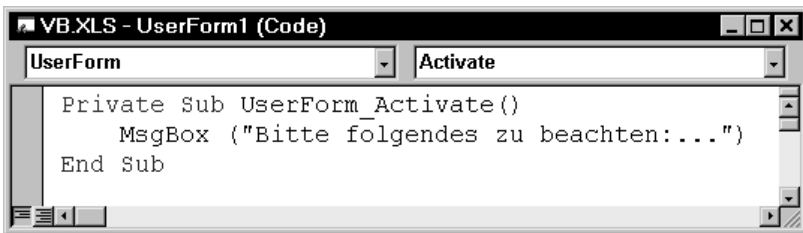
Abbildung 37.11:
Klassenmodul eines
Formulars



Im *Prozedur*-Listenfeld können Sie sich beliebige andere Formularereignisse aussuchen, um die zugehörigen Ereignisprozeduren zu öffnen. Besonders interessant ist beispielsweise das *Activate*-Ereignis, das beim Aktivieren des Formulars eintritt, also unmittelbar nach dem Ausführen.

Beispielsweise können Sie an dieses Ereignis eine Prozedur binden, die dem Anwender sofort nach Erscheinen des Formulars irgendwelche Hinweise zu seiner Benutzung gibt (Abbildung 37.12).

Abbildung 37.12:
»Activate«-Ereignis



Daraus resultiert, dass unmittelbar nach dem Ausführen den Formulars zunächst diese Hinweise erscheinen und vom Anwender studiert und mit OK bestätigt werden müssen, bevor er mit dem Formular selbst arbeiten kann (Abbildung 37.13).

Andere Anwendungsmöglichkeiten könnten sein, beim Öffnen des Formulars wichtige Variablen zu initialisieren oder im Formular darzustellende Texte aus einer Datei einzulesen etc.

Auch so manch anderes Formularereignis ist recht interessant. Wählen Sie einfach mal im *Objekt*-Listenfeld das Formularobjekt aus und schauen Sie sich dann im *Prozedur*-Listenfeld die für dieses Objekt verfügbaren Ereignisse an. Die meisten Ereignisnamen wie *Click*, *DoubleClick* etc. sind selbsterklärend.

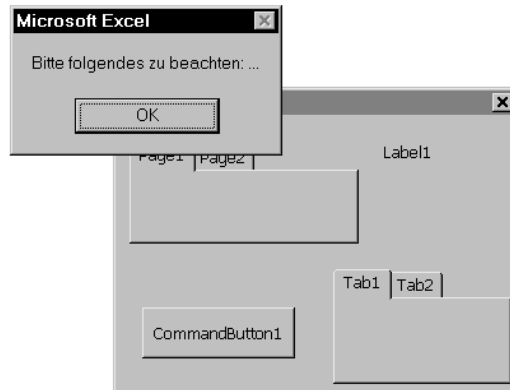


Abbildung 37.13:
Ausführung der
Prozedur

37.5 Formulare in Excel-Anwendungen integrieren

Sie wissen, dass Sie ein Formular mit `AUSFÜHREN|SUB/USERFORM AUSFÜHREN` oder dem zugehörigen Symbol ausführen können. Diese Technik ist jedoch nur zum Testen geeignet. Ist Ihre Excel-Anwendung fertig gestellt, können Sie sie schlecht einem Anwender mit den Worten überreichen: »Um das Formular zur Eingabe von Aktienkursen zu öffnen, aktivieren Sie Visual Basic, öffnen mit dem Projekt-Explorer das Formular »Kurseingabe« und wählen `AUSFÜHREN|SUB/USERFORM AUSFÜHREN`.«

Sie benötigen eine Technik, die sich für den *Benutzer* Ihrer Excel-Anwendung eignet, nicht für den Programmierer der Anwendung!

Sie haben die Wahl unter verschiedensten Techniken, die alle darauf beruhen, dass Formulare selbstverständlich auch programmgesteuert ausgeführt werden können. Dafür zuständig ist die *Show*-Methode:

```
Formularname.Show
```

Show führt das angegebene Formular aus bzw. lädt es gegebenenfalls sogar, falls das momentan nicht der Fall ist.

Noch einmal: Verwechseln Sie keinesfalls (Name) und Caption! Caption definiert die Formularaufschrift, den in der Titelleiste angezeigten Text, der jedoch eine reine Äußerlichkeit ist. Lautet der Formularname, also die Eigenschaft (Name), »UserForm1«, so müssen Sie in Bezügen auf das Formular diesen »UserForm1« verwenden, auch wenn der Titel des Formulars beispielsweise »Kurseingabe« lautet.

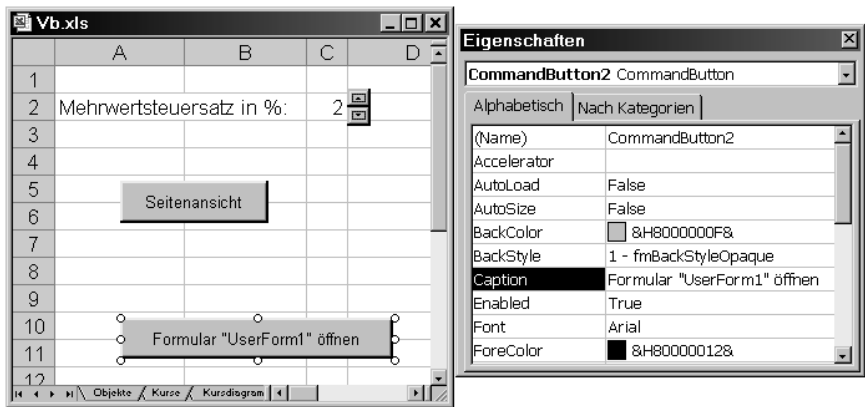
**!!
STOP**

Entsprechend würde der folgende Ausdruck dieses Formular ausführen:

```
UserForm1.Show
```

Sie können diesen Ausdruck unterschiedlich einsetzen. Eine Möglichkeit besteht darin, in Ihre Tabelle eine Befehls-Schaltfläche einzufügen, die mit »Formular »Kurseingabe« öffnen« beschriftet ist. Die Tabelle »Integriert« von VB.XLS enthält eine solche Schaltfläche (Abbildung 37.14).

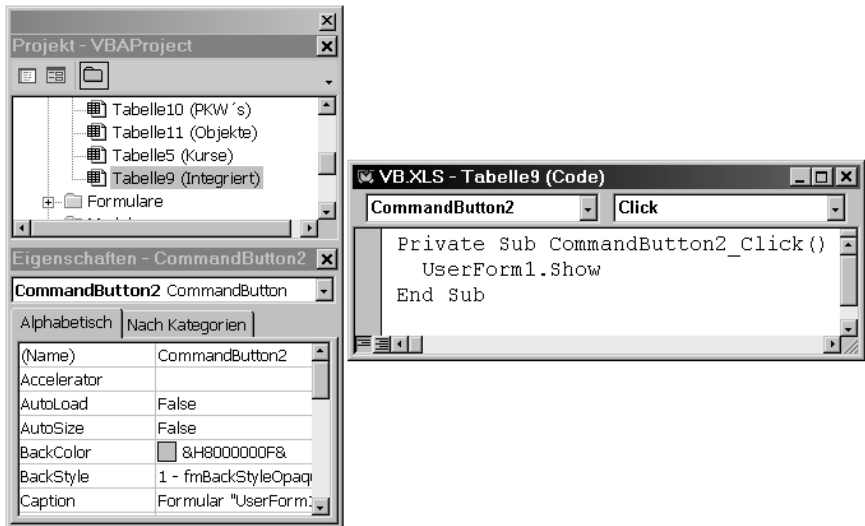
Abbildung 37.14:
Vb.xls, Tabelle
»Integriert«



Entscheidend ist, dass an das *Click*-Ereignis dieser Schaltfläche eine Ereignisprozedur gebunden ist, die die Anweisung *UserForm1.Show* enthält.

Dazu öffnete ich per Doppelklick auf die Schaltfläche (im Entwurfsmodus) das Klassenmodul der Tabelle »Integriert« und VBA gab automatisch die Ereignisprozedur *CommandButton2_Click* vor, in der ich diese Anweisung eintrug (Abbildung 37.15).

Abbildung 37.15:
Vb.xls, Klassen-
modul »Integriert«,
Prozedur
»CommandButton2
_Click«



Das hat zum Resultat, dass der Anwender das Formular »UserForm1« jederzeit öffnen kann, indem er einfach auf diese Schaltfläche klickt. Dadurch wird diese an das *Click*-Ereignis der Schaltfläche gebundene Prozedur aufgerufen und die darin enthaltene *UserForm1.Show*-Anweisung ausgeführt, die das Formular ausführt (Abbildung 37.16).

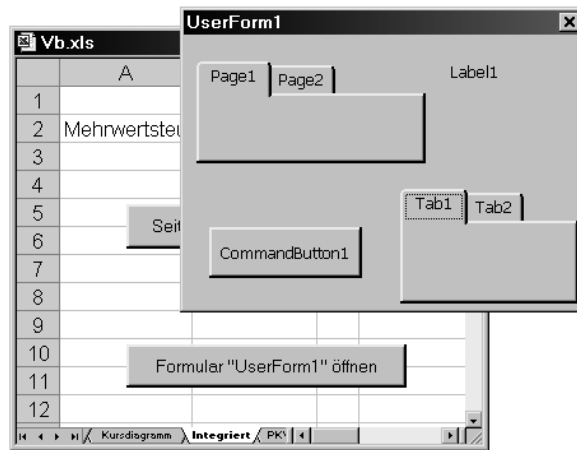


Abbildung 37.16:
Ausführen des
Formulars

Eine weitere Möglichkeit, die ich in Kapitel 37.7, »Umgang mit Steuerelementen«, benutze, besteht darin, diese Anweisung im Klassenmodul der Tabelle in die Ereignisprozedur Worksheet_Activate einzufügen, die ausgeführt wird, wenn das Ereignis Activate für die Tabelle eintritt, also wenn die Tabelle aktiviert wird. Dann wird entsprechend automatisch das zugehörige Dialogfeld erscheinen.



Die dritte und sicherlich professionellste Möglichkeit ist die Verwendung selbstdefinierter Befehls- und Symbolleisten. In Kapitel 31.3, »Makros mit Objekten verknüpfen«, beschrieb ich, wie Sie einem selbstdefinierten Befehl oder Symbol ein Makro zuweisen können, das beim Klicken auf das Symbol ausgeführt wird.



Nun sind Makros nichts anderes als *Sub*-Prozeduren, mit dem Resultat, dass Sie mit der dort beschriebenen Technik dem Symbol eine beliebige *Sub*-Prozedur zuweisen können!

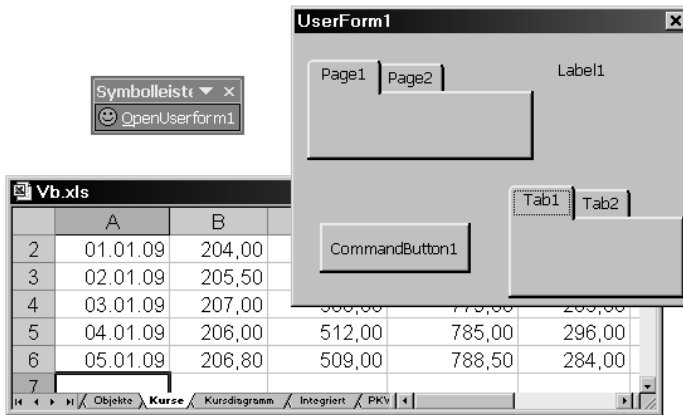
Damit wäre das Problem gelöst: Sie erstellen eine *Sub*-Prozedur, nennen wir sie »OpenUserform1«, die genau eine Anweisung enthält: *UserForm1.Show*.

Diese Prozedur binden Sie an das Symbol, indem Sie das *Anpassen*-Dialogfeld öffnen, mit der rechten Maustaste auf das Symbol klicken, im Kontextmenü MAKRO ZUWEISEN... wählen und im zugehörigen Dialogfeld Ihre Prozedur *OpenUserform1* auswählen.

Ab jetzt genügt dem Anwender ein Klick auf dieses Symbol, um das zugehörige Formular zu öffnen.

Um das auszuprobieren, sollten Sie nun VB.XLS öffnen, die Leiste *Symbolleiste1* einblenden, falls sie deaktiviert ist, und auf das einzige darin vorhandene Symbol klicken (Abbildung 37.17).

Abbildung 37.17:
Formular per
Symbol ausführen



Nun wissen Sie auch, wie echte, integrierte Excel-Anwendungen zustande kommen:

- ➔ Sie erstellen zunächst alle Tabellen, Prozeduren und Formulare, die Sie in Ihrer Anwendung benötigen.
- ➔ Danach erzeugen Sie selbstdefinierte Befehls- und Symbolleisten.
- ➔ An jeden Befehl und an jedes Symbol binden Sie eine *Sub*-Prozedur.
- ➔ Viele Befehle/Symbole wie (im Falle von Excel selbst) DATEI|SPEICHERN, die Aktionen ausführen, für die keine weiteren Anwendereingaben erforderlich sind, erfordern nur diese eine Prozedur, an die sie gebunden sind.
- ➔ Andere Befehle wie DATEI|SEITENANSICHT, bei denen Eingaben des Anwenders erforderlich sind, beginnen mit der Anweisung *Formularname.Show*, führen also zunächst ein Formular aus, in dem der Anwender die gewünschten Aktionen näher spezifiziert. Die folgenden Programmzeilen erledigen dann anhand dieser Anwendereingaben die betreffende Aufgabe.

Sie benötigen nicht mehr viel zusätzliches Handwerkszeug, um derartige Anwendungen erstellen zu können:

- ➔ Sie müssen wissen, wie Ihr Programm an die Eingaben herankommt, die der Anwender in einem Formular vornimmt, um diese Eingaben anschließend weiterverarbeiten zu können.

➔ Und Sie müssen über die Möglichkeiten und Eigenschaften der verschiedenen Steuerelemente Bescheid wissen, die Sie in Ihre Formulare einfügen können.

Beides erläutere ich in den folgenden Kapiteln, in denen auf die einzelnen Steuerelemente eingegangen wird.

Vergessen Sie nicht, eine selbstdefinierte Symbolleiste an die betreffende Arbeitsmappe anzubinden, wenn Sie die Mappe an einen Kollegen weitergeben wollen (siehe Praxisführer A, »Menü- und Symbolleisten anpassen«). Sonst wird zwar die Mappe selbst weitergegeben, die zugehörige Symbolleiste fehlt jedoch.



37.6 Programmgesteuerte Veränderung von Eigenschaften

Steuerelemente sind Unterobjekte des Formulars, in dem sie sich befinden, und in der Auflistung *Controls* des Formulars enthalten.

Wie jede Auflistung besitzt auch *Controls* unter anderem die Eigenschaft *Count*, sodass der Ausdruck *Controls.Count* die Anzahl der im Formular enthaltenen Steuerelemente übergibt. Entsprechend können Sie beispielsweise mit der folgenden Schleife alle in einem Formular enthaltenen Steuerelemente der Reihe nach ablaufen:

```
For Each Control in Formularname.Controls
..Anweisung Control.Methode/Eigenschaft
Next
```

Beispielsweise würden die folgenden Zeilen allen Steuerelementen des Formulars »UserForm1« die Schriftart »Arial« zuweisen, sodass alle zugehörigen Texte (Beschriftungen der Steuerelemente; Eingaben, die der Benutzer im betreffenden Element vornimmt) in dieser Schriftart dargestellt würden:

```
For Each Control in UserForm1.Controls
..Control.Font = "Arial"
Next
```

Derartige programmgesteuerte Veränderungen von Steuerelement-Eigenschaften können nur erfolgen, während das betreffende Dialogfeld ausgeführt wird, beispielsweise in einer an das Formularereignis Initialize gebundenen Ereignisprozedur (dieses Ereignis tritt genau einmal nach dem Laden des Formulars ein).



Beachten Sie außerdem, dass diese Veränderungen nicht dauerhaft sind! Wird das Formular geschlossen, gelten für die betreffenden Steuerelemente wieder die im Eigenschaftenfenster eingestellten Eigenschaften.

Ein Beispiel: Gibt der Benutzer im Feld »Verheiratet« »Nein« ein, könnte es sinnvoll sein, unmittelbar darauf die Eigenschaft *Visible* des folgenden Steuerelements mit der Aufschrift »Seit wann?« auf *False* einzustellen, um dieses nun sinnlose Steuerelement unsichtbar zu machen.

Wird das Formular am nächsten Tag wieder geöffnet, ist das betreffende Feld wieder ganz normal sichtbar, da wieder die im Eigenschaftenfenster vorgegebene Einstellung *True* für *Visible* gilt.

Die wichtigste Eigenschaft eines Steuerelements ist *Value*, sein Inhalt. Ebenso wie bei Zellen von Tabellen kann auf den aktuellen Inhalt eines Steuerelements mit einer Anweisung der Art

```
v = Steuerelement.Value
```

lesend zugegriffen, das heißt der Variablen *v* der aktuelle Inhalt zugewiesen werden, oder genau umgekehrt mit einer Anweisung der Art

```
Steuerelement.Value = Wert
```

dieser Wert verändert werden, was sich sofort darauf im Steuerelement widerspiegelt. Weisen Sie beispielsweise einem der im folgenden Kapitel erläuterten Textfelder, das den Namen »Textbox1« besitzt, mit

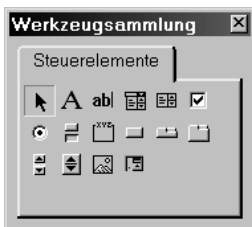
```
Textbox1.Value = "Maier"
```

als neuen Inhalt die Zeichenkette zu, erscheint im Textfeld entsprechend sofort diese Zeichenkette, so als hätte der Anwender sie eingetippt.

37.7 Umgang mit Steuerelementen

Die Werkzeugsammlung enthält verschiedenste Steuerelemente mit unterschiedlichen Eigenschaften (Abbildung 37.18).

Abbildung 37.18:
Werkzeug-
sammlung



Objekte auswählen: Macht das Anklicken eines Toolbox-Symbols rückgängig und verleiht dem Cursor wieder seine normale Zeigeform



Bezeichnungsfeld: Erzeugt ein Bezeichnungsfeld, das statischen Text enthält



Textfeld: Erzeugt ein Textfeld, in dem Text angezeigt oder vom Benutzer eingegeben werden kann



Listenfeld: Feld, das eine Liste von Elementen enthält, von denen eines oder mehrere ausgewählt werden können



Kombinationsfeld: Kombination eines Text- und eines Listenfeldes



Kontrollkästchen: Quadratische Fläche mit zwei Zuständen und entsprechend variierender Optik (Häkchen oder leer)



Optionsfeld: Runde Fläche mit zwei Zuständen und entsprechend variierender Optik (ausgefüllt oder leer)



Umschaltfeld: Rechteckige Fläche, die genau zwei Zustände besitzt, zwischen denen der Benutzer durch Anklicken umschaltet. Je nach Zustand erscheint die Fläche hervorgehoben oder normal. Beispielsweise sind alle Symbole Umschaltfelder.



Rahmen: Fasst mehrere davon umschlossene Kontrollkästchen oder Optionsfelder zu einer Optionsgruppe zusammen, in der jeweils nur eine Option ausgewählt sein kann



Befehls-Schaltfläche: Fügt eine Befehls-Schaltfläche mit einer Beschriftung wie zum Beispiel *OK*, *Abbrechen* oder *Daten eingeben* ein. Klickt der Benutzer auf die Schaltfläche, wird ein zugeordnetes Makro ausgeführt.



Multiseiten: Ermöglicht die Erzeugung von Formularen, die wie Dialogfelder mehrere voneinander unabhängige Register enthalten



Register: Ermöglicht ebenfalls die Erzeugung von Formularen mit mehreren Registern, wobei jedoch das Layout aller Register identisch ist



Bildlaufleiste: Vertikale oder horizontale Bildlaufleiste, mit der der Inhalt eines Fensters gerollt werden kann



Drehfeld: In entgegengesetzter Richtung gesetzte Pfeile, mit denen der Inhalt eines Bearbeitungsfelds mit numerischem Inhalt durchblättert werden kann, um darin beispielsweise der Reihe nach die Werte 1, 2, 3 etc. anzuzeigen





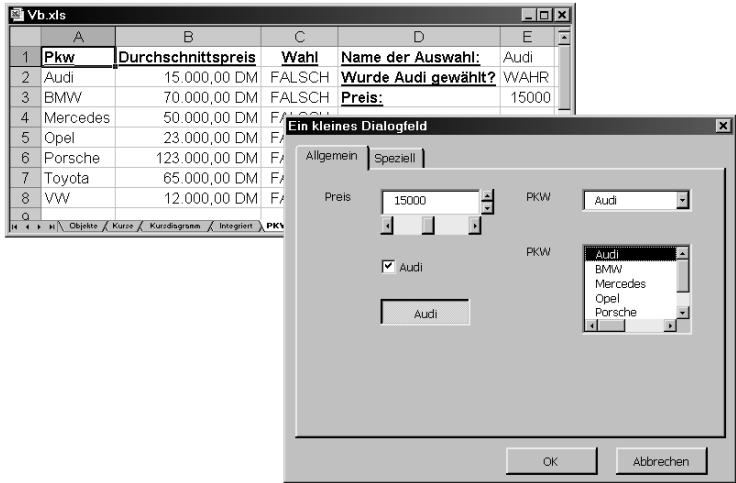
Anzeige: Fügt einen Rahmen ein, in dem eine Bilddatei angezeigt wird, beispielsweise ein Paint-Bild



RefEdit: Ähnelt einem Textfeld, ermöglicht jedoch durch Zeigen einen Bezug in das Feld einzufügen und das Dialogfeld dabei gegebenenfalls zu verkleinern, wenn es dabei im Weg ist

Ich erläutere die meisten dieser Steuerelemente anhand eines Dialogfelds mit dem Namen »UserForm2«, das mit der Tabelle »PKWs« von VB.XLS zusammenarbeitet (Abbildung 37.19).

Abbildung 37.19:
Vb.xls, Tabelle
»PKWs« und
Formular
»UserForm2«



Aktivieren Sie die Tabelle »PKWs«, so erscheint dieses Formular automatisch. Dafür verantwortlich ist die Ereignisprozedur *Worksheet_Activate* im Klassenmodul der Tabelle. Sie ist an das Tabellenereignis *Activate* gebunden, wird also beim Aktivieren der Tabelle ausgeführt. Die einzige darin enthaltene Anweisung

UserForm2.Show

führt daraufhin das abgebildete Formular »UserForm2« aus.

37.7.1 Mehrseitige Dialogfelder



Die vorhergehende Abbildung zeigt bereits den Einfluss des Steuerelements Multiseiten, mit dem Sie Formulare erzeugen können, die mehrere Register enthalten.

Dazu fügen Sie dieses Steuerelement ein, klicken auf ein Register, fügen in dieses Register die gewünschten Steuerelemente ein, klicken auf das nächste Register, fügen auch dort Steuerelemente ein und so weiter.

VBA gibt nach dem Einfügen zwar immer nur zwei Register vor, das können Sie jedoch ändern, indem Sie mit der rechten Maustaste auf eine Registerzung klicken (Abbildung 37.20).

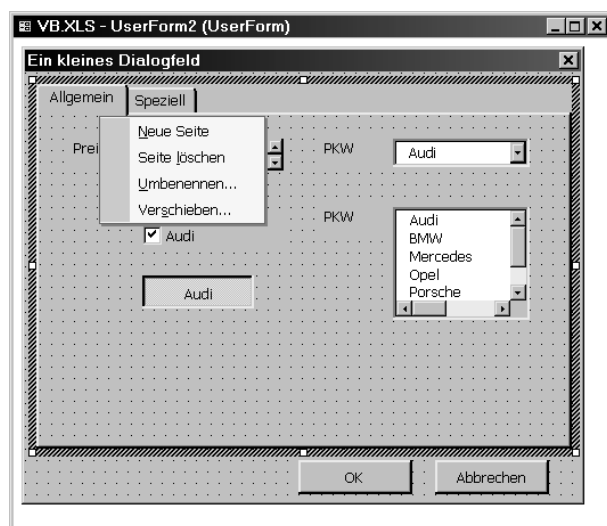


Abbildung 37.20:
Register-Kontext-
menü

Der Befehl NEUE SEITE fügt vor diesem Register ein neues ein, SEITE LÖSCHEN löscht die aktuelle Seite, UMBENENNEN... ermöglicht den in der Registerzung angezeigten Text zu editieren und mit VERSCHIEBEN... können Sie die Reihenfolge der Register verändern.

Das Steuerelement »Register« ermöglicht ebenfalls die Erzeugung von Formularen mit mehreren Registern, wobei jedoch das Layout aller Register identisch ist und im Gegensatz zum Multiseiten-Steuerelement nicht unabhängig voneinander gestaltet werden kann.



37.7.2 Bezeichnungs-, Text- und RefEdit-Felder

Bezeichnungsfelder sind recht langweilig. Sie enthalten statischen, das heißt unveränderlichen Text, und werden normalerweise nur als Zusatz zu Textfeldern verwendet.



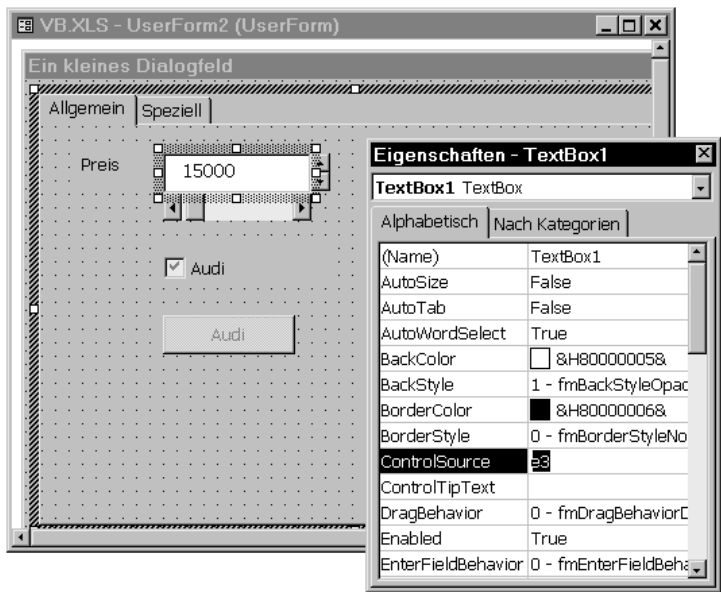
Das zuvor abgebildete Dialogfeld enthält genau ein Textfeld, neben dem sich ein mit »Preis« beschriftetes Bezeichnungsfeld befindet.



Die wichtigste Eigenschaft eines Textfelds ist *ControlSource*. Mit dieser Eigenschaft kann ein Textfeld an ein Feld einer Tabelle gebunden werden: Im Textfeld wird der Inhalt der zugehörigen Zelle angezeigt und nach Änderungen des Textfeldinhalts wird der neue Inhalt in der betreffenden Zelle gespeichert.

Im Beispiel ist das Textfeld an die Zelle E3 gebunden (Abbildung 37.21).

Abbildung 37.21:
Textfeld-
Eigenschaften



Ohne Angabe einer bestimmten Tabelle ist automatisch die während der Ausführung des Formulars aktive Tabelle gemeint, im Beispiel also die Tabelle »PKWs«.

Das Ergebnis zeigte Abbildung 37.19: Im Textfeld wird die in E3 enthaltene Zahl 15000 dargestellt. Entscheidend ist, dass sich jede Änderung des Textfeldinhalts auf E3 auswirkt: geben Sie im Textfeld 20000 ein, wird E3 sofort entsprechend aktualisiert, wenn das Textfeld den Fokus verliert, das heißt wenn ein anderes Steuerelement aktiviert oder das Formular geschlossen wird.



RefEdit-Felder ähneln Textfeldern, werden jedoch für ganz spezielle Eingaben verwendet: zur Eingabe von Zell- oder Bereichsbezügen. Ist das Feld aktiv, können Sie durch Zeigen einen Bezug in das Feld einfügen.

Sie können das Dialogfeld, in dem sich dieses Steuerelement befindet, sogar wie in Excel gewohnt verkleinert, indem Sie auf die Markierung am rechten Feldrand klicken.

37.7.3 Drehfelder und Bildlaufleisten



Am rechten Rand des Textfelds befindet sich ein Drehfeld. Dieses Steuerelement besitzt ebenfalls die Eigenschaft *ControlSource* (Abbildung 37.22).

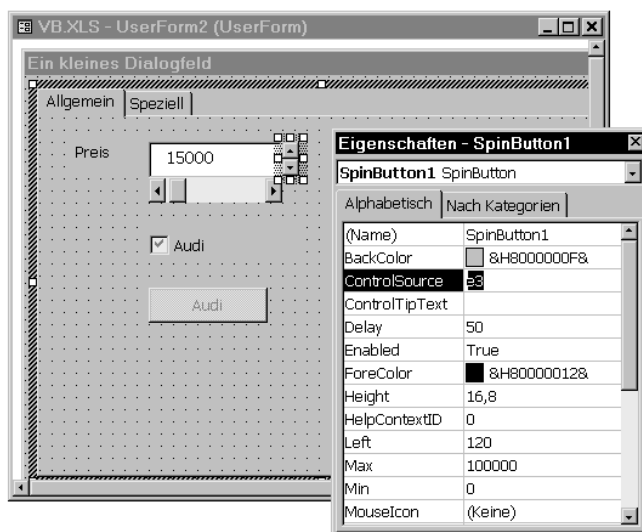


Abbildung 37.22:
Drehfeld-
Eigenschaften

Sie bestimmt erneut, an welches Tabellenfeld das Steuerelement gebunden ist. Die Bindung an Zelle E3 bewirkt, dass beim Klicken auf den aufwärts gerichteten Knopf der in E3 enthaltene Wert 15000 um eins erhöht wird.

Das tut er allerdings nur unter der Voraussetzung, dass Sie die Voreinstellung der Drehfeld-Eigenschaft Max ändern. Min und Max bestimmen die zulässigen Werte des Drehfelds und damit auch der zugehörigen Zelle!



Danach besitzen nach Klicken auf den aufwärts gerichteten Pfeil wie erläutert sowohl E3 als auch das Drehfeld den neuen Inhalt 15001. Allerdings wird dieser neue Wert nicht automatisch im Formular angezeigt, da das Drehfeld selbst keine Möglichkeit besitzt, seinen Wert anzuzeigen.

Um dafür zu sorgen, dass dieser neue Inhalt von E3 im Textfeld erscheint, das sich neben dem Drehfeld befindet, müssen Sie eine harmlose kleine Ereignisprozedur an das *Change*-Ereignis des Drehfelds binden, die nach jedem Klicken auf einen der beiden Drehpfeile aufgerufen wird (Abbildung 37.23).

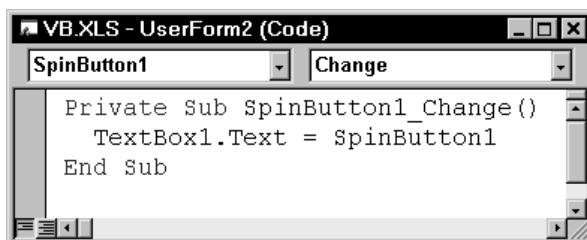


Abbildung 37.23:
Aktualisierungs-
Ereignisprozedur

Die Anweisung

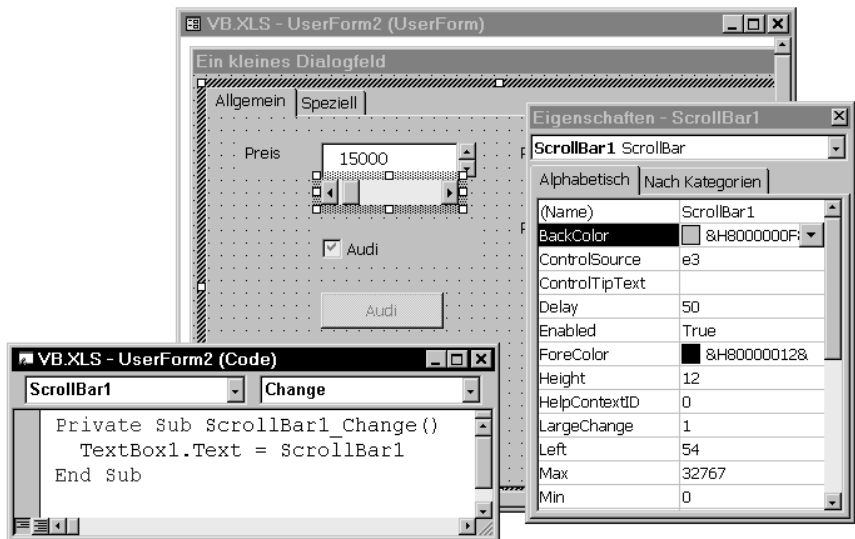
```
TextBox1.Text = SpinButton1
```

weist dem Textfeld »TextBox1« als neue Einstellung der Eigenschaft *Text*, das heißt als neuen darzustellenden Feldinhalt, den aktuellen Wert des Drehfelds »SpinButton1« zu. Nun sind beide Steuerelemente in geeigneter Art und Weise miteinander verknüpft.



Das gleiche gilt für das Bildlaufleiste-Steuerelement, das sich direkt unter dem Textfeld befindet (Abbildung 37.24).

Abbildung 37.24:
Bildlaufleiste-Eigen-
schaften und Aktua-
lisierungsprozedur



Auch in diesem Fall müssen Sie die Grenzwerteinstellungen *Min* und *Max* anpassen und eine Ereignisprozedur erstellen, die an das *Change*-Ereignis der Bildlaufleiste gebunden ist und mit der Anweisung

```
TextBox1.Text = ScrollBar1
```

nach jeder Änderung des Werts dieses Steuerelements den neuen Wert dem Textfeld »TextBox1« oder einem anderen Textfeld zuweist.

Falls Sie sich fragen, was der Unterschied zwischen Drehfeldern und Bildlaufleisten ist: nur mit Bildlaufleisten kann der Anwender den ihn interessierenden Wert (im durch *Min* und *Max* definierten Zahlenintervall) durch Ziehen des Bildlauffelds in die gewünschte Richtung in größeren Sprüngen verändern.

37.7.4 Kontrollkästchen, Umschaltfelder und Optionsgruppen

Kontrollkästchen, Umschaltfelder und Optionen sind funktional identische Steuerelemente, die als Wert nur die beiden Zustände *WAHR* oder *FALSCH* annehmen können und sich nur optisch unterscheiden.

Das Register *Allgemein* des Demoformulars enthält je ein solches Steuerelement mit der Aufschrift »Audi«. In beiden Fällen ist für *ControlSource* E2 eingestellt (Abbildung 37.25).

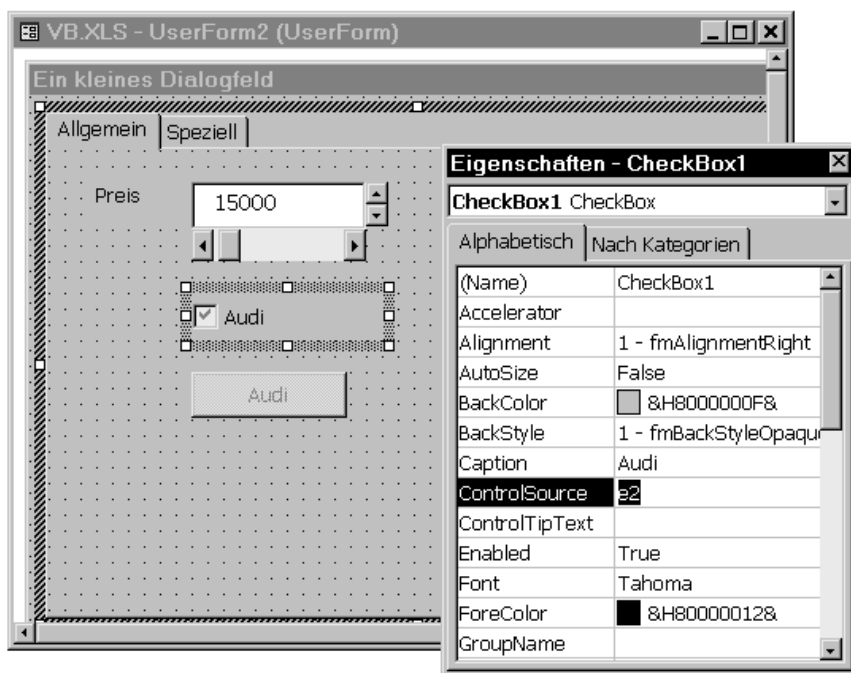


Abbildung 37.25:
Kontrollkästchen-
Eigenschaften

Dadurch wird in E2 der Wert *WAHR* gespeichert, wenn das Kontrollkästchen aktiviert ist, also ein Häkchen enthält. Analog dazu wird in E2 *WAHR* gespeichert, wenn das Umschaltfeld hervorgehoben ist.

Umgekehrt stellen beide Steuerelemente den aktuellen Inhalt von E2 dadurch dar, dass sie aktiviert oder deaktiviert sind, je nachdem, welchen der beiden Wahrheitswerte E2 gerade enthält.

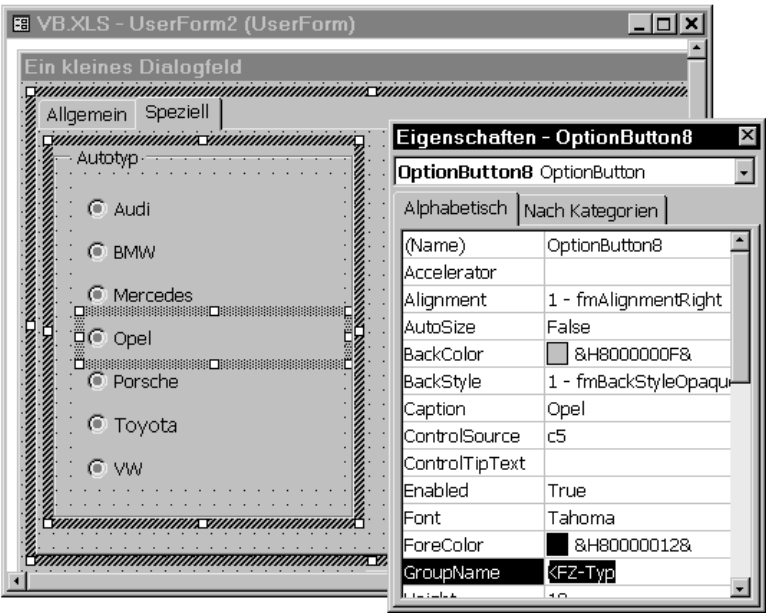
Einzelne Optionsfelder verhalten sich genauso wie Kontrollkästchen und Umschaltfelder. Optionsfelder können jedoch zu Optionsgruppen zusammengefasst werden, die wesentlich interessanter sind, da immer nur eine Option der Gruppe aktiviert sein kann; wird eine Option aktiviert, werden alle anderen somit automatisch deaktiviert.



Um mehrere Optionen zu einer solchen Gruppe zusammenzufassen, gibt es zwei Möglichkeiten: einmal können Sie zunächst mit dem abgebildeten Symbol ein Rahmen-Steuerelement einfügen und danach in die davon umschlossene Fläche die Optionen einfügen. Dadurch werden automatisch alle Optionen im Rahmen zu einer solchen Gruppe zusammengefasst.

Wollen Sie zwar ein Gruppenverhalten mehrerer Optionen erzielen, aber keinen Rahmen darum herumfügen, verwenden Sie stattdessen die Eigenschaft *GroupName* von Optionsfeldern (Abbildung 37.26).

Abbildung 37.26:
Optionsfelder-
Eigenschaften



Geben Sie für alle Optionen, die Sie zu einer Gruppe zusammenfassen wollen, den gleichen Gruppennamen ein, beispielsweise entsprechend der Abbildung »KFZ-Typ«.

Übrigens: da ich im Demoformular diese Technik anwandte und allen Optionsfelder diesen identischen Gruppennamen gab, ist der Rahmen um sie herum überflüssig und dient nur der Optik, besitzt jedoch keine funktionalen Auswirkungen.

Auch in einer Gruppe ist jedes Optionsfeld an eine Zelle gebunden und spiegelt deren Inhalt wider (*WAHR* oder *FALSCH*) bzw. verändert ihn.

Die sieben Optionsfelder des Demoformulars sind an die Zellen C2 bis C8 der Tabelle »PKWs« gebunden. Selektieren Sie Option wie »Opel«, wird somit in der zugehörigen Zelle der Wert *WAHR* gespeichert und in allen anderen *FALSCH* (Abbildung 37.27).

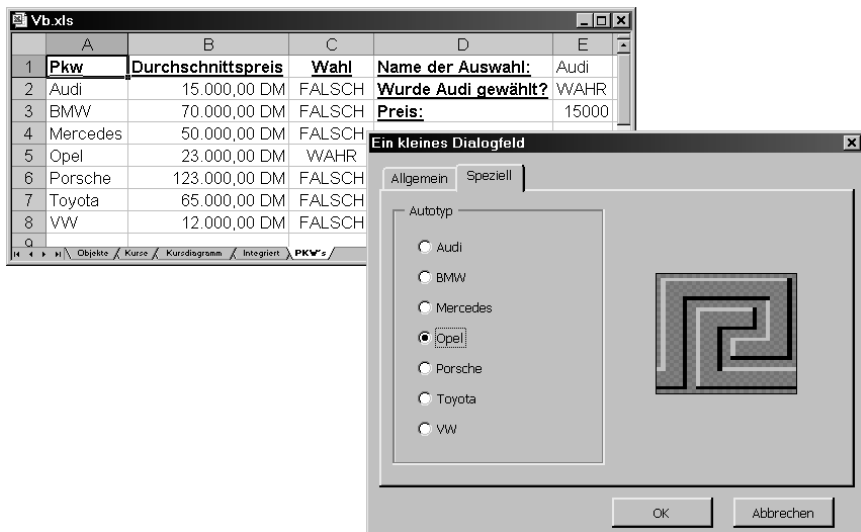


Abbildung 37.27:
Optionsgruppe

37.7.5 Anzeigefelder

Abbildung 37.27 zeigt übrigens gleichzeitig den Sinn des Anzeige-Steuerelements, nämlich im Formular ein Bild anzuzeigen. Die interessantesten Eigenschaften von Anzeige-Steuerelementen sind *Picture*, *PictureAlignment* und *PictureSizeMode* (Abbildung 37.28)

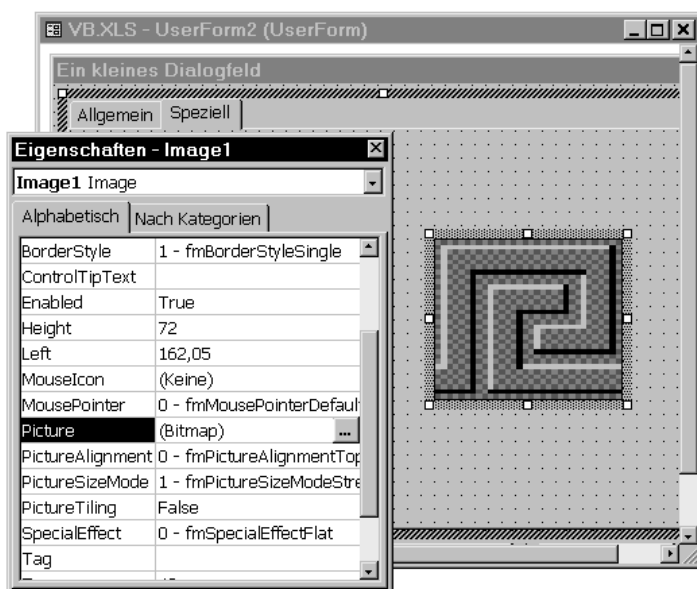


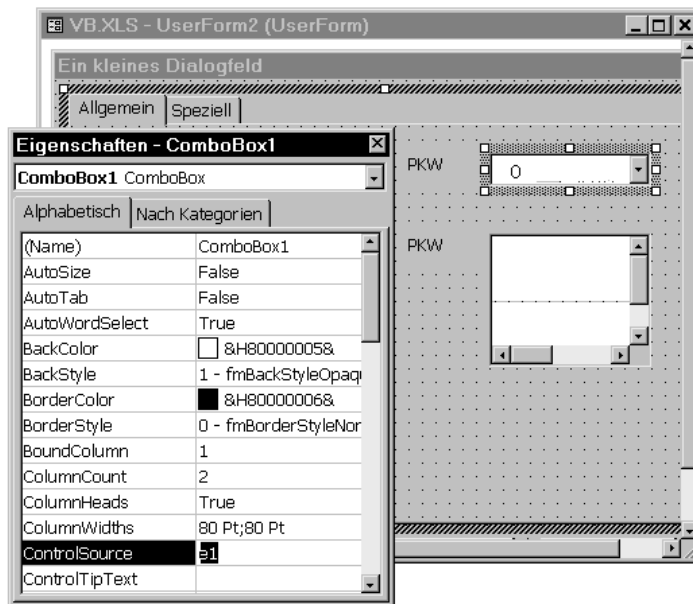
Abbildung 37.28:
Anzeigefelder-
Eigenschaften

- ➔ Ist die Eigenschaftszeile *Picture* aktiviert, befindet sich am rechten Rand eine kleine Schaltfläche. Klicken Sie darauf, wird das *Dateiauswahl*-Dialogfeld geöffnet und Sie können darin die darzustellende Grafikdatei auswählen.
- ➔ *PictureAlignment* bestimmt, wo innerhalb des Steuerelement-Rahmens die Grafik dargestellt wird: in der oberen linken Ecke, in der oberen rechten Ecke und so weiter.
- ➔ *PictureSizeMode* legt die Darstellungsgröße fest und ermöglicht das Vergrößern der Grafik auf die gesamte Fläche des Steuerelementrahmens.

37.7.6 Listen-, Dropdown- und Kombinationsfelder

Die verschiedenen Listen-Steuerelemente unterscheiden sich nur optisch. Ihre wesentlichen Eigenschaften sind jedoch identisch (Abbildung 37.29).

Abbildung 37.29:
Kombinationsfeld-
Eigenschaften



ControlSource ist die Zelle, in der das ausgewählte Listenelement gespeichert wird, im Demoformular ist das die Zelle E1.

Die in der Abbildung nicht sichtbare Eigenschaft *RowSource* bestimmt, welche Daten in der Liste angezeigt werden.

Stellen Sie diese Eigenschaft wie im Demoformular auf A2:B8 ein und geben Sie als Eigenschaft *ColumnCount*, also als Anzahl darzustellender Spalten,

entsprechend »2« ein, so werden die in diesem zweispaltigen Bereich enthaltenen Namen von Automarken bzw. die zugehörigen Preise angezeigt (Abbildung 37.30).

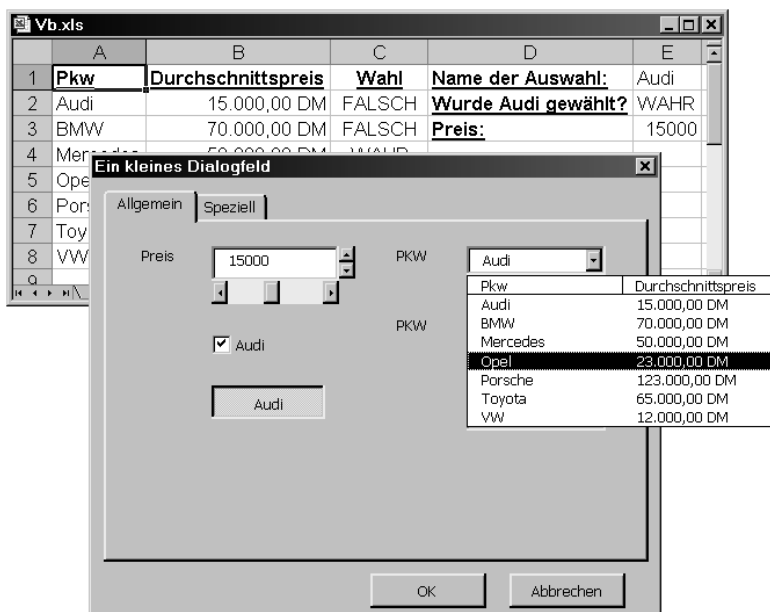


Abbildung 37.30:
Listenfeld-Daten

Die wichtigste Eigenschaft ist *BoundColumn*. Sie legt fest, welche Spalte des anzuzeigenden Bereichs die wichtigste ist.

Bedenken Sie: Wählt der Anwender aus der mehrspaltigen Liste eine Zeile aus, wählt er dadurch ja mehrere Texte oder Zahlen gleichzeitig aus. *BoundColumn* legt die Nummer der Spalte fest, deren Inhalt weiterverwendet werden soll.

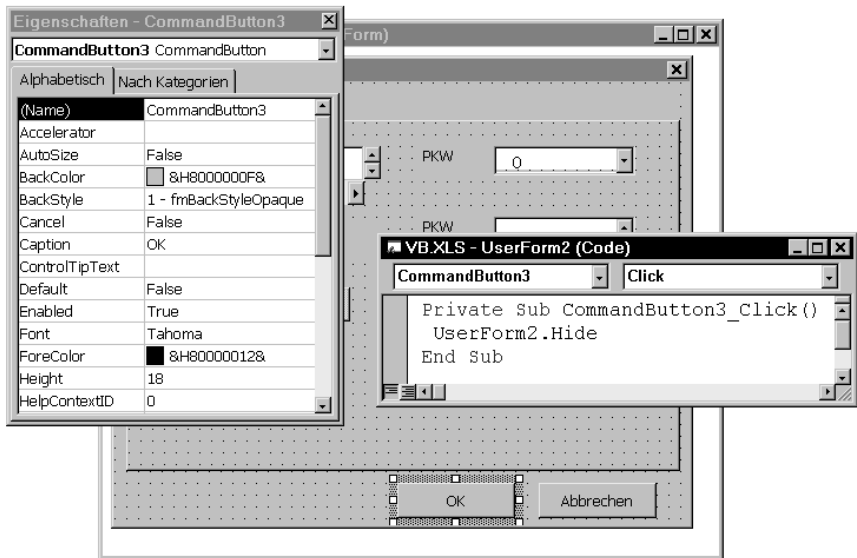
Im Beispiel ist *BoundColumn* auf »1« eingestellt, das heißt der Wert der ersten Spalte der selektierten Zeile ergibt den Wert des Listenfelds, der in Zelle E1 gespeichert wird, also der in dieser Spalte enthaltene Name einer Automarke.

37.7.7 Befehls-Schaltflächen

Normalerweise enthält ein Dialogfeld zumindest die beiden mit *OK* und *Abbrechen* beschrifteten Schaltflächen. Die Beschriftungen entscheiden jedoch nicht über die Funktionsweise einer Schaltfläche. Darüber entscheidet die Ereignisprozedur, die Sie an das *Click*-Ereignis der Schaltfläche binden (Abbildung 37.31).



Abbildung 37.31:
Befehls-Schalt-
flächen-Ereignisse



Im Beispiel enthält diese Prozedur genau eine Anweisung:

```
UserForm2.Hide
```

Die Methode *Hide* bewirkt das Gegenteil der *Show*-Methode: das Formular wird wieder ausgeblendet, praktisch geschlossen.

An die *Abbrechen*-Schaltfläche ist im Beispiel eine Ereignisprozedur gebunden, die exakt die gleiche Anweisung enthält. Klickt der Anwender auf *Abbrechen*, sollen jedoch üblicherweise alle im Dialogfeld vorgenommenen Änderungen wieder rückgängig gemacht und der Ausgangszustand wiederhergestellt werden, mit dem Resultat, dass Sie dafür oft eine weitaus komplexere Prozedur benötigen, genauer zwei Prozeduren.

Im Beispiel würde zusätzlich eine an das Formularereignis *Initialize* gebundene Prozedur benötigt, die beim Öffnen des Formulars zunächst einmal alle Werte der Tabelle »PKWs«, die mithilfe des Dialogfelds verändert werden können, in Variablen speichert.

Wird zum Schließen auf *Abbrechen* geklickt, muss die zugehörige Ereignisprozedur entsprechend diese ursprünglichen Werte wieder in den betreffenden Zellen der Tabelle speichern und damit alle eventuell inzwischen per Dialogfeld vorgenommenen Änderungen rückgängig machen.

37.8 Demo »Formulargesteuerte Kursaktualisierung«

Zum Abschluss dieses Kapitels werde ich Ihnen noch ein Beispiel mit auf den Weg geben, das zeigen soll, wozu Sie Formulare in der Praxis einsetzen können. Wie wäre es mit einem Formular, das die Kursaktualisierung in der Tabelle »Kurse« von VB.XLS erleichtert?

»Kurse« enthält zu diesem Zweck eine mit »Tageskurse hinzufügen« beschriftete Befehls-Schaltfläche. Klicken Sie darauf, so wird ein Formular ausgeführt (Abbildung 37.32).



Abbildung 37.32:
Formular zur Kursaktualisierung

Mithilfe dieses Formulars können Sie neue Tageskurse eingeben, ohne sich darum kümmern zu müssen, in welcher Zeile und Spalte die betreffenden Daten einzutragen sind, sodass es wesentlich weniger Fehlerquellen als beim direkten Eintragen in die Tabelle gibt.

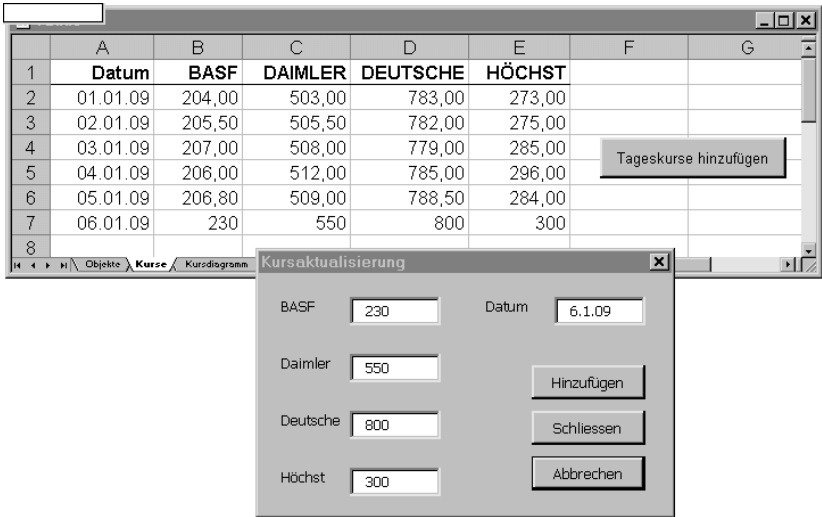
Sie geben im Textfeld »Datum« das aktuelle Datum ein und danach in den vier restlichen Textfeldern die aktuellen Kurse der vier Aktien.

Wollen Sie weitere Tageskurse hinzufügen, klicken Sie auf *Hinzufügen*. Dadurch werden die eingegebenen Kurse in die erste freie Zeile der Tabelle eingetragen, das Formular bleibt jedoch weiterhin geöffnet (Abbildung 37.33).

Sie können beliebig viele weitere Tageskurse eintragen und jeweils mit *Hinzufügen* der Tabelle hinzufügen.

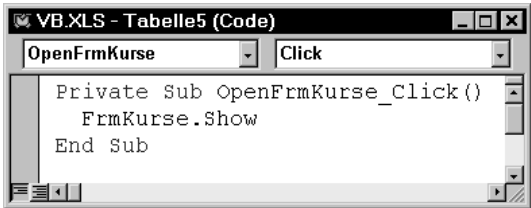
Schließen fügt ebenfalls die momentan eingegebenen Kurse hinzu, schließt das Formular jedoch anschließend. *Abbrechen* schließt das Formular dagegen, ohne die momentan darin vorhandenen Kurse einzutragen.

Abbildung 37.33:
Tageskurse
hinzufügen



Analysieren wir die Vorgänge im Einzelnen, zunächst, was beim Klicken auf die ins Diagramm integrierte Schaltfläche mit der Beschriftung »Tageskurse hinzufügen« passiert: die Schaltfläche besitzt den Namen »OpenFrmKurse« und an das Ereignis *Click* ist eine Prozedur gebunden, die beim Klicken auf die Schaltfläche aufgerufen wird (Abbildung 37.34).

Abbildung 37.34:
Vb.xls,
Klassenmodul
»Tabelle5«,
Prozedur
»OpenFrmKurse_
Click«



Die einzige darin enthaltene Anweisung *FrmKurse.Show* führt das Formular »FrmKurse« aus.

Klicken Sie darin nach der Eingabe eines Datums und der vier Kurse auf die mit *Hinzufügen* beschriftete Schaltfläche, die den Namen *CmdHinzufuegen* besitzt, so wird die zugehörige Ereignisprozedur *CmdHinzufuegen_Click* ausgeführt (Abbildung 37.35).



Diese Prozedur demonstriert zunächst eine Alternative zur in Kapitel 36.4, »Demo »Diagrammaktualisierung"«, beschriebenen Suchtechnik mit der Find-Methode: die erste leere Zelle in Spalte A wird diesmal mithilfe einer Schleife gesucht, die der Reihe nach alle Elemente der Spalte abtastet.

Wenn der Ausdruck *IsEmpty(DatumZelle)* den Wert *True* annimmt, ist die untersuchte Zelle leer und die Schleife wird mit *Exit For* vorzeitig verlassen.

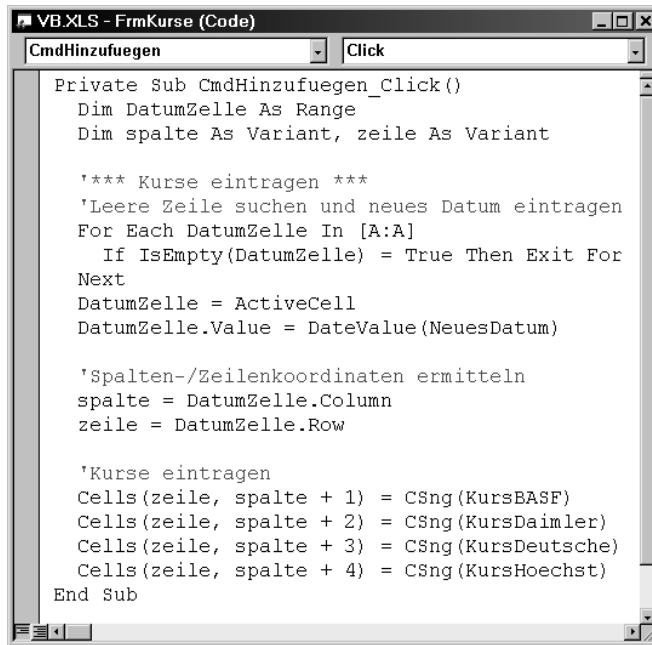


Abbildung 37.35:

Vb.xls, Klassenmodul »FrmKurse«, Prozedur »CmdHinzufuegen_Click«

Anschließend wird mit *DatumZelle = ActiveCell* der Range-Variablen *DatumZelle* ein Bezug auf die zuletzt untersuchte (leere) Zelle zugewiesen. *DatumZelle* enthält nun einen Bezug auf jene Zelle, in der das neue Datum einzutragen ist, im Beispiel auf Zelle A7. Mit

```
DatumZelle.Value = DateValue(NeuesDatum)
```

wird dieser Zelle nun der aktuelle Inhalt des mit »Datum« beschrifteten Formulartextfelds zugewiesen, das den Namen »NeuesDatum« besitzt. Das im Textfeld enthaltene Datum wird also in A7 gespeichert.

Das im Eingabefeld des Formulars eingegebene Datum »6.1.09« ist ein Text. Der Ausdruck DateValue(NeuesDatum) wandelt diesen Text durch Anwendung der Funktion DateValue in einen Datumswert um. Statt des Textes »6.1.09« wird somit das Datum 6.1.09 in die Zelle A7 eingetragen und Excel weist der Zelle nach dem Eintragen automatisch ein Datumsformat zu (wie beim Eingeben eines Datums per Hand).



Anschließend sollen in den vier rechts benachbarten Zellen nacheinander die vier Aktienkurse gespeichert werden, die momentan in vier Formulartextfeldern mit den Namen »KursBASF«, »KursDaimler«, »KursDeutsche« und »KursHoechst« enthalten sind.

Um diese Zellen möglichst einfach referenzieren zu können, ermitteln die beiden folgenden Anweisungen zunächst die Spalten- und Zeilenkoordinaten der aktuellen Zelle (im Beispiel A7):

```
spalte = DatumZelle.Column
zeile = DatumZelle.Row
```

Angewendet auf einen Bezug ermittelt die Methode *Column* den Spaltenindex (1, 2, 3 etc.) und *Row* den Zeilenindex (1, 2, 3 etc.).

Da *DatumZelle* einen Bezug auf die Zelle enthält, in der soeben das Datum eingetragen wurde, im Beispiel auf A7, enthalten die Variablen *spalte* und *zeile* nun den Spaltenindex 1 und den Zeilenindex 7.

Diese Indizes können nun sehr bequem dazu verwendet werden, nacheinander die gewünschten benachbarten Zellen anzusprechen. Da *zeile* den Wert 7 besitzt und der Ausdruck *spalte + 1* den Wert 2, referenziert *Cells(zeile, spalte + 1)* die Zelle B7. Die Anweisung

```
Cells(zeile, spalte + 1) = CSng(KursBASF)
```

speichert somit in B7 den in eine Zahl konvertierten Inhalt des Textfelds »KursBASF«. Analog dazu speichern die Anweisungen

```
Cells(zeile, spalte + 2) = CSng (KursDaimler)
Cells(zeile, spalte + 3) = CSng (KursDeutsche)
Cells(zeile, spalte + 1) = CSng (KursHoechst)
```

in C7, D7 und E7 die Inhalte der drei restlichen Textfelder.



Da der Inhalt eines Textfelds nun mal ein Text ist, keine Zahl, werden die einzutragenden vier Texte (analog zum vorhergehenden Eintragen des Datums) mit der CSngl-Funktion in die zugehörigen numerischen Werte konvertiert und aus dem Text »230« wird beispielsweise der numerische Wert 230. In den vier Zellen werden daher echte Zahlen gespeichert, keine Texte, und Excel weist den Zellen automatisch ein entsprechendes Zahlenformat zu.



Anschließend können Sie übrigens noch mit nur zwei weiteren Anweisungen das im Diagrammblatt »Kursdiagramm« enthaltene Diagramm aktualisieren, also um die neuen Kurse ergänzen (siehe Kapitel 36.4, »Demo ›Diagrammaktualisierung«).

Zunächst benötigen Sie die Anweisung

```
Range(Cells(zeile, spalte), Cells(zeile, spalte + 4)).Copy.
```

Der Ausdruck *Cells(zeile, spalte)* ist wie erläutert ein Bezug auf A7 und *Cells(zeile, spalte + 4)* ein Bezug auf E7. Der Ausdruck *Range(Cells(zeile, spalte), Cells(zeile, spalte + 4))* entspricht somit der Bereichsangabe *Range(A7:E7)*.

Dieser Bereich, in dem sich die neuen Daten befinden, wird mit der *Copy*-Methode in die Zwischenablage kopiert und danach mit

```
Charts("Kursdiagramm").Paste
```

als zusätzliche Daten ins Diagrammblatt eingefügt.

Soviel zur Kurs- und Diagrammaktualisierung, die mit der Formular-Schaltfläche *Hinzufügen* ausgelöst wird. Hinter den beiden anderen Schaltflächen *Abbrechen* und *Schließen* mit den Namen *CmdAbbrechen* und *CmdSchließen* verbergen sich zwei winzige und recht harmlose Prozeduren (Abbildung 37.36).

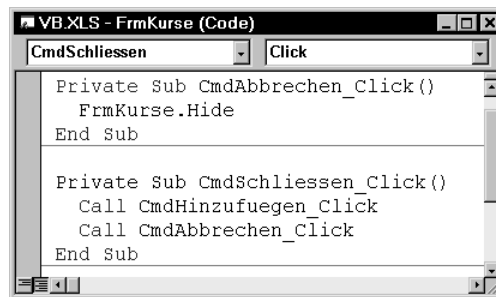


Abbildung 37.36:

Vb.xls, Klassenmodul »FrmKurse«, Prozeduren »CmdAbbrechen_Click« und »CmdSchließen_Click«

Klicken Sie auf *Abbrechen*, schließt die zugehörige Ereignisprozedur *CmdAbbrechen* das Formular mit *FrmKurse.Hide*.

Beim Klicken auf *Schließen* wird zunächst mit

```
Call CmdHinzufuegen_Click
```

die zuvor erläuterte Ereignisprozedur aufgerufen, die für das Hinzufügen der im Formular enthaltenen Daten verantwortlich ist. Diese Daten werden somit wie erläutert der Tabelle und dem Diagramm hinzugefügt. Danach wird mit

```
Call CmdAbbrechen_Click
```

zum Schließen des Formulars einfach die an die *Abbrechen*-Schaltfläche gebundene Prozedur aufgerufen, die das Formular wie erläutert mit der *Hide*-Methode ausblendet.



Wie diese Beispiele zeigen, können Ereignisprozeduren keineswegs nur durch Ereignisse aufgerufen werden, sondern auch aus anderen Prozeduren heraus mit Call.

38 Praxisführer

Der folgende Praxisführer behandelt unterschiedlichste Themen. Fast allen ist gemeinsam, dass es sich um relativ anspruchsvolle Gebiete handelt, die den Einsteiger (noch) nicht im Geringsten interessieren, sondern ausschließlich für fortgeschrittene Excel-Anwender interessant sind.

Fragen wie »Wie schütze ich meine Excel-Dokumente vor unbefugten Einblicken oder gar Veränderungen?« oder »Wie bekomme ich heraus, welche fehlerhaften Bezüge für die Fehlermeldung #NAME? verantwortlich sind?« interessieren nun mal erst nach der Erstellung einiger komplexer Tabellen.

Daher sind diese Themen am besten in diesem separaten Teil am Ende des Buches aufgehoben und werden nun Schritt für Schritt erläutert.

38.1 Menü- und Symbolleisten anpassen

Die Menü- und Symbolleisten von Excel können weitaus flexibler an individuelle Bedürfnisse angepasst werden als ich das zu Beginn dieses Buches beschrieb. Die betreffenden Schritte sind jedoch teilweise relativ komplex und daher beschreibe ich sie erst jetzt, in diesem Praxisführer.

Im Register *Optionen* des *Anpassen*-Dialogfelds können Sie das Erscheinungsbild der Symbolleisten beeinflussen (Abbildung 38.1).

Register Optionen

- ➔ Standard- und Format-Symbolleiste in zwei Zeilen anzeigen: Aktivieren Sie dieses Kontrollkästchen, werden diese beiden meistverwendeten Symbolleisten in zwei separaten Zeilen untereinander angezeigt statt in einer einzigen Zeile (Abbildung 38.2).

Diese Umschaltung können Sie jederzeit blitzschnell vornehmen, indem Sie den abwärts weisenden Pfeil am rechten Ende einer Symbolleiste anklicken und im daraufhin geöffneten Menü den entsprechenden Befehl zur Anzeige in einer bzw. zwei Zeilen wählen.

- ➔ Menüs zeigen nach dem Öffnen nur dann alle darin enthaltenen Befehle an, wenn *Menüs immer vollständig anzeigen* aktiviert ist, sonst werden zunächst nur die nach Meinung Microsofts wichtigsten Befehle gezeigt.

Abbildung 38.1:
Register »Optionen«

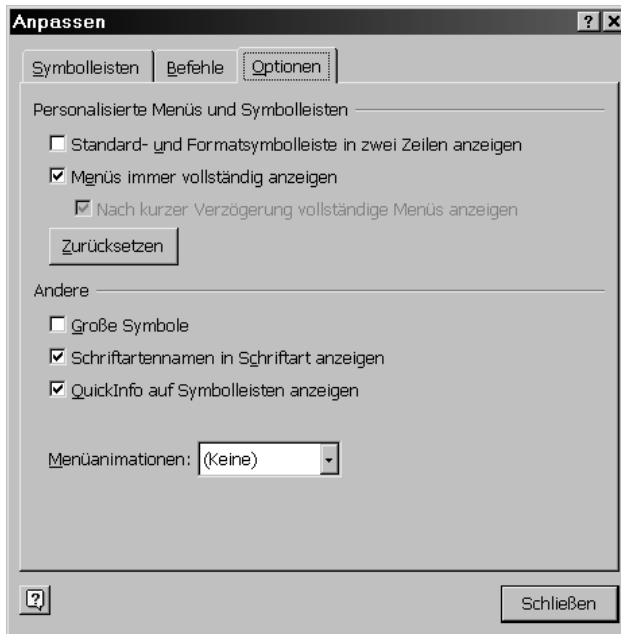
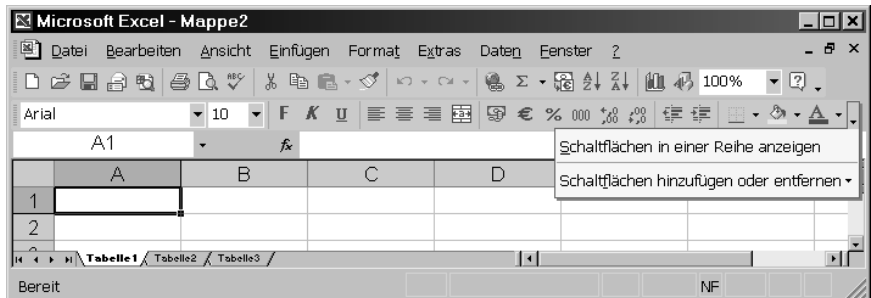


Abbildung 38.2:
Standard- und
Format-Symbol-
leiste in zwei Zeilen



- ➔ Ist *Menüs immer vollständig anzeigen* deaktiviert und *Nach kurzer Verzögerung vollständige Menüs anzeigen* aktiviert, werden zwar nach dem Öffnen zunächst nur die wichtigsten Befehle, nach kurzer Verweildauer jedoch automatisch auch die restlichen dargestellt. Ist letztere Option deaktiviert, bleibt es bei den kurzen Menüs und alle Befehle bekommen Sie nur zu sehen, wenn Sie auf den nach unten weisenden Doppelpfeil am unteren Ende eines Menüs klicken.
- ➔ *Zurücksetzen*: Auf den folgenden Seiten erläutere ich, wie Sie aus Excel-Menüs Symbole entfernen, zusätzliche Symbole einfügen oder gar eingebaute Symbole verändern. Mit dieser Schaltfläche können Sie die Veränderungen eingebauter Symbole gegebenenfalls rückgängig machen (zusätzlich eingefügte Symbole bleiben dabei erhalten).
- ➔ *Große Symbole*: Stellt die Symbole größer dar

- ➔ *Schriftartennamen in Schriftart anzeigen* bewirkt, dass in Listenfeldern zur Auswahl der Schriftart das Aussehen der einzelnen Schriften angezeigt wird und nicht nur die reinen Schriftartnamen ohne diese »Vorschau« (Abbildung 38.3).

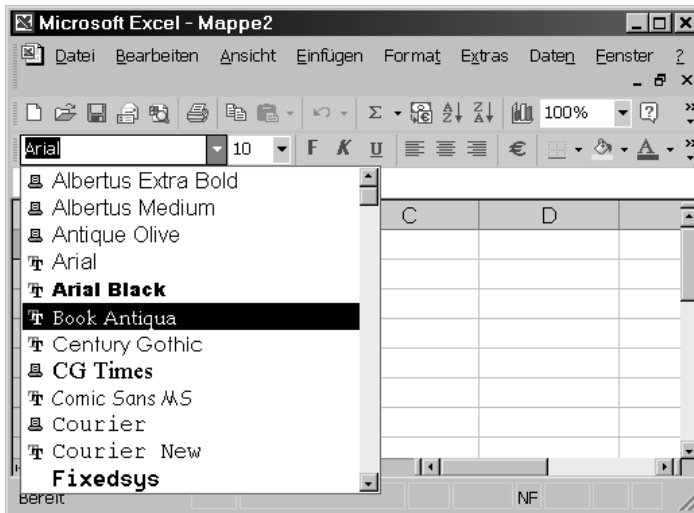


Abbildung 38.3:
Schriftartvorschau

- ➔ *QuickInfo auf Symbolleisten anzeigen* bewirkt, dass beim Verharren der Maus »auf« einem Symbol ein Fensterchen mit einer Kurzbeschreibung des Symbols erscheint.
- ➔ *Menüanimationen*: Mit den verfügbaren Alternativen können Sie festlegen, ob sich Menüs und Kontextmenüs konservativ öffnen oder mit ein wenig Pep – viel Spaß beim Spielen mit dieser Option.

Das Register *Befehle* des *Anpassen*-Dialogfelds ermöglicht die individuelle Zusammenstellung von Menü- und Symbolleisten, das Einfügen oder Löschen von Menüs, einzelnen Befehlen und Symbolen und das Verschieben oder Kopieren von Menüs, Befehlen oder Symbolen von einer Leiste in eine andere (Abbildung 38.4).

Register Befehle

Dazu wählen Sie unter *Kategorien* die Gruppe, die Sie interessiert, zum Beispiel »Diagramm erstellen«. Unter *Befehle* zeigt Ihnen Excel alle zu diesem Thema verfügbaren Symbole an. Klicken Sie eines der Symbole an und danach auf *Beschreibung*, so erscheint ein Fenster mit einer Kurzbeschreibung des Symbols.

Um es in eine Menü- oder Symbolleiste einzufügen, klicken Sie es an und ziehen es bei gedrückter Maustaste auf die betreffende Leiste (am Mauscursor befindet sich als Hinweis darauf, dass Sie an einer Stelle sind, an der das Symbol eingefügt werden kann, ein »+«-Symbol) und lassen es dort los, wo es in die Leiste eingefügt werden soll.

Abbildung 38.4:
Leisten anpassen



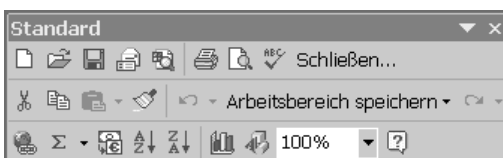
Um dieses oder ein anderes Symbol aus einer Leiste zu entfernen, ziehen Sie es bei gedrückter Maustaste heraus und lassen es einfach außerhalb der Leiste fallen. Lassen Sie es stattdessen auf eine andere Leiste fallen, wird es dort eingefügt und gleichzeitig aus der ursprünglichen Leiste entfernt. Halten Sie beim Loslassen die **[Strg]**-Taste gedrückt, so bleibt es in der ursprünglichen Leiste erhalten, wird also einfach in die neue Leiste kopiert (was durch ein »+«-Symbol am Mauscursor angezeigt wird).

Um komplette Menüs einzufügen, selektieren Sie die Kategorie »Neues Menü«. Ziehen Sie den einzigen darin enthaltenen Eintrag »Neues Menü« auf eine Symbolleiste, wird dort ein neues leeres Menü eingefügt, in das Sie anschließend wie beschrieben Symbole einfügen können.

Alternativ dazu können Sie mit den Einträgen der Kategorie »Integrierte Menüs« sogar vollständige Excel-Menüs mit allen darin enthaltenen Befehlen in die Symbolleisten einfügen.

Sie können auf diese Weise beliebige Menüs, Befehle oder Symbole aus Menü- bzw. Symbolleisten entfernen oder in andere verschieben oder kopieren. Sie können sogar mischen: Symbole in Menüleisten oder Menüs bzw. Befehle in Symbolleisten einfügen (Abbildung 38.5).

Abbildung 38.5:
Standardleiste
erweitern



Da Sie auf diese Weise ein katastrophales Durcheinander schaffen können, in dem sich niemand mehr zurechtfindet, sieht Excel eine Möglichkeit vor, den Ausgangszustand einer Menü- oder Symbolleiste wiederherzustellen. Dazu öffnen Sie zunächst im *Anpassen*-Dialogfeld das Symbolleisten-Register.

*Register
Symbolleisten*

Im Listenfeld selektieren Sie die Menü- bzw. Symbolleiste, deren Originalzustand Sie wiederherstellen wollen, und wählen *Zurücksetzen...*. Der Office-Assistent fragt, ob Sie sich auch wirklich sicher sind, Sie antworten mit OK und der Originalzustand der betreffenden Leiste ist wiederhergestellt.

Zurücksetzen

Um eine neue Leiste zu erzeugen, klicken Sie im Symbolleisten-Register auf *Neu...* und geben der zu erzeugenden Leiste im folgenden Dialogfeld einen Namen wie »Test« oder »Meine Symbole« und klicken auf OK.

*Zusätzliche
Leisten*

Auf dem Bildschirm erscheint eine winzige leere Leiste. Sie können darin wie beschrieben Symbole aus anderen Leisten per Verschieben oder Kopieren einfügen oder aber dazu wie erläutert das Befehle-Register benutzen, das alle verfügbaren Symbole enthält.

Selbstdefinierte Leisten können Sie jederzeit nachträglich umbenennen, indem Sie die betreffende Leiste im Symbolleisten-Register selektieren, *Umbenennen...* wählen und im zugehörigen Eingabefeld einen neuen Namen eingeben.

Ist in diesem Register eine selbstdefinierte Leiste selektiert, wird übrigens statt der Schaltfläche *Zurücksetzen...* die Schaltfläche *Löschen* aktiviert, mit der Sie Ihre Leiste wieder entfernen können.

Erstellen Sie eine Symbolleiste, die den Umgang mit einer ganz bestimmten Arbeitsmappe vereinfachen soll, dann gibt es Probleme, wenn Sie diese Mappe einem Kollegen weitergeben. Auf dessen Rechner ist Ihre selbstdefinierte Symbolleiste ja nicht vorhanden. Ihr Kollege besitzt somit zwar Ihre Excel-Mappe, aber nicht die zugehörige Symbolleiste, die den Umgang mit dieser Mappe vereinfachen soll!

*Symbolleiste an
Mappe anfügen*

Um das zu ändern, binden Sie die von Ihnen definierte Symbolleiste an die betreffende Mappe. Klicken Sie im *Anpassen*-Dialogfeld auf die Schaltfläche *Anfügen...*: im linken Listenfeld des zugehörigen Dialogfelds sind alle benutzerdefinierten Symbolleisten aufgeführt. Selektieren Sie eine davon und klicken Sie auf die Schaltfläche *Kopieren >>*, so wird die betreffende Leiste ins rechte Listenfeld verschoben, das die Symbolleisten enthält, die an die momentan aktive Mappe angebunden sind.

Das Resultat ist, die betreffende Leiste gehört nun zu dieser Mappe. Der Beweis ist, selbst wenn Sie die Mappe schließen und die Symbolleiste danach löschen, ist sie nach dem erneuten Öffnen der Mappe wieder vorhanden. Geben Sie nun die Mappe an Ihren Kollegen weiter, so wird somit automatisch auch die zugehörige Symbolleiste weitergegeben.

**Symbolabstand
und -breite**

Wenn das *Anpassen*-Dialogfeld geöffnet ist, können Sie sowohl integrierte als auch selbstdefinierte Symbolleisten zusätzlich mit folgenden Techniken manipulieren:

- ➔ Sie können die Abstände zwischen den Symbolen einer Leiste beeinflussen, indem Sie ein Symbol anklicken und bei gedrückter Maustaste minimal nach rechts oder links ziehen.
- ➔ Mehrere Symbole enthalten Listenfelder, beispielsweise das *Zoom*-Symbol der Standardleiste. Um die Breite dieser Listen zu verändern, klicken Sie das Symbol an, lassen wieder los und bewegen den Mauscursor zum linken oder rechten Symbolrand, bis er zu einem Doppelpfeil wird. Dann drücken Sie die Maustaste, ziehen den Feldrand nach rechts oder links und lassen die Maustaste los, wenn die gewünschte Breite erreicht ist.



Um alle Veränderungen an Menü- oder Symbolleisten rückgängig zu machen, öffnen Sie einfach die Datei mit der Endung XLB, die sich (tief verschachtelt) im Windows-Verzeichnis befindet, wodurch automatisch der Originalzustand wiederhergestellt wird.

Feinmanipulationen

Ist das *Anpassen*-Dialogfeld geöffnet, öffnet ein Klick mit der *rechten* Maustaste auf ein Symbol oder ein Menü ein Kontextmenü, mit dem Sie dieses Element manipulieren können.

Um einen bestimmten Befehl eines Menüs zu manipulieren, öffnen Sie zunächst das Menü ganz normal mit der linken Maustaste und klicken danach mit der *rechten* Maustaste auf den Sie interessierenden Befehl (Abbildung 38.6).

In diesem Kontextmenü geht es vor allem um die Optik eines Kommando-knöpfchens, egal ob dieses ein Befehl in einem Menü oder ein Symbol einer Symbolleiste ist. Sie können *jedem* Knöpfchen einen Text *und* ein Symbolbild zuweisen! Dafür verantwortlich sind die zweite und die dritte Befehlsgruppe dieses Menüs. Die zweite Gruppe besteht aus einem einzigen Befehl:

- ➔ **NAME:** [XXX] zeigt den Text an, der dem Knopf zugeordnet ist bzw. ermöglicht Ihnen, diesen Text im Eingabefeld zu verändern (beachten Sie dabei, dass Sie einen Buchstaben durch Voranstellen von »&« zur Kurztaste machen, die die Direktanwahl des Befehls ermöglicht). Dieser Befehl kann auch auf Menüs angewendet werden, beispielsweise um aus dem Menü DATEI das Menü DATEIHANDLING zu machen.

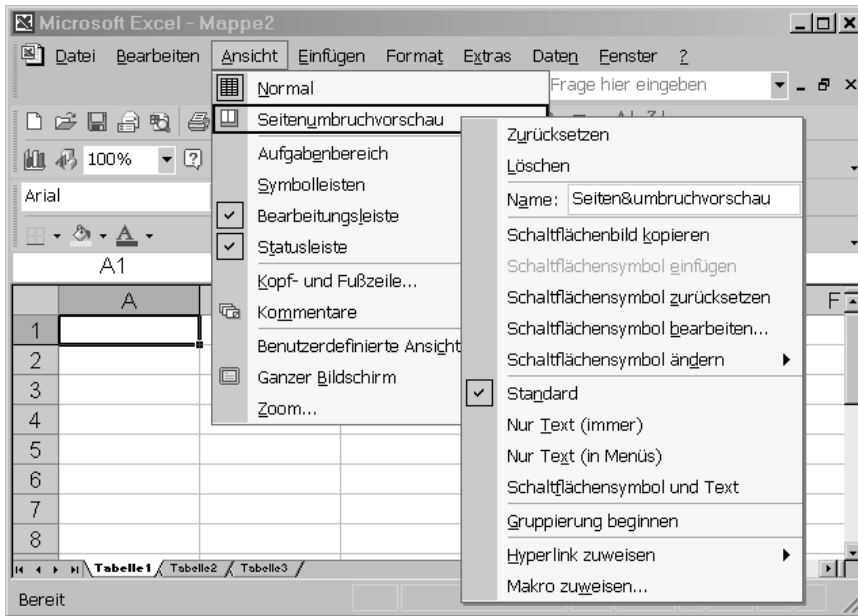


Abbildung 38.6:
Feinmanipulations-
Befehle

Die dritte Gruppe enthält Befehle zur Zuweisung eines Symbolbilds:

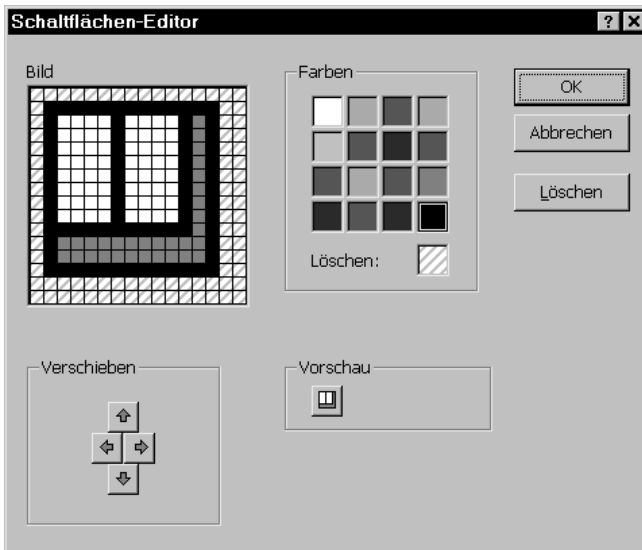
- ➔ SCHALTFLÄCHENSYMBOL ÄNDERN öffnet ein Untermenü mit mehreren Dutzend Symbolen. Klicken Sie eines davon an, wird dem Knopf das betreffende Symbolbild zugeordnet.
- ➔ SCHALTFLÄCHENSYMBOL BEARBEITEN... ermöglicht Ihnen, das Symbolbild zu bearbeiten (Abbildung 38.7).

Unter *Farben* klicken Sie die gewünschte Farbe an. Danach klicken Sie unter *Bild* eines der kleinen Rechtecke an, um es mit der betreffenden Farbe auszufüllen. Um einen solchen Farbpunkt wieder zu entfernen, klicken Sie das Feld *Löschen*: und anschließend den betreffenden Punkt an. Die Schaltfläche *Löschen* entfernt sogar alle momentan gesetzten Punkte und verschafft Ihnen eine leere Zeichenfläche.

Mit den Pfeilen unter *Verschieben* können Sie die Zeichenfläche komplett nach rechts, links, oben oder unten verschieben, wobei es jedoch nicht möglich ist, gesetzte Punkte über einen der Ränder hinaus zu verschieben.

- ➔ SCHALTFLÄCHENSYMBOL KOPIEREN überträgt das dem Symbol zu Grunde liegende Pixelmuster in die Zwischenablage. Deren Inhalt können Sie anschließend in ein Grafikprogramm einfügen und als zu bearbeitende Vorlage benutzen – oder in ein anderes Symbol übertragen, indem Sie dieses mit der rechten Maustaste anklicken und im Kontextmenü SCHALTFLÄCHE EINFÜGEN wählen.

Abbildung 38.7:
Symbolbild
bearbeiten



- ➔ **SCHALTFLÄCHENSYMBOL EINFÜGEN** verleiht somit dem Symbol das in der Zwischenablage enthaltene Pixelmuster. Sie können beispielsweise mit Paint eine Zeichnung erstellen, sie mit **BEARBEITEN|KOPIEREN** in die Zwischenablage befördern und mit diesem Befehl in ein Schaltflächen-Symbol einfügen.
- ➔ **SCHALTFLÄCHENSYMBOL ZURÜCKSETZEN** stellt den ursprünglichen Zustand des Symbolbilds wieder her.

Die vierte Befehlsgruppe bestimmt nun, wie der Knopf beschriftet wird, ob mit dem zugeordneten Text, dem Symbolbild oder mit beidem:

- ➔ **STANDARD** zeigt immer die Standard-Darstellung; der Befehl **DATEI|NEU...** wird beispielsweise als Text »Neu...« plus zugehörigem Ordnerbildchen dargestellt.
- ➔ **NUR TEXT (IMMER)** stellt nur den Text dar, ohne zugehöriges Symbolbild. Damit würde beispielsweise anstelle des weißen Blatt-Symbols in der Standard-Symboleiste nur der zugehörige Name »&Neu« angezeigt werden.
- ➔ **NUR TEXT (IN MENÜS)** zeigt bei einem Knopf wie dem Blatt-Symbol zwar weiterhin das Bild an, falls er sich in einer Symboleiste befindet. Verschieben oder kopieren Sie ihn jedoch in eine Menüleiste, wird dort nur der zugehörige Text angezeigt.
- ➔ **SCHALTFLÄCHENSYMBOL UND TEXT** zeigt sowohl den zugeordneten Text als auch das zugehörige Bildchen an, analog zur Standard-Darstellung des Befehls **DATEI|NEU...**

Letzteres ermöglicht Ihnen beispielsweise die Darstellung des Befehls DATEI|SCHLIESSEN so zu ändern, dass nicht nur wie üblich der Befehlstext »Schließen« angezeigt wird, sondern zusätzlich auch das zugehörige Symbolbild (Abbildung 38.8).

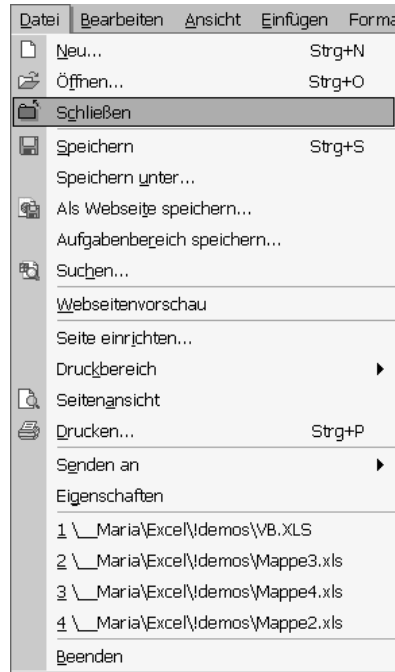


Abbildung 38.8:
Individuelle
Änderungen

Die erste Befehlsgruppe kann außer auf Befehle und Symbole auch auf Menüs angewendet werden:

- ➔ ZURÜCKSETZEN stellt den ursprünglichen Zustand des Menüs, Befehls oder Symbols wieder her.
- ➔ LÖSCHEN entfernt den Befehl, das Symbol oder gar das Menü (letzteres können Sie durch Selektion im Symbolleisten-Register und Aktivierung von *Zurücksetzen...* rückgängig machen).

Die letzten Befehle:

- ➔ Aktivieren Sie GRUPPIERUNG BEGINNEN, so erscheint vor dem betreffenden Symbol bzw. Befehl die von den Excel-Menü- bzw. Symbolleisten her bekannte dünne Trennlinie.
- ➔ MAKRO ZUWEISEN... ermöglicht die Auswahl eines der von Ihnen erstellten Makros bzw. einer VBA-Prozedur, die beim Klicken auf den Befehl bzw. das Symbol aufgerufen wird.



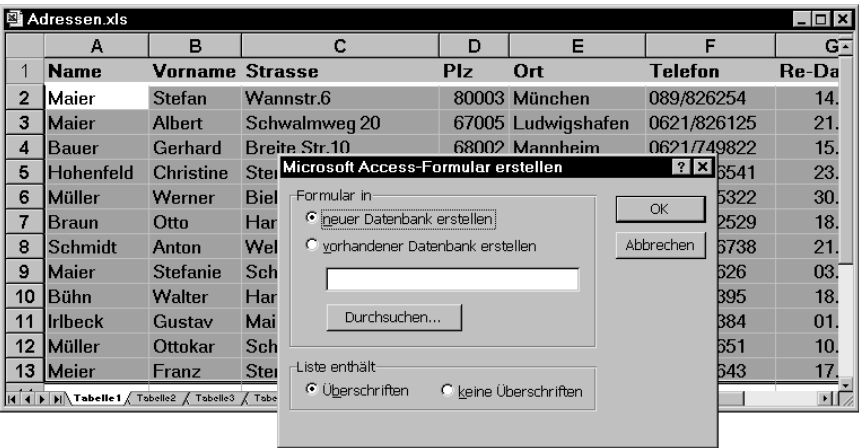
HYPERLINK ZUWEISEN... ermöglicht es, einem Symbol/Befehl einen Verweis auf eine Datei, Webseite oder E-Mail-Adresse zuzuweisen, sodass beim Anklicken des Symbols das betreffende Objekt geöffnet wird (siehe Kapitel 25, »Hyperlinks«).

38.2 Excel-Datenbanklisten mit Access verwalten

Tools im Web Excel kann mit Access zusammenarbeiten, wenn es um die Verwaltung von Listen geht. Dazu benötigen Sie jedoch das Add-In »AccessLinks«, das Sie mit EXTRAS|TOOLS IM WEB... ebenso wie manch anderes Tool von einer Microsoft-Website downloaden können.

Nach Einbindung dieses Tools können Sie beispielsweise Ihre gesamte Liste oder eine einzelne Zelle darin selektieren und danach DATEN|MS ACCESS-FORMULAR... wählen (Abbildung 38.9).

Abbildung 38.9:
Datenbank
auswählen



Excel will wissen, in welche Access-Datenbank die markierte Liste eingebunden werden soll. Sie können eine neue Access-Datenbank erstellen lassen, die einen Standardnamen erhält oder aber den Namen einer bereits vorhandenen Access-Datenbank eingeben bzw. diese nach Klicken auf *Durchsuchen...* auswählen.

OK startet Access, öffnet bzw. erstellt die angegebene Datenbank, verknüpft die Excel-Liste (im Register »Tabellen« der Access-Datenbank befindet sich anschließend eine neue Tabelle, die die Excel-Liste darstellt) und startet den Formular-Assistenten von Access.

Sind Sie mit Access und diesem Assistenten vertraut, können Sie nun ein Access-Formular gestalten. Das Formular wird – nach geraumer Zeit! – erstellt, in der Access-Datenbank gespeichert und zur Darstellung der Excel-Liste benutzt (Abbildung 38.10).

The screenshot shows an Access form titled 'Tabelle1'. It contains several text boxes for data entry, each with a label to its left. The labels and their corresponding values are: Kennnummer (empty), Name (Maier), Vorname (Stefan), Strasse (Wannstr.6), Plz (80003), Ort (München), Telefon (089/826254), Re-Datum (14.01.99), and Fehlbetrag (0). At the bottom, there is a 'Datensatz:' section with navigation buttons and a counter showing '1 von 12'.

Abbildung 38.10:
Access-Formular

Später können Sie jederzeit auf die oberhalb der Excel-Liste eingefügte Schaltfläche *Access-Formular ansehen...* klicken, um Access zu starten, die Datenbank und das Formular zu öffnen und dieses erneut zur Editierung der Excel-Liste zu benutzen.

Analog dazu können mit DATEN|MS ACCESS-BERICHT... aus Excel-Listen Access-Berichte erzeugt werden. Der Ablauf ist wie zuvor, nur dass diesmal der Berichts-Assistent von Access gestartet wird, mit dem Sie danach den gewünschten Bericht gestalten (Abbildung 38.11).

Access-Berichte

The screenshot shows an Access report titled 'Tabelle1'. It displays a list of two records. The first record has the following data: Kennnummer 1, Name Maier, Vorname Stefan, Strasse Wannstr.6, Plz 80003, Ort München, Telefon 089.826254, Re-Datum 14.01.09, and Fehlbetrag 0. The second record has: Kennnummer 2, Name Maier, Vorname Albert, Strasse Schwalmweg 20, and Plz 67005. The report includes a footer with 'Seite: 1' and navigation buttons.

Kennnummer	1
Name	Maier
Vorname	Stefan
Strasse	Wannstr.6
Plz	80003
Ort	München
Telefon	089.826254
Re-Datum	14.01.09
Fehlbetrag	0
Kennnummer	2
Name	Maier
Vorname	Albert
Strasse	Schwalmweg 20
Plz	67005

Abbildung 38.11:
Access-Bericht

Ebenfalls analog kann die Excel-Liste mit DATEN|ZU MICROSOFT ACCESS KONVERTIEREN... in eine Access-Tabelle konvertiert werden.

Access-Konvertierungen

Auch diesmal ist der Ablauf wie zuvor, allerdings wird bei Zweideutigkeiten der Import-Assistent von Access aufgerufen; anschließend teilt Ihnen Excel mit, in welcher Access-Datenbank Sie die importierte Tabelle finden (Abbildung 38.12).

Abbildung 38.12:
In Access importierte Excel-Liste



Kennnr	Name	Vorname	Straße	PLZ	Ort	Telefc
1	Maier	Stefan	Wannstr.6	80003	München	089/8262
2	Maier	Albert	Schwalmweg 20	67005	Ludwigshafen	0621/826
3	Bauer	Gerhard	Breite Str.10	68002	Mannheim	0621/749
4	Hohenfeld	Christine	Sternstr. 22	67009	Ludwigshafen	0621/736
5	Müller	Werner	Bielefelder Weg 2	10023	Berlin	0372/735
6	Braun	Otto	Hansestr. 75	40007	Düsseldorf	0125/262
7	Schmidt	Anton	Weberweg 54	60002	Heidelberg	0662/326
8	Maier	Stefanie	Schweriner Weg 18	20034	Hamburg	0326/836
9	Bühn	Walter	Hans-Pinsel-Str.2	80042	München	089/8363
10	Irlbeck	Gustav	Maiersbacherweg 21	80017	München	089/7263
11	Müller	Ottokar	Schmalweg 9	10065	Berlin	0372/936
12	Meier	Franz	Stendaler Weg 23	68003	Mannheim	0621/836

38.3 Fehler in Tabellen aufdecken



Der Befehl EXTRAS|FORMELÜBERWACHUNG|DETEKTIVSYMBOLLEISTE ANZEIGEN blendet die Detektiv-Symbolleiste ein, deren Symbole den verschiedenen Unterbefehlen von EXTRAS|FORMELÜBERWACHUNG entsprechen und die großteils in diesem Kapitel abgebildet sind.

Ich erläutere zunächst, wie Sie eine Tabelle auf Fehler hin überprüfen können und bestimmen, was genau von Excel als aufzudeckende Fehler betrachtet wird. Danach erläutere ich, welche Möglichkeiten das Fehler-Smart Tag bietet.

Oft ist die Fehlerursache eines Fehlers in einer komplexen Tabelle nicht auf Anhieb zu entdecken. Dann müssen Sie Formeln überprüfen, die Werte »verdächtiger« Formelzellen überwachen und so weiter. Dabei helfen Ihnen das Überwachungsfenster, der Formelüberwachungsmodus und die Formelauswertung.

38.3.1 Fehlerüberprüfung

Der Befehl **EXTRAS|FEHLERÜBERPRÜFUNG...** veranlasst Excel, die aktuelle Tabelle auf Fehler zu überprüfen. Dabei muss es sich keineswegs nur um »harte« Fehler wie in der folgenden Abbildung handeln, bei denen ein Fehlerwert wie **#NAME** in einer Formelzelle erscheint. Sie können Excel dazu bringen, beispielsweise auch nicht geschützte Zellen, die Formeln enthalten, oder Formeln, die auf leere Zellen verweisen, als Fehler zu betrachten (mehr dazu in den folgenden Kapiteln).

Nach der Prüfung wird Ihnen entweder mitgeteilt, dass die Tabelle keine Fehler enthält oder aber die erste fehlerhafte Zelle wird angezeigt (Abbildung 38.13).

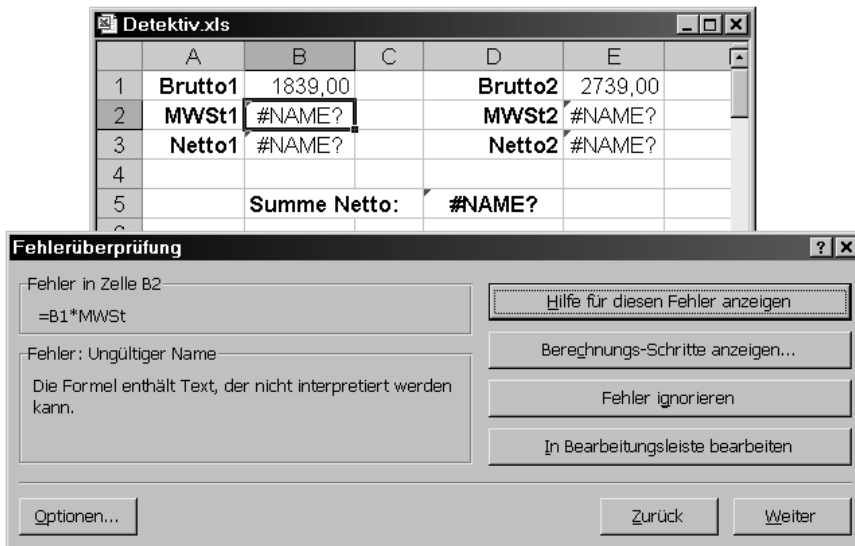


Abbildung 38.13:
Fehlerüberprüfung

Was tun, wenn eine Tabelle Fehler enthält? Damit beschäftigen sich die folgenden Unterkapitel, die zugleich die verschiedenen Optionen des Dialogfelds erläutern – abgesehen von *Zurück* und *Weiter*, die den vorhergehenden bzw. den nächsten Fehler anzeigen.

38.3.2 Fehler-Smart Tag

Zunächst einmal blendet Excel den Fehler-Smart Tag-Indikator ein, ein kleines Dreieck in den oberen linken Ecken der fehlerhaften Zellen. Klicken Sie auf eine davon, beispielsweise auf B2, und öffnen Sie danach das daraufhin erscheinende Fehler-Smart Tag, so stellt Ihnen Excel verschiedene Hilfestellungen im Kontextmenü zur Verfügung:

- ➔ *Hilfe für diesen Fehler anzeigen* entspricht der gleichnamigen Schaltfläche im Dialogfeld und gibt im Hilfefenster Hinweise zu dem Fehler.
- ➔ *Fehler ignorieren* hebt die Kennzeichnung der fehlerhaften Zelle per Dreieck in der oberen linken Ecke auf.
- ➔ *In Bearbeitungsleiste bearbeiten* hebt die Zellen (und die korrespondierenden Zellen) farblich hervor und ermöglicht das Editieren in der Eingabezeile.
- ➔ *Optionen zur Fehlerüberprüfung* öffnet das gleiche Dialogfeld wie die Schaltfläche *Optionen...* des *Fehlerüberprüfungs*-Dialogfelds (oder das Register *Fehlerüberprüfung* des Befehls EXTRAS|OPTIONEN...). Darin können Sie festlegen, ob die Fehlerüberprüfung im Hintergrund stattfinden soll und die Farbe des Fehlerindikators wählen. Vor allem aber können Sie damit verschiedene Regeln aktivieren bzw. deaktivieren, die bestimmen, was als Fehler betrachtet wird und was nicht, ob beispielsweise nicht geschützte Zellen, die Formeln enthalten, als Fehler betrachtet werden sollen.
- ➔ *Formelüberwachungs-Symbolleiste anzeigen* blendet die *Detektiv*-Symbolleiste ein.



Berechnungs-Schritte anzeigen entspricht ebenfalls der gleichnamigen Dialogfeld-Schaltfläche und öffnet ein Dialogfeld, das ich im Abschnitt »Formelauswertung« erläutere.

38.3.3 Überwachungsfenster



Der Befehl EXTRAS|FORMELÜBERWACHUNG|ÜBERWACHUNGSFENSTER ANZEIGEN aktiviert Excels einfachste Hilfestellung zum Aufdecken von Fehlern, das Überwachungsfenster, mit dem Sie den Inhalt ausgewählter Zellen überwachen können (Abbildung 38.14).

Abbildung 38.14:
Das Überwachungs-
fenster

Überwachungsfenster					
Überwachung hinzufügen...			Überwachung löschen		
Arbeitsmappe	Tabelle	Name	Ze...	Wert	Formel
DETEKTIV.XLS	Tabelle1		E1	2739,00	
DETEKTIV.XLS	Tabelle1		B3	#NAME?	=B1-B2
DETEKTIV.XLS	Tabelle2		B13	2050,00	=SUMME(B1:B12)
DETEKTIV.XLS	Tabelle2		C11	1074,00	=MITTELWERT(B13;B10)

Nach dem Aktivieren ist es zunächst leer. Um eine zu überwachende Zelle hinzuzufügen, klicken Sie mit der rechten Maustaste auf die Zelle und wählen ÜBERWACHUNG HINZUFÜGEN. Um mehrere Zellen hinzuzufügen, ist es einfacher, die betreffenden Zellen zunächst alle zu selektieren und danach mit der Schaltfläche ÜBERWACHUNG HINZUFÜGEN des Überwachungsfensters alle gleichzeitig hinzuzufügen.

Daraufhin wird im Überwachungsfenster ständig sowohl der genaue Ort der Zelle angezeigt als auch der aktuelle Wert dieser Zelle und die eventuell darin enthaltene Formel, damit Sie wissen, wie dieser Wert zustande kommt.

Interessant wird es, wenn darin wie im Beispiel eine überwachte Zelle plötzlich einen Fehlerwert wie #NAME annimmt. Dann sind Sie einem Fehler auf der Spur und können zur Aufdeckung die nachfolgend beschriebenen Techniken verwenden.

Sie können das Überwachungsfenster auch bei fehlerfreien Tabellen sinnvoll einsetzen: Scrollen Sie häufig in einer Tabelle, wollen aber ständig bestimmte Werte im Blick haben, vielleicht sogar Werte, die sich in einer anderen Tabelle befinden, so fügen Sie diese einfach ins Überwachungsfenster ein.



38.3.4 Formelüberwachungsmodus

Der mit EXTRAS|FORMELÜBERWACHUNG|FORMELÜBERWACHUNGSMODUS aktivierte Formelüberwachungsmodus verschafft Ihnen einen schnellen Überblick darüber, wo welche Formeln in Ihrer Tabelle enthalten sind (Abbildung 38.15).

	A	B	C
1			Quartal
2	Jan	173	
3	Feb	94	
4	Mrz	184	=SUMME(B2:B4)
5			
6	Apr	193	
7	Mai	152	
8	Jun	152	=SUMME(B6:B8)
9			
10	Jul	194	
11	Aug	281	
12	Sep	182	=SUMME(B10:B12)
13			
14	Okt	98	
15	Nov	163	
16	Dez	184	=SUMME(B14:B16)
17			
18	Gesamt		=C4 + C8 + C12 + C16

Abbildung 38.15:

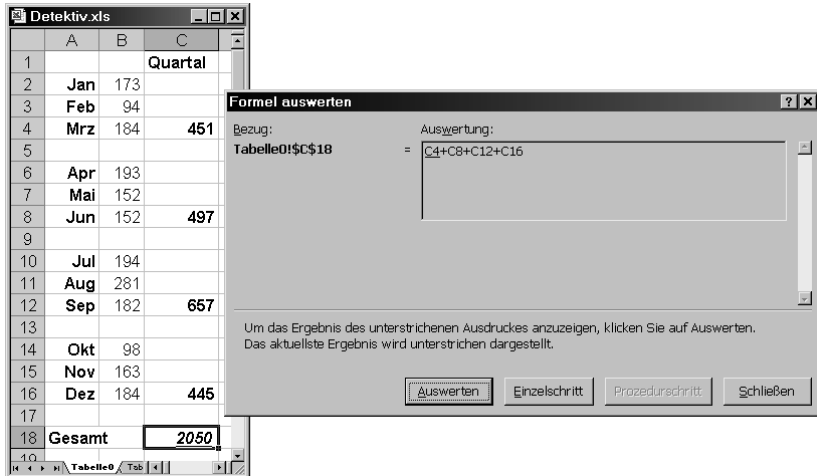
Der Formelüberwachungsmodus

Ist er aktiviert (erneute Anwahl deaktiviert ihn) werden in der Tabelle die Formeln selbst angezeigt statt der aktuellen Werte dieser Formeln.

38.3.5 Formelauswertung

Das mit EXTRAS|FORMELÜBERWACHUNG|FORMELAUSWERTUNG eingeblendete Formelauswertungsfenster hilft dabei, den aktuellen Wert einer Formel zu ermitteln und festzustellen, wie er zustande kommt. Nehmen wir als Beispiel eine Tabelle, die zwar fehlerfrei ist, an der sich die Funktionsweise dieses Fensters jedoch sehr schön demonstrieren lässt (Abbildung 38.16).

Abbildung 38.16:
Das Formel-
auswertungsfenster



Spalte B enthält zwölf Werte wie zum Beispiel monatliche Telefonausgaben, Spalte C summiert jeweils drei dieser Werte zu Quartalssummen und ermittelt ganz unten in C18 die jährliche Gesamtsumme, mit der Formel $=C4+C8+C12+C16$.



Setzen Sie den Cursor in C18 und klicken Sie das abgebildete Symbol an, wird das Formelauswertungsfenster geöffnet und zeigt die in B18 enthaltene Formel an, die Sie nun Schritt für Schritt auswerten können.

Auswerten berechnet den unterstrichenen Ausdruck, wobei es sich immer um den nächsten gemäß der Auswertungsreihenfolge (Punkt-vor-Strich-Regel etc.) auszuwertenden Ausdruck handelt, im Beispiel schlicht den ersten Ausdruck, den Inhalt von C4.

Wollen Sie genauer wissen, was dabei passiert, so klicken Sie stattdessen auf *Einzelschritt*. Excel blendet daraufhin die in der auszuwertenden Zelle C4 enthaltene Formel an, im Beispiel $=\text{Summe}(B2:B4)$. *Auswerten* wertet diesen Ausdruck aus und zeigt das Resultat dieser Summierung an, 451. *Prozedurschritt* deaktiviert diese Einblendung wieder und zeigt nur den Hauptausdruck an, $=451+C8+C12+C16$ – das heißt nun, nach erfolgtem ersten Auswertungsschritt wird stattdessen $=451+C8+C12+C16$ angezeigt, da ja soeben C4 mit dem Wert 451 ausgewertet wurde.

Nun ist der zweite Auswertungsschritt C8 unterstrichen. Sie können erneut diesen Ausdruck auswerten und das Resultat einsetzen lassen oder sich zuvor mit *Einzelschritt* die in C8 enthaltene und nun auszuwertende Formel zeigen lassen usw.

Das Auswertungsfenster ermöglicht Ihnen somit, sich die Auswertung einer Formel Schritt für Schritt vorführen zu lassen und dabei in aller Ruhe die verschiedenen Zwischenergebnisse und die daran beteiligten abhängigen Zellen zu überprüfen.

38.3.6 Formelüberwachung

Die restlichen Unterbefehle von EXTRAS|FORMELÜBERWACHUNG sind bei komplizierteren Fällen nützlich. Der besseren Verständlichkeit wegen erläutere ich sie jedoch an der folgenden recht einfachen fehlerhaften Tabelle (Abbildung 38.17).

=B1*MWSt

	A	B	C	D	E
1	Brutto1	1839,00		Brutto2	2739,00
2	MWSt1	#NAME?		MWSt2	#NAME?
3	Netto1	#NAME?		Netto2	#NAME?
4					
5		Summe Netto:		#NAME?	

Abbildung 38.17:
Fehlerhafte Tabelle

B1 und E1 enthalten je einen Bruttobetrag. Darunter soll mit der Formel $=B1 * USt$ (in B2) bzw. mit $=E1 * USt$ (in E2) die zugehörige Umsatzsteuer ermittelt werden. Darunter wiederum mit $=B1 - B2$ (in B3) bzw. $=E1 - E2$ (in E3) der resultierende Nettobetrag. In D5 wird mit $=B3 + E3$ die Summe der beiden Nettobeträge ermittelt.

Der Fehler: Ich vergaß, mit EINFÜGEN|NAME|DEFINIEREN... den in B2 und E2 benutzten Namen *USt.* zu definieren und ihm die Formel $=16\%$ zuzuordnen. Dadurch ist der Inhalt dieser beiden Zellen nicht definiert, ebenso wenig wie die Inhalte aller Zellen, die vom Inhalt dieser beiden Zellen abhängig sind.

Sollten Sie im Gegensatz zum Beispiel Schwierigkeiten haben, die Fehlerursache zu entdecken, benutzen Sie dazu die ausgefeilten Möglichkeiten Excels.

Dazu selektieren Sie irgendeine der problematischen Zellen, beispielsweise D5, und wählen EXTRAS|FORMELÜBERWACHUNG|SPUR ZUM FEHLER oder klicken einfach auf das zugehörige Symbol (Abbildung 38.18).



Abbildung 38.18:
Spur zum Fehler

	A	B	C	D	E
1	Brutto1	1839,00		Brutto2	2739,00
2	MWSt1	#NAME?		MWSt2	#NAME?
3	Netto1	#NAME?		Netto2	#NAME?
4					
5		Summe Netto:		#NAME?	
6					

D5 enthält die Formel $=B3+E3$. Excel entdeckt, dass B3 selbst einen undefinierten Inhalt besitzt und aufgrund der darin enthaltenen Formel von den Zellen B1 und B2 abhängt.

Diese beiden Spuren verfolgt Excel und erkennt, dass auch B2 einen undefinierten Inhalt enthält, B1 jedoch nicht. Die auf dem Monitor rot dargestellte Fehlerspur endet daher bei B2.

Die zusätzliche Spur von dieser Zelle zu Zelle B1 ist blau. Diese Farbe bedeutet »ist korrekt«, sodass Sie nun wissen, dass Zelle B2 der Ausgangspunkt des Fehlers ist, also die Fehler verursachende Zelle.



EXTRAS|FORMELÜBERWACHUNG|ALLE SPUREN ENTFERNEN entfernt ebenso wie das abgebildete Symbol alle momentan vorhandenen Spuren.



EXTRAS|FORMELÜBERWACHUNG|SPUR ZUM VORGÄNGER zeichnet eine Spur von der aktiven Zelle zu allen Zellen, von deren Inhalt sie abhängt, auf die sie also Bezüge enthält (Abbildung 38.19).

Abbildung 38.19:
Spur zum
Vorgänger

	A	B	C	D	E
1	Brutto1	1839,00		Brutto2	2739,00
2	MWSt1	#NAME?		MWSt2	#NAME?
3	Netto1	#NAME?		Netto2	#NAME?
4					
5		Summe Nett		#NAME?	
6					

Ist D5 aktiv, wird entsprechend in diesem Beispiel je eine Spur zu B3 und zu E3 gezeichnet, da D5 die Formel $=B3+E3$ enthält, also vom Inhalt beider Zellen abhängig ist.

Wählen Sie diesen Befehl erneut, werden die Spuren weiterverfolgt und Spuren zu jenen Zellen gelegt, von denen wiederum B3 und E3 abhängig sind (Abbildung 38.20).

	A	B	C	D	E
1	Brutto1	1839,00		Brutto2	2739,00
2	MWSt1	#NAME?		MWSt2	#NAME?
3	Netto1	#NAME?		Netto2	#NAME?
4					
5		Summe Nett		#NAME?	
6					

Abbildung 38.20:
Spur weiter-
verfolgen

Sie können die Abhängigkeiten somit über beliebig viele Ebenen weiterverfolgen, was in komplexen Tabellen zu stark verzweigten Spurbäumen führt.

Mit dem abgebildeten Symbol (kein Menü-Äquivalent) können Sie die Vorgänger-Spuren entfernen, die sich am Ende der Kette befinden (auf der tiefsten Verzweigungsebene). Erneutes Anklicken entfernt die nun letzten Vorgänger-Spuren und so weiter.



EXTRAS|FORMELÜBERWACHUNG|SPUR ZUM NACHFOLGER zeichnet genau umgekehrt eine Spur von der aktiven Zelle zu all jenen, die von ihr abhängen. Ist B1 aktiv, werden Spuren zu B2 und zu B3 gezeichnet, da beide Zellen Bezüge auf B1 enthalten (Abbildung 38.21).



	A	B	C	D	E
1	Brutto1	1839,00		Brutto2	2739,00
2	MWSt1	#NAME?		MWSt2	#NAME?
3	Netto1	#NAME?		Netto2	#NAME?
4					
5		Summe Netto:		#NAME?	
6					

Abbildung 38.21:
Spur zum
Nachfolger

Dieses Symbol (kein Menü-Äquivalent) entfernt die Nachfolger-Spuren der tiefsten Ebene, die sich am Anfang der Kette befindet. Erneutes Anklicken entfernt die nun letzten Nachfolger-Spuren und so weiter.



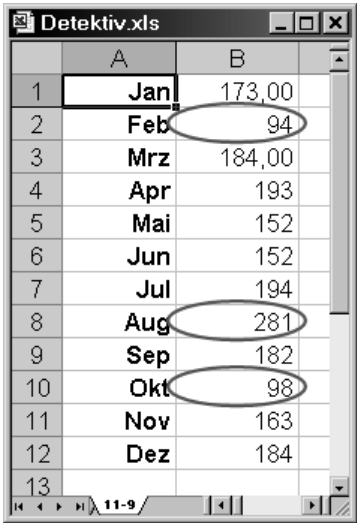
Dieses Symbol kreist alle Zellen ein (Abbildung 38.22), die Werte enthalten, die die zuvor mit DATEN|GÜLTIGKEIT... festgelegten Gültigkeitsregeln nicht einhalten (siehe Praxisführer D, »Gültigkeitsregeln benutzen«).



Dieses Symbol hebt die Zelleinkreisungen auf.



Abbildung 38.22:
Nicht eingehaltene
Gültigkeitsregeln



	A	B
1	Jan	173,00
2	Feb	94
3	Mrz	184,00
4	Apr	193
5	Mai	152
6	Jun	152
7	Jul	194
8	Aug	281
9	Sep	182
10	Okt	98
11	Nov	163
12	Dez	184
13		

38.4 Gültigkeitsregeln benutzen

DATEN|GÜLTIGKEIT... ermöglicht die Festlegung von Gültigkeitsregeln und Gültigkeitsmeldungen für Zellen, in die der Benutzer nur bestimmte Daten eingeben darf (Abbildung 38.23).

Abbildung 38.23:
Gültigkeitsregeln



	A	B
1	Jan	173,00
2	Feb	94
3	Mrz	184,00
4	Apr	193
5	Mai	152
6	Jun	152
7	Jul	194
8	Aug	281
9	Sep	182
10	Okt	98
11	Nov	163
12	Dez	184
13		

Einstellungen

Eingabemeldung

Fehlermeldung

Gültigkeitskriterien

Zulassen:

Ganze Zahl

Leere Zellen ignorieren

Daten:

zwischen

Minimum:

100

Maximum:

200

Änderungen auf alle Zellen mit den gleichen Einstellungen anwenden

Alle löschen

OK

Abbrechen

Sie markieren zuvor die Sie interessierenden Zellen oder Bereiche und legen danach im Register *Einstellungen* fest, welche Werte in diese Zellen eingegeben werden dürfen, beispielsweise entsprechend der Abbildung nur ganze Zahlen zwischen 100 und 200.

Stattdessen können Sie bei Zellen, die Texte enthalten, auch eine maximale Textlänge festlegen, bei Zellen, die ein Datum enthalten, ein Mindestdatum und so weiter.

Im Register *Eingabemeldung* können Sie eine Meldung (und eine Meldungsüberschrift, einen »Titel«) festlegen, die in Form eines Kommentars angezeigt wird (bzw. vom Office-Assistenten, wenn er aktiviert ist), wenn die betreffende Zelle markiert ist oder eine Eingabe in diese Zelle erfolgt (Abbildung 38.24).

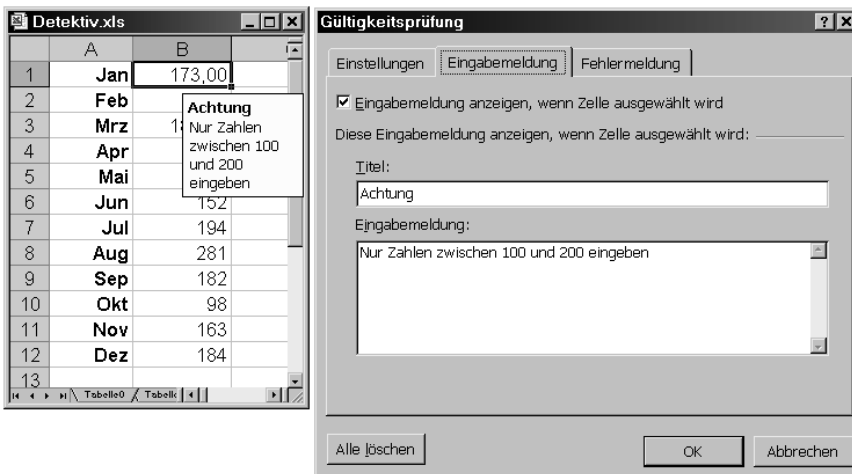


Abbildung 38.24:
Eingabemeldung

Im Register *Fehlermeldung* können Sie eine Meldung inklusive Überschrift und Hinweiszeichen festlegen. Sie erscheint, wenn in die betreffende Zelle ein Wert eingegeben wird, der die Gültigkeitsregel nicht erfüllt.

Benutzen Sie dieses Register nicht, zeigt Excel eine Standardmeldung an (»Der eingegebene Wert ist ungültig...«). In jedem Fall wird die Eingabe nicht akzeptiert und kann danach wiederholt werden.

Um eine Gültigkeitsregel zu ändern, selektieren Sie die betreffende Zelle und wählen erneut DATEN|GÜLTIGKEIT... . Aktivieren Sie im Register Einstellungen das Kontrollkästchen Änderungen auf..., so werden alle Zellen markiert, für die die gleiche Gültigkeitsregel gilt. OK weist nun all diesen Zellen die geänderte Gültigkeitsregel und die zugehörigen Meldungen zu.

Sie können die Gültigkeitsregel und die beiden Meldungen unabhängig voneinander wieder löschen, indem Sie nach der Selektion der betreffenden Zelle auf Alle löschen klicken.

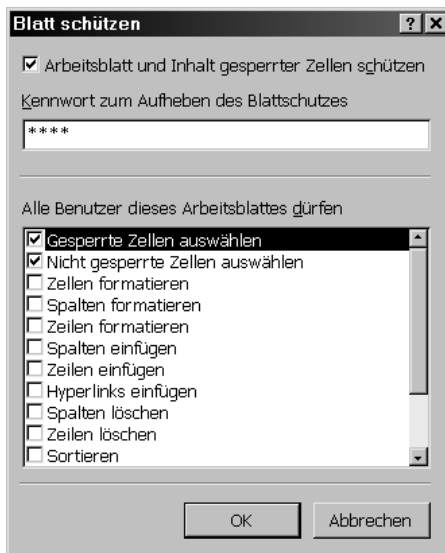


38.5 Dokumente schützen

38.5.1 Blätter schützen

Wählen Sie EXTRAS|SCHUTZ|BLATT SCHÜTZEN..., so fordert Excel Sie auf, ein Passwort einzugeben (Abbildung 38.25).

Abbildung 38.25:
Blatt schützen



Geben Sie »Test« ein (die Eingabe ist nicht sichtbar) und lassen Sie den Zustand der Kontrollkästchen unverändert. Dadurch werden sowohl die Zellinhalte der Tabelle als auch eventuell zusätzlich in ihr enthaltene Objekte wie Grafiken und auch Szenarien vor Veränderungen geschützt. Mit den Kontrollkästchen in der Liste darunter können Sie detailliert festlegen, welche Möglichkeiten Benutzer des geschützten Blattes besitzen sollen.

Nach OK müssen Sie das Passwort sicherheitshalber ein zweites Mal eingeben, um ein Vertippen auszuschließen. Abhängig von den zuvor aktivierten/deaktivierten Optionen sind anschließend alle möglichen nicht mehr zulässigen Befehle nun gar nicht erst anwählbar.

Versuchen Sie, eine nicht mehr zulässige Aktion durchzuführen, beispielsweise in einer Zelle einen Wert einzutippen, erhalten Sie eine Meldung, die darauf hinweist, dass gesperrte Zellen nicht verändert werden können (Abbildung 38.26).

Um den Schutz aufzuheben, wählen Sie EXTRAS|SCHUTZ|BLATTSCHUTZ AUFHEBEN..., der den Befehl EXTRAS|SCHUTZ|BLATT SCHÜTZEN... inzwischen ersetzt hat, und geben das Passwort ein.

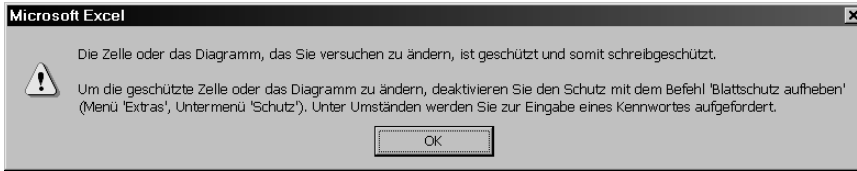


Abbildung 38.26:
Zellspernung

Die einfachste Möglichkeit, eine Tabelle ohne Aufwand vorübergehend zu schützen, besteht darin, EXTRAS|SCHUTZ|BLATT SCHÜTZEN... zu wählen und das Dialogfeld mit OK zu schließen, ohne ein Passwort zu vergeben. Anschließend genügt es, EXTRAS|SCHUTZ|BLATTSCHUTZ AUFHEBEN... anzuwählen, um den Schutz augenblicklich wieder aufzuheben.

: -)
TIPP

Der Blattschutz wirkt nur auf Zellen und Objekte, die das Attribut Gesperrt besitzen. Da zunächst alle Zellen dieses Attribut besitzen, sind nach Aktivierung des Blattschutzes entsprechend alle Zellen der Tabelle gesperrt.

f
INFO

38.5.2 Zellen entsperren

Soll ein Tabellenbereich weiterhin veränderbar bleiben, selektieren Sie zuvor den betreffenden Bereich und benutzen das Register *Schutz* des Befehls FORMAT|ZELLEN..., um die Zellen des betreffenden Bereichs zu entsperren (Abbildung 38.27).

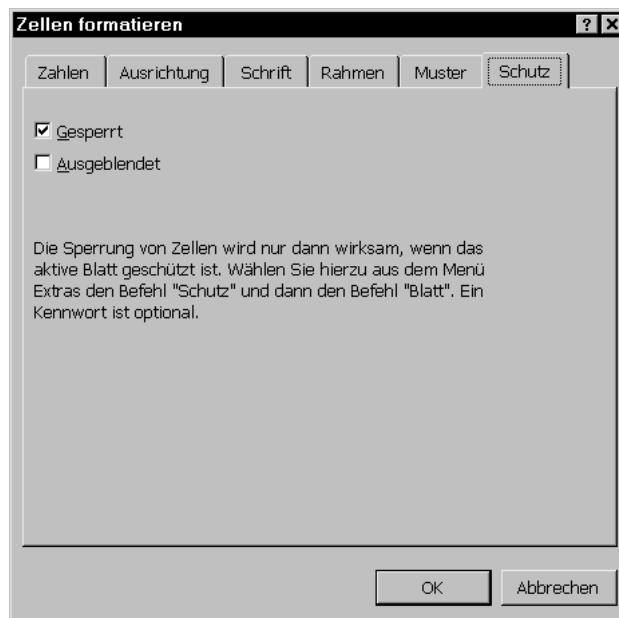


Abbildung 38.27:
Gesperrte Zellen



Gesperrt ist der Ausgangszustand. Deaktivieren Sie das Kontrollkästchen bzw. klicken Sie auf das zugehörige Symbol (Kategorie *Format* des *Anpassen*-Dialogfelds), so wird der Zellschutz für den selektierten Bericht aufgehoben und Sie können diese Zellen auch nach Aktivierung des Blattschutzes weiterhin wie gewohnt editieren.



Dieser teilweise Zellschutz ist sehr nützlich, wenn eine Anwendung fertig gestellt ist. Im Falle der Buchführung sollte man für die Zellen im Eingabebereich, in dem die zu erfassenden Werte eingegeben werden, den Zellschutz entfernen. Anschließend schützen Sie das Blatt mit einem Passwort.

Das Resultat: Im eigentlichen Eingabebereich kann der Anwender ganz normal seine Einnahmen und Ausgaben eintragen. Da alle anderen Zellen jedoch immer noch das Attribut *Gesperrt* besitzen, kann er diese nicht editieren und somit auch nicht versehentlich die Überschriften oder die Zellen mit den Summenformeln verändern oder löschen.

*Formeln
verbergen*

Zur Option *Ausgeblendet* im Register *Schutz* ist zu sagen, dass auch in geschützten Zellen darin enthaltene Zahlen und Formeln wie üblich in der Bearbeitungszeile angezeigt werden, wenn sich der Cursor auf der Zelle befindet. Genau das können Sie mit dieser Option verhindern.



Mit EXTRAS|SCHUTZ|BENUTZER DÜRFEN BEREICHE BARBEITEN... können Sie Bereiche noch gezielter zum Editieren freigeben: Klicken Sie im zugehörigen Dialogfeld auf Neu..., dann können Sie anschließend einen Bereich und ein Passwort angeben, die diesen Bereich entsperren. Sie können beliebig vielen Bereichen derartige »Editierpasswörter« zuordnen.

Schützen Sie das Blatt anschließend, kann ein Benutzer, der versucht, einen dieser Bereiche zu editieren, das nur tun, nachdem er auf die entsprechende Aufforderung hin das zugehörige Bereichspasswort eingab.

Übrigens legt das Kontrollkästchen *Benutzerberechtigungsinformation in neue Arbeitsmappe einfügen* eine neue Mappe an, die die zuvor erteilten Bereichsfreigabe-Informationen enthält.

38.5.3 Arbeitsmappen schützen

Mit EXTRAS|SCHUTZ|ARBEITSMAPPE SCHÜTZEN... können Sie eine Arbeitsmappe schützen (Abbildung 38.28).

Die Option *Struktur* verhindert, dass Blätter verschoben, kopiert oder gelöscht werden können. *Fenster* verhindert, dass die Position und Größe des Arbeitsmappenfensters verändert werden kann und sorgt dafür, dass das Fenster beim nächsten Öffnen genauso angeordnet ist wie zuvor.

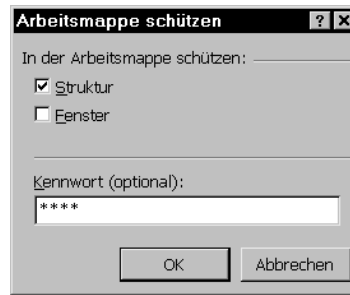


Abbildung 38.28:
Arbeitsmappe
schützen

Um den Schutz wieder aufzuheben, wählen Sie EXTRAS|SCHUTZ|ARBEITSMAPPENSCHUTZ AUFHEBEN... .

Mit EXTRAS|SCHUTZ|ARBEITSMAPPE SCHÜTZEN UND FREIGEBEN... können Sie eine Arbeitsmappe schützen, die freigegeben werden soll, und durch Aktivierung des Kontrollkästchens *Freigabe mit Änderungsprotokoll* können Sie verhindern, dass die Verfolgung von Änderungen ausgeschaltet werden kann (Abbildung 38.29).

*Freizugebende
Mappe schützen*

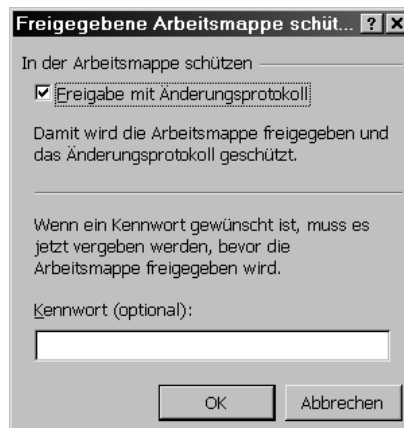


Abbildung 38.29:
Freizugebende
Mappe schützen

Um den Schutz wieder aufzuheben, wählen Sie EXTRAS|SCHUTZ|FREIGABESCHUTZ AUFHEBEN... .

38.6 Zusätzliche Add-Ins verwenden

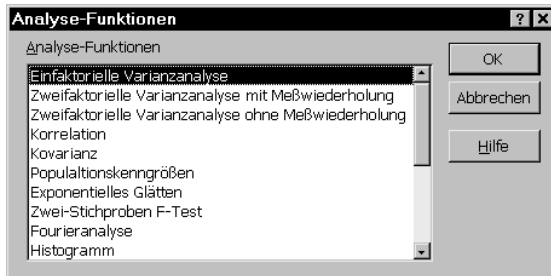
Große Add-Ins wie den Solver erläuterte ich dort, wo es thematisch angebracht war. Nachfolgend beschreibe ich nun Schritt für Schritt den Umgang mit den nützlichsten und am häufigsten verwendeten der bisher zu kurz gekommenen kleineren Add-Ins, wobei man sich zugegebenermaßen darüber streiten kann, was die nützlichsten Add-Ins sind, meine Auswahl also zwangsläufig höchst subjektiv ist.

38.6.1 Statistische Analysen

Die Analyse-Funktionen erweitern Excel um Unmengen an unterschiedlichsten statistischen Funktionen, von relativ harmlosen Stichprobentests (F-Test, Chi-Quadrat etc.) bis hin zu Funktionen, die ausschließlich für echte Statistik-Profis interessant sind.

Nach der Einbindung enthält Excel den zusätzlichen Befehl EXTRAS|ANALYSE-FUNKTIONEN... . Dahinter verbirgt sich ein Dialogfeld mit einer Liste aller Analyse-Funktionen, in der Sie sich unter einer Vielzahl statistischer Verfahren das Gewünschte aussuchen (Abbildung 38.30).

Abbildung 38.30:
Analyse-Funktionen



Alle diese Verfahren zu erläutern würde das vorliegende Excel-Buch zwangsläufig zu einem Statistikbuch zweckentfremden – und mich ein wenig überfordern.

Laien dürften darüber hinaus meist nur an einfachen deskriptiven Verfahren wie dem arithmetischen Mittel, Maxima und Minima und – Aktienspezialisten – eventuell noch an gleitenden Durchschnitten interessiert sein.

Statistik-Profis hingegen Verfahren wie F- und t-Tests, ein- und zweifaktorielle Varianzanalysen erläutern zu wollen, entspräche wohl »Eulen nach Athen tragen«. Daher beschränke ich mich darauf, nicht die angebotenen Verfahren selbst, sondern den prinzipiellen Umgang mit ihnen zu erläutern.

Als Beispiel verwende ich eine kleine Tabelle, die vier spaltenweise angeordnete Variablen mit jeweils fünf Werten enthält (Abbildung 38.31).

Welche Bedeutung diese Variablen besitzen, darüber sagt die Tabelle nichts aus. Einige Möglichkeiten:

- ➔ die Umsatzentwicklung von vier verschiedenen Produkten über fünf Jahre hinweg
- ➔ die Entwicklung von vier Aktienkursen an fünf aufeinander folgenden Tagen
- ➔ die fünf Messwerte bei vier verschiedenen physikalischen Experimenten

	A	B	C	D
1	Variable1	Variable2	Variable3	Variable4
2	13	156	12	365
3	24	190	14	200
4	15	180	10	163
5	19	187	7	124
6	22	222	8	197
7				

Abbildung 38.31:
Demo-Tabelle

Eine Möglichkeit der Auswertung wäre eine Korrelationsanalyse, um festzustellen, ob zwischen den vier Variablen ein nicht zufälliger Zusammenhang besteht. Sie wählen EXTRAS|ANALYSE-FUNKTIONEN... und selektieren *Korrelation* (Abbildung 38.32).

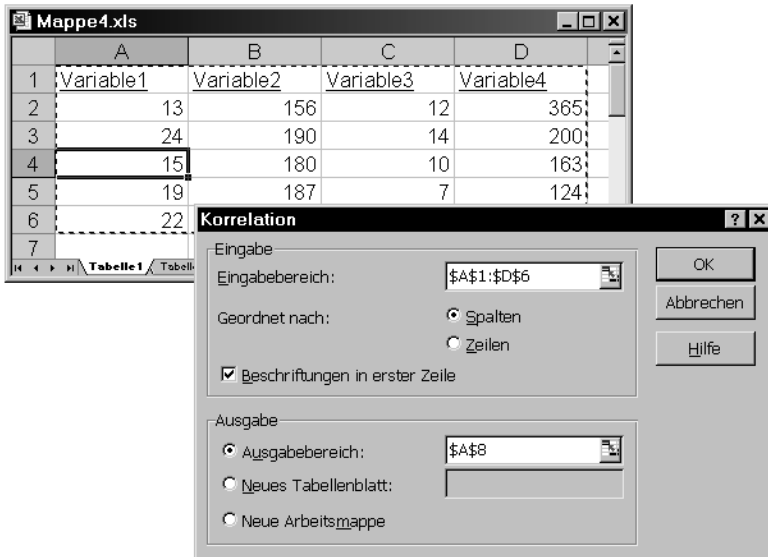


Abbildung 38.32:
Ein- und Ausgabe-
bereich definieren

Wie bei den meisten Verfahren müssen Sie einen Ein- und einen Ausgabebereich angeben.

Der Eingabebereich enthält die Daten, auf die das betreffende Verfahren angewendet wird, in unserem Fall den Bereich A1:D6, wenn man die Spaltenüberschriften mit einbezieht.

Sie können diesen Bereich bereits vor der Auswahl des Verfahrens selektieren, dann wird er in der Dialogbox vorgegeben, oder nachträglich, indem Sie das Feld *Eingabebereich* aktivieren, den Bereich nun mit der Maus oder der Tastatur selektieren oder den Bereichsbezug per Hand eingeben.

Unter *Geordnet nach* geben Sie an, ob die Daten im angegebenen Bereich zeilen- oder spaltenweise angeordnet sind. Entsprechend muss in unserem Fall wie in der Abbildung die Option *Spalten* selektiert werden.



Zusätzlich müssen Sie im Beispiel entsprechend der Abbildung das Kontrollkästchen Beschriftungen in erster Zeile aktivieren, um Excel anzuzeigen, dass die erste Zeile Überschriften enthält und keine Daten. Sonst wird Excel diese Texte als Daten mit dem numerischen Wert 0 behandeln und entsprechend verfälscht auswerten.

Aktivieren Sie die Option *Ausgabebereich* und geben Sie im zugehörigen Eingabefeld – am besten ebenfalls interaktiv durch Zeigen mit Maus oder Tastatur – die obere linke Ecke des Bereichs an, der die Auswertung aufnehmen soll, zum Beispiel A8.

Das Resultat nach Aktivierung von OK (Abbildung 38.33) ist das folgende:

Abbildung 38.33:
Korrelationsanalyse

The screenshot shows an Excel spreadsheet titled 'Mappe2.xls'. It contains two tables. The first table, 'Tabelle1', has 5 columns (A-E) and 7 rows (1-7). The first row contains headers 'Variable1' through 'Variable4'. The following rows contain numerical data. The second table, 'Tabelle2', is a 5x5 correlation matrix starting from cell A8, with headers 'Variable1' through 'Variable4' in the first row. The diagonal elements are all 1. The off-diagonal elements represent the correlation coefficients between the variables.

	A	B	C	D	E
1	Variable1	Variable2	Variable3	Variable4	
2	13	156	12	365	
3	24	190	14	200	
4	15	180	10	163	
5	19	187	7	124	
6	22	222	8	197	
7					
8		Variable1	Variable2	Variable3	Variable4
9	Variable1	1			
10	Variable2	0,763862594	1		
11	Variable3	0,045399861	-0,442316253	1	
12	Variable4	-0,48921308	-0,568807925	0,525578314	1
13					

Der resultierende Korrelationskoeffizient ist eine Zahl zwischen -1 und 1, wobei 0 für einen nicht existierenden, -1 für einen perfekten negativen (hohe Werte einer Variablen korrespondieren mit niedrigen Werten der anderen Variablen) und 1 für einen perfekten positiven (hohe Werte einer Variablen korrespondieren mit hohen Werten der anderen Variablen) Zusammenhang steht, wobei 1 in der Praxis nur bei der Korrelation einer Variablen mit sich selbst auftritt.

Der Wert 0,76 in Zelle B10 deutet auf eine recht hohe Korrelation zwischen den Variablen 1 und 2 hin.

Für die Praxis bedeutet das, dass zwischen diesen beiden Variablen – zum Beispiel der Umsatzentwicklung zweier Produkte in fünf Jahren – möglicherweise ein nicht zufälliger Zusammenhang besteht, dass also möglicher-

weise eine der beiden Variablen von der anderen oder beide gemeinsam von einer dritten Variablen abhängig sind.

Statt einen Ausgabebereich zu selektieren, der sich in der gleichen Tabelle wie der Eingabebereich befindet, können Sie mit den beiden alternativen Optionen unterhalb von *Ausgabebereich* das Ergebnis in einer neu angelegten Tabelle der aktiven Mappe beziehungsweise in einer komplett neu angelegten Arbeitsmappe speichern lassen.

Soviel zu den bei nahezu allen Verfahren notwendigen Angaben. Verfügen Sie über den entsprechenden statistischen Hintergrund, genügen Ihnen sicherlich die Hilfstexte, um sich näher über ein bestimmtes Verfahren zu informieren.

Drücken Sie einfach **[F1]**, wenn das Dialogfeld des Befehls EXTRAS|ANALYSE-FUNKTIONEN... geöffnet ist, und selektieren Sie im Hilfefenster das Sie interessierende Verfahren.

Wollen Sie die Statistik-Funktionen auch in Visual Basic-Programmen einsetzen, müssen Sie das Add-In »Analyse Funktionen – VBA« einbinden.



38.6.2 Excel-4.0-Kompatibilität

Makros, die Sie mit früheren Excel-Versionen benutzten und die wiederum Befehls- oder Funktionsmakros aufrufen, die zu Add-Ins der damaligen Excel-Version gehörten, funktionieren möglicherweise nicht mehr, weil es das aufgerufene Add-In in der aktuellen Excel-Version vielleicht nicht mehr gibt.

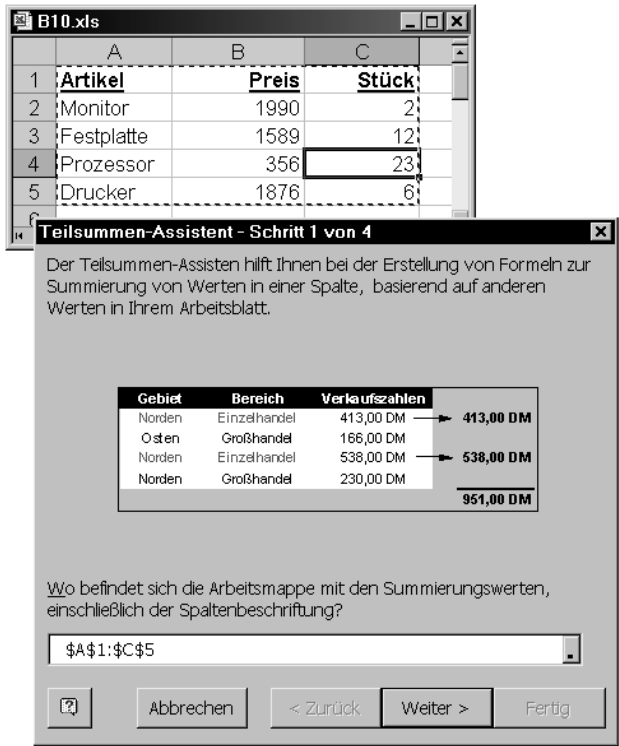
Nahezu alle fehlenden Add-Ins sind jedoch insofern weiterhin vorhanden, als sie inzwischen einfach serienmäßig in Excel enthalten sind. Damit der Aufruf von Funktionen dieser nun integrierten Add-Ins aus Ihren Makros heraus weiterhin klappt, binden Sie einfach das Add-In »Add-In-Verknüpfungen aktualisieren« ein.

Es stellt in solchen Fällen die Kompatibilität mit alten Excel-Makros her, indem es nach dem Öffnen dafür sorgt, dass von Ihren Makros aufgerufene Add-In-Funktionen zur entsprechenden nun integrierten Funktion umgeleitet werden.

38.6.3 Teilsummen

EXTRAS|TEILSUMME... ermöglicht es, nur jene Werte in einer Spalte zu summieren, für die ganz bestimmte Bedingungen zutreffen. Im ersten Schritt selektieren Sie den Sie interessierenden Bereich inklusive der Spaltenüberschriften (Abbildung 38.34).

Abbildung 38.34:
Teilsummen-
Assistent, Schritt 1



Im zweiten Schritt wählen Sie anhand dieser Spaltenüberschriften aus, welche Daten summiert werden sollen, wenn welche Bedingung erfüllt ist: Im Listenfeld *Summierungsspalte* geben Sie die Spalte an, die die zu summierenden Werte enthält, beispielsweise »Preis«. Darunter legen Sie die Summierungsbedingung fest. Dazu wählen Sie unter *Spalte* eine Spaltenüberschrift aus, daneben einen Vergleichsoperator und einen Vergleichswert.

Sie können beispielsweise festlegen, dass die in der Spalte »Stück« enthaltenen Werte nur summiert werden sollen, wenn der zugehörige Wert in der Spalte »Stück« größer als 5 ist. Dann wird nur die Anzahl jener Artikel summiert, von denen größere Mengen auf Lager sind, genauer von denen mehr als fünf Stück vorhanden sind.

Bedingung hinzufügen fügt die Bedingung in das untere Listenfeld ein. Sie können danach eine weitere Bedingung definieren und hinzufügen und so weiter. Mit *Bedingung entfernen* können Sie eine im Listenfeld selektierte Bedingung jederzeit wieder daraus entfernen.

Sie können beispielsweise mithilfe zweier Bedingungen erreichen, dass die Gesamtanzahl aller wertvollen und in größeren Mengen vorhandenen Artikel ermittelt wird, beispielsweise alle Artikel, von denen a) mehr als fünf Stück vorhanden sind und die b) teurer als 1000 DM sind.

1. Im Listenfeld *Summierungsspalte* wählen Sie als zu summierende Spalte »Stück« aus.
2. Für die erste Bedingung verwenden Sie den Parameter »Stück«, die Vergleichsbedingung »>« und den Vergleichswert »5« und fügen die Bedingung mit *Bedingung hinzufügen* ins Listenfeld ein.
3. Für die zweite Bedingung verwenden Sie den Parameter »Preis«, erneut die Vergleichsbedingung »>« und diesmal den Vergleichswert »1000« und Sie fügen die Bedingung ebenfalls mit *Bedingung hinzufügen* ins Listenfeld ein (Abbildung 38.35).

Abbildung 38.35:
Teilsommen-
Assistent, Schritt 2

Im dritten Schritt wählen Sie, ob nur die resultierende Formel in Ihrem Arbeitsblatt gespeichert werden soll oder auch die Werte der zugehörigen Bedingungen (Abbildung 38.36).

Wählen Sie letzteres, müssen Sie in den beiden folgenden Schritten jeweils eine Zelle angeben, in der die Teilsommenergebnisse gespeichert werden.

In beiden Fällen folgt danach die Angabe der Zelle, in der das Summierergebnis gespeichert werden soll (Abbildung 38.37).

Das Resultat besteht in jedem Fall darin, dass in der zuletzt gewählten Zielzelle eine Matrixformel gespeichert wird (siehe Kapitel 18, »Matrixformeln«), die das gewünschte Ergebnis liefert.

Im Beispiel 18 sind das nur der zweite Artikel, von dem zwölf Stück vorhanden sind, und der letzte Artikel, von dem es noch sechs Stück gibt, die beide Bedingungen gleichzeitig erfüllen (Abbildung 38.38).



Abbildung 38.36:
Teilsommen-
Assistent, Schritt 3

Teilsommen-Assistent - Schritt 3 von 4

Der Teilsommen-Assistent hat zwei Möglichkeiten die Formel in Ihr Arbeitsblatt zu übernehmen.

☐ Nur Formel in Zelle übernehmen.

☒ Formel und Bedingungswerte in Zelle übernehmen.

18

5	1000	18

Abbrechen

< Zurück

Weiter >

Fertig

Abbildung 38.37:
Teilsommen-
Assistent,
letzter Schritt

Teilsommen-Assistent - Schritt 6 von 6

Wählen Sie eine Zelle für den Eintrag der Summierungsformel aus..

Zelle auswählen oder eingeben und auf 'Fertig' klicken.

Tabelle1!\$C\$7

Abbrechen

< Zurück

Weiter >

Fertig

Wählten Sie wie im Beispiel die Option, auch die Werte der beiden Bedingungen zu speichern, so befinden sich in den ausgewählten Zellen die beiden Bedingungswerte 5 bzw. 1000, auf die sich die Matrixformel bezieht. Ändern Sie die in diesen Zellen enthaltenen Werte, ersetzen Sie die Preisuntergrenze 1000 beispielsweise durch 1600, wird das Ergebnis sofort entsprechend aktualisiert.

```
{=SUMME(WENN($C$2:$C$5>A7;WENN($B$2:$B$5>B7;$C$2:$C$5;0);0))}
```

Abbildung 38.38:
Ergebnis

	A	B	C	D	E
1	Artikel	Preis	Stück		
2	Monitor	1990	2		
3	Festplatte	1589	12		
4	Prozessor	356	23		
5	Drucker	1876	6		
6					
7	5	1000	18		

Mit der Option, nur die resultierende Formel im Arbeitsblatt zu speichern, müssten Sie stattdessen die Formel selbst editieren.

38.6.4 Verweise

Kommen Sie bitte nicht auf die Idee, die im folgenden Beispiel gezeigte Einkommensteuertabelle für real zu halten. Es handelt sich hier nur um ein Beispiel, dessen Zahlen Sie gegebenenfalls an die aktuell gültigen Einkommensteuertabellen anpassen müssen.

**!!
STOP**

Das mit EXTRAS|ASSISTENT|VERWEIS... aufgerufene Add-In »Verweis-Assistent« hilft bei der Erstellung von Formeln, die in Listen bestimmte Werte suchen sollen.

Zum Beispiel wäre es oft nützlich, in einer Einkommensteuertabelle anhand des zu steuernden Einkommens die fälligen Steuern zu ermitteln. Dazu geben Sie dem Assistenten im ersten Schritt den zu durchsuchenden Bereich inklusive eventueller Spalten- und Zeilenüberschriften an (Abbildung 38.39).

Im zweiten Schritt geben Sie die Spalte an, in der sich der Wert befindet, den Sie suchen. Interessiert Sie die resultierende Steuer bei einem zu steuernden Einkommen von 30000 DM, wäre das die Spalte »30000«. Interessiert Sie die gesamte anfallende Steuer, geben Sie als zugehörige Zeile »Gesamt« an (Abbildung 38.40).

Im dritten Schritt geben Sie analog zum zuvor erläuterten Teilsummen-Assistenten wieder an, ob nur die erzeugte Formel oder auch die zugehörigen Suchparameter in die Tabelle kopiert werden sollen, wobei letzteres die spätere Änderung der Parameter erleichtert (Abbildung 38.41).

Beim Kopieren auch der Suchparameter müssen Sie in den beiden folgenden Schritten jeweils eine Zelle angeben, in der diese Parameter gespeichert werden.

Abbildung 38.39:
Verweis-Assistent,
Schritt 1

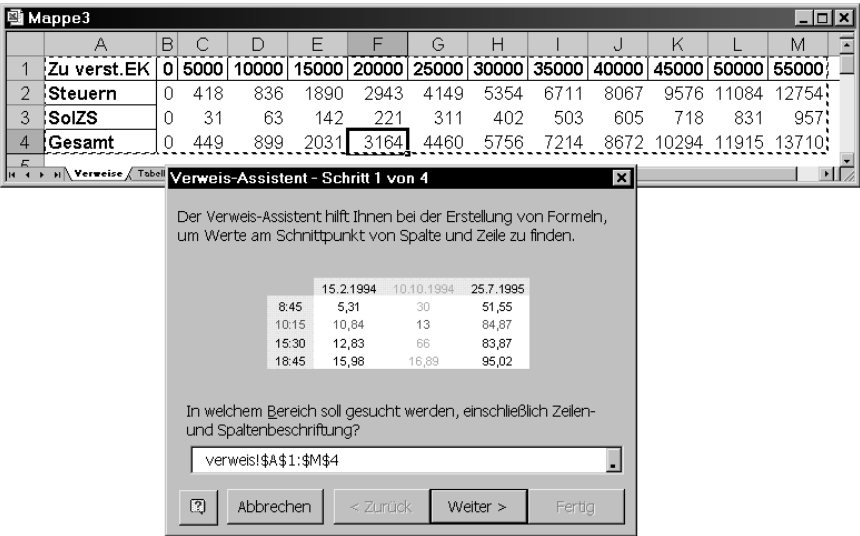
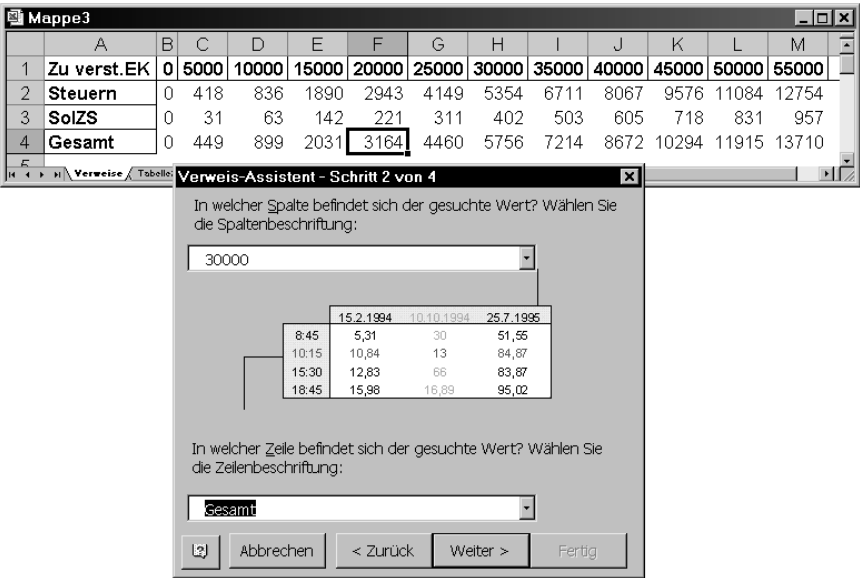
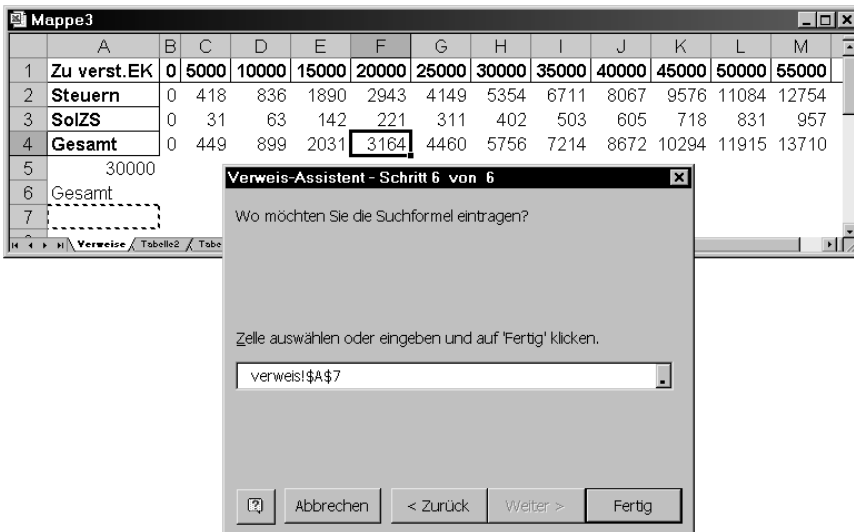
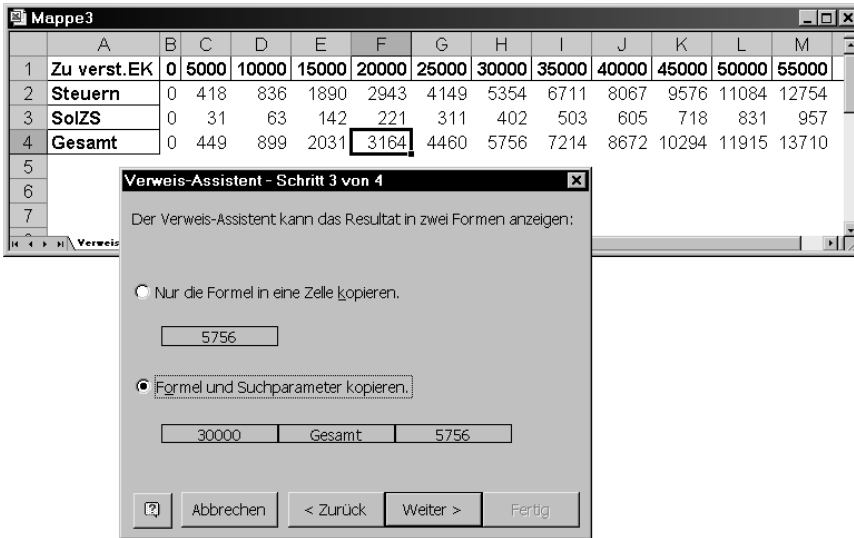


Abbildung 38.40:
Verweis-Assistent,
Schritt 2

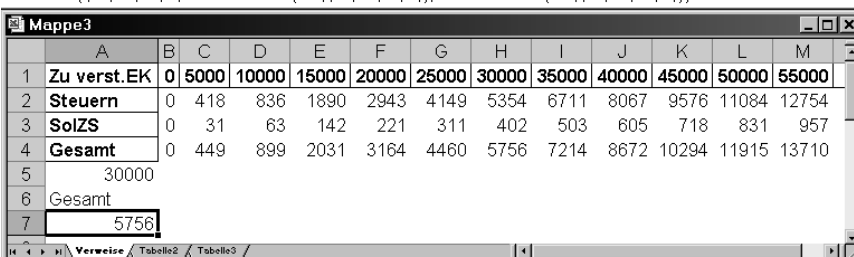


In beiden Fällen folgt danach die Angabe der Zelle, in der die erzeugte Formel gespeichert werden soll (Abbildung 38.42).

Das Resultat besteht in jedem Fall darin, dass in der zuletzt gewählten Zielzelle eine Formel gespeichert wird, die das gewünschte Ergebnis liefert. Im Beispiel ist das 5756, da dies die insgesamt fällige Steuer bei einem zu versteuernden Einkommen von 30000 DM ist (Abbildung 38.43).



=INDEX(\$A\$1:\$M\$4; VERGLEICH(A6,\$A\$1:\$A\$4;); VERGLEICH(A5,\$A\$1:\$M\$1;))



Wählen Sie wie im Beispiel die Option, auch die beiden Suchparameter zu speichern, dann genügt es, für eine Suche mit anderen Werten diese Parameter direkt in den Zellen zu ändern, in denen sie gespeichert wurden.

Ersetzen Sie beispielsweise Wert 30000 durch 40000, wird in der Formelzelle sofort die bei diesem zu versteuernden Einkommen insgesamt fällige Steuer angezeigt.

Ersetzen Sie den Text »Gesamt« durch »SolZS«, wird entsprechend nur der fällige Solidaritätszuschlag angezeigt.

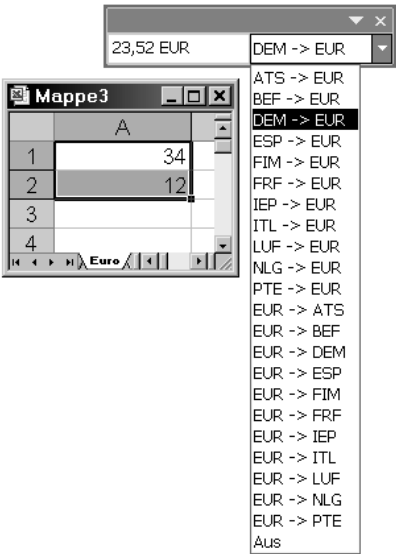
Mit der Option, nur die resultierende Formel im Arbeitsblatt zu speichern, müssten Sie stattdessen für geänderte Suchläufe die Formel selbst editieren.

38.6.5 Euro-Währungen umrechnen

Das Euro-Währungen-Add-In vereinfacht Umrechnungen einer Euro-Währung in eine andere Euro-Währung (anhand der feststehenden Wechselkurse dieser Währungen zueinander).

Nach der Einbindung dieses Add-Ins ist eine Symbolleiste verfügbar, die eine Liste aller Euro-Währungen enthält und es ermöglicht, Beträge in einer dieser Währungen in EUR zu konvertieren (»xxx -> EUR«) bzw. Euro-Beträge gemäß der feststehenden Wechselkurse genau umgekehrt in eine dieser Währungen zu konvertieren (»EUR -> xxx«). Um eine einzelne Zelle oder einen selektierten Bereich umzurechnen, klicken Sie einfach auf das entsprechende Symbol (Abbildung 38.44).

Abbildung 38.44:
Euro-Währungen



Klicken Sie beispielsweise auf »DEM -> EUR«, wird davon ausgegangen, dass die Zellen im selektierten Bereich als DM zu interpretieren sind und es wird die Summe dieser DM-Werte in den entsprechenden Euro-Betrag umgerechnet und in der Symbolleiste angezeigt.

Zusätzlich fügt dieses Add-In den Befehl EXTRAS|EUROUMRECHNUNG... ein, der ebenso wie das zugehörige Symbol ein Dialogfeld öffnet, das die einfache Umrechnung aller Werte in einem anzugebenden Bereich ermöglicht (Abbildung 38.45).

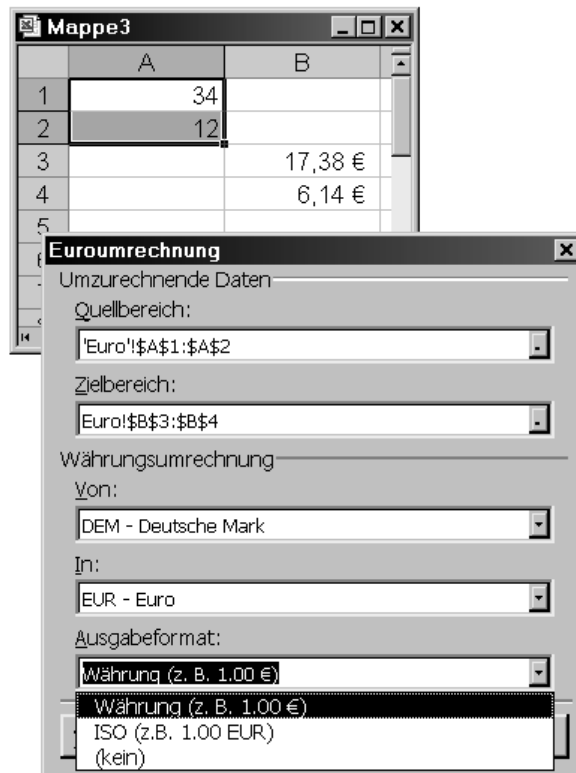


Abbildung 38.45:
Währungs-
umrechnung

Dazu sind folgende Angaben notwendig:

- ➔ **Quellbereich:** Der Bereich mit den zu konvertierenden Werten, beispielsweise A1:B3
- ➔ **Zielbereich:** Der Bereich (oder dessen obere linke Ecke), in dem die umgerechneten Werte gespeichert werden sollen (Überschreiben des Quellbereichs ist nicht zulässig)
- ➔ **Von:** Die im Quellbereich verwendete Währung, beispielsweise »DEM - Deutsche Mark«

➔ *In*: Die Währung, in die jene Werte umgerechnet werden sollen, beispielsweise »EUR – Euro« oder »FRP – Französische Franc«

➔ *Ausgabeformat*: Das im Zielbereich zu verwendende Währungsformat

OK speichert die konvertierten Werte im Zielbereich.



Die Schaltfläche Weitere öffnet ein Dialogfeld, in dem beispielsweise die Umrechnungsgenauigkeit bestimmt oder festgelegt werden kann, sodass die umgerechneten Werte im Zielbereich mit den Werten im Quellbereich verknüpft und somit bei Änderungen der Originalwerte automatisch aktualisiert werden.

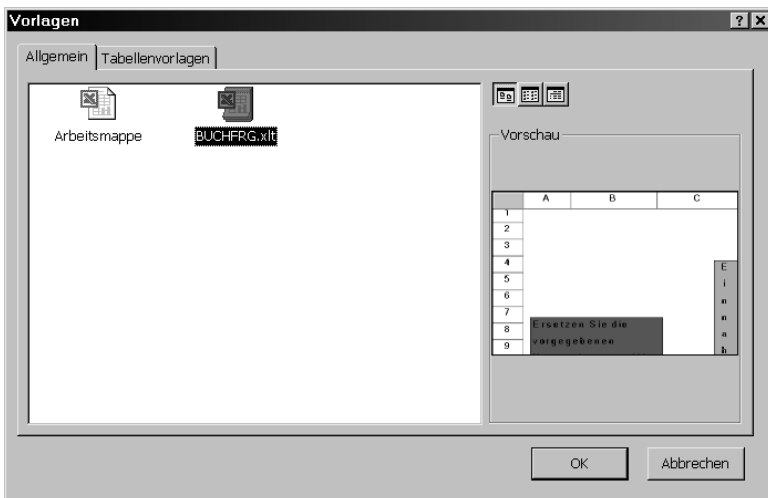
38.7 Buchführung leicht gemacht



Auf der Begleit-CD befindet sich unter anderem die Datei BUCHFRGF.XLT. Dabei handelt es sich um eine als Mustervorlage gespeicherte Excel-Mappe. Öffnen Sie diese Vorlage im Windows-Explorer, so wird nicht die Mustervorlage selbst geöffnet, sondern eine gleichartig aufgebaute Kopie der Mappe erzeugt (siehe Kapitel 8.8.2, »Mustervorlagen«).

Noch einfacher ist das Erzeugen einer solchen Kopie, wenn Sie die Mustervorlage in das Mustervorlagenverzeichnis kopieren (WINDOWS\ANWENDUNGSDATEN\MICROSOFT\VORLAGEN). Dann wählen Sie einfach DATEI|NEU... und selektieren im Register *Allgemein* des *Vorlagen*-Dialogfelds (im Aufgabenbereich *Allgemeine Vorlagen...* wählen) den Eintrag »Buchfrgr.xlt«, um eine neue Mappe zu erzeugen, die auf dieser Vorlage basiert (Abbildung 38.46).

Abbildung 38.46:
Mustervorlage
Buchfrgf.xlt
benutzen



Die neue Mappe speichern Sie anschließend unter einem Namen wie 2009.XLS und benutzen Sie während des gesamten Jahres 2009 für Ihre Buchführung. Am Jahresende erstellen Sie wieder eine neue Mappe auf Basis der Mustervorlage, speichern sie unter dem Namen 2010.XLS und benutzen diese Mappe im folgenden Jahr.

Die Mappe selbst enthält eine erweiterte und flexiblere Version der in diesem Buch entwickelten einfachen Buchführung für Selbstständige und Freiberufler, die

- ➔ ihre Einnahmen und Ausgaben überwachen müssen,
- ➔ zur monatlichen oder quartalsmäßigen Abgabe von Umsatzsteueranmeldungen verpflichtet sind
- ➔ und zur jährlichen Abgabe einer Einkommenssteuererklärung.

All diese Aufgaben vereinfacht die nun erläuterte Anwendung. Beispielsweise werden nach Erfassung der Einnahmen/Ausgaben automatisch alle Daten für die Umsatzsteueranmeldungen ermittelt und müssen nur noch von Ihnen in den entsprechenden amtlichen Vordruck übertragen werden.

Auf eines muss ich Sie jedoch hinweisen: Ich selbst verwende eine leicht variierte Version meiner Anwendung, die in langen Jahren erprobt ist, allerdings einige Bedienungseigenheiten besitzt, die ich niemandem außer mir selbst zumuten will.

Daher bastelte ich aus meiner privaten Anwendung die vorliegende allgemeinere und leichter bedienbare Version. Ob sie ebenso korrekt arbeitet und rechnet, muss sich jedoch erst noch im Dauereinsatz herausstellen. Daher kann ich keinerlei Haftung für die Korrektheit der Berechnungen übernehmen!

Verlassen Sie sich also zunächst nicht blind auf die Anwendung, sondern rechnen Sie die Ergebnisse stichprobenartig nach – und passen Sie gegebenenfalls Formeln, Texte oder was auch immer an Ihre individuellen Gegebenheiten an!

Sollte Excel das nicht zulassen, liegt es am Blattschutz, den ich benutzte, um das versehentliche Ändern speziell von Formeln zu verhindern (Abbildung 38.47).

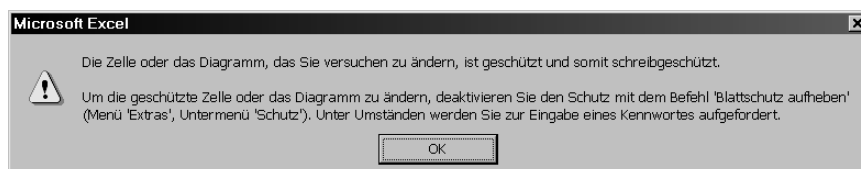


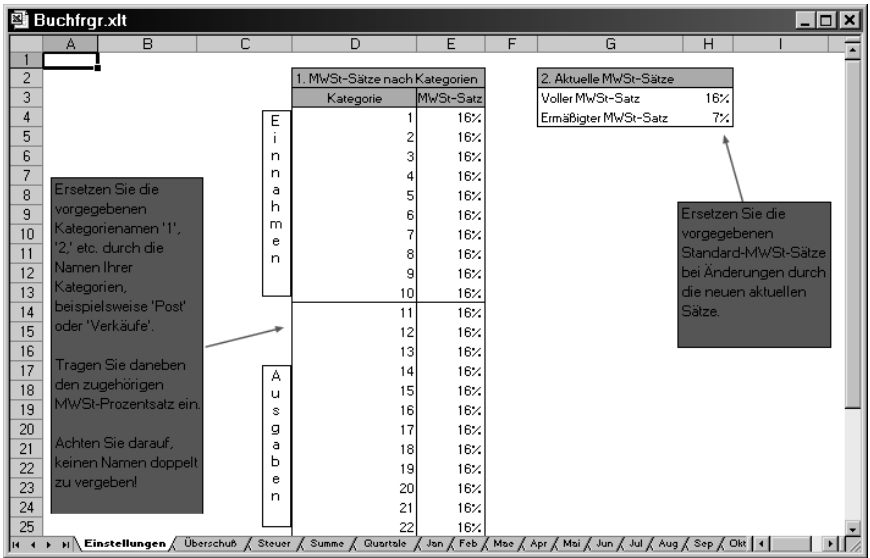
Abbildung 38.47:
Blattschutz

Um ihn aufzuheben, genügt es, EXTRAS|SCHUTZ|BLATTSCHUTZ AUFHEBEN... zu wählen, da ich kein Kennwort benutzte.

38.7.1 Anpassen der Anwendung

Bevor Sie die Buchführung einsetzen können, müssen Sie sie zunächst individuell anpassen. Am besten passen Sie die Mustervorlage selbst an, erzeugen also keine neue Mappe auf Basis dieser Vorlage, sondern öffnen mit DATEI|ÖFFNEN... die Mustervorlage BUCHFRGF.XLT (Abbildung 38.48).

Abbildung 38.48:
Inhalt der
Mustervorlage



Nahezu alle Anpassungen werden in der Tabelle »Einstellungen« vorgenommen. Prüfen Sie zunächst, ob die in H3 und H4 vorgegebenen Umsatzsteuer-Prozentsätze noch gültig sind; falls nicht, wenn diese Sätze beispielsweise zwischenzeitlich erhöht wurden, aktualisieren Sie die beiden Werte bitte entsprechend.

Wichtiger ist, dass Sie im Bereich D4:E33 Ihre Einnahmen-/Ausgabenkategorien benennen und die zugehörigen Umsatzsteuersätze eingeben (Abbildung 38.49).

Im Bereich D4:D13 tragen Sie die Bezeichnungen von bis zu zehn Einnahmekategorien wie »Softwareverkäufe« oder »Buchverkäufe« ein, daneben in E4:E13 tragen Sie die zugehörigen Umsatzsteuersätze ein. Bei Softwareverkäufen werden beispielsweise 16% Umsatzsteuer fällig, auf Buchverkäufe wird dagegen der ermäßigte Umsatzsteuersatz von 7% angewandt.

1. MwSt-Sätze nach Kategorien

Kategorie	MwSt-Satz
Softwareverkäufe	16%
Buchverkäufe	7%
3	16%
4	16%
5	16%
6	16%
7	16%
8	16%
9	16%
10	16%
Post	0%
Telefon	16%
Bücher	7%
14	16%
15	16%
16	16%
17	16%
18	16%
19	16%
20	16%
21	16%
22	16%

2. Aktuelle MwSt-Sätze

Voller MwSt-Satz	16%
Ermäßigter MwSt-Satz	7%

Einweisungen

Ersetzen Sie die vorgegebenen Kategorienamen '1', '2' etc. durch die Namen Ihrer Kategorien, beispielsweise 'Post' oder 'Verkäufe'.

Tragen Sie daneben den zugehörigen MwSt-Prozentsatz ein.

Achten Sie darauf, keinen Namen doppelt zu vergeben!

Ausgaben

Ersetzen Sie die vorgegebenen Standard-MwSt-Sätze bei Änderungen durch die neuen aktuellen Sätze.

Abbildung 38.49:

Tabelle
»Einstellungen«

Analog dazu benennen Sie im Bereich D14:D33 Ihre Ausgaben und geben daneben in E14:E33 ebenfalls die zugehörigen Umsatzsteuersätze an; zum Beispiel sind in Telefonrechnungen 16% Umsatzsteuer enthalten, Postwertzeichen dagegen sind noch umsatzsteuerfrei (0%).

In der Tabelle »Überschuss« sind ebenfalls ein paar Anpassungen erforderlich (Abbildung 38.50).

Kategorie	Betriebsanteil	Betrag
Post	100%	0,00
Telefon	100%	0,00
Bücher	100%	0,00
14	100%	0,00
15	100%	0,00
16	100%	0,00
17	100%	0,00
18	100%	0,00
19	100%	0,00
20	100%	0,00
21	100%	0,00
22	100%	0,00
23	100%	0,00
24	100%	0,00
25	100%	0,00
26	100%	0,00
27	100%	0,00
28	100%	0,00
29	100%	0,00
30	100%	0,00
Zugehörige USt		0,00

Abbildung 38.50:

Tabelle
»Überschuss«

In I5:I24 müssen Sie die Vorgaben »100%« für den Betriebsanteil der betreffenden Ausgabenkategorie anpassen.

Angenommen, Ihre Telefonkosten und somit auch die darin enthaltene Umsatzsteuer können Sie nur zu 80% betrieblich absetzen, 20% sind also Privatnutzung. Dann müssen Sie die zu Ihrer Kategorie »Telefon« gehörende Vorgabe entsprechend korrigieren, angewandt auf Abbildung 38.50 also die in Zelle I6 enthaltene Vorgabe »100%« durch »80%« ersetzen.

Analog verfahren Sie mit den restlichen Ausgabenkategorien. Gibt es eine Kategorie »Autokosten« und können Sie die nur zu 60% absetzen, dann ändern Sie die zugehörige Vorgabe »100%« entsprechend in »60%« um und so weiter.

Das war's bereits. Die Mustervorlage ist nun an Ihre individuellen Umstände angepasst und kann im Vorlagenordner gespeichert werden.

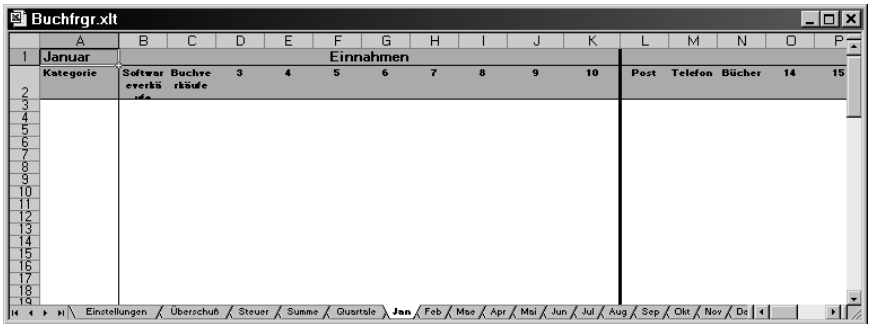
Anschließend erzeugen Sie wie erläutert mit DATEI|NEU... und Auswahl dieser Vorlage eine neue Mappe, die wie die Mustervorlage selbst aufgebaut ist, und benutzen diese Mappe nun für die Buchführung des aktuellen Jahres.

38.7.2 Einnahmen und Ausgaben erfassen

Die praktische Arbeit mit der Anwendung besteht im täglichen, wöchentlichen oder zumindest monatlichen Erfassen Ihrer Einnahmen und Ausgaben. (Oder öffnen Sie etwa nur einmal im Jahr Ihren Quittungsschuhkarton?)

Nehmen wir an, Sie wollen die Belege für den Januar erfassen. Dann aktivieren Sie die gleichnamige Tabelle (Abbildung 38.51).

Abbildung 38.51:
Tabelle »Januar«



1	Januar																		
2	Kategorie	Softwareverkäufe	Buchverkäufe	3	4	5	6	7	8	9	10	Post	Telefon	Bücher	14	15			
3																			
4																			
5																			
6																			
7																			
8																			
9																			
10																			
11																			
12																			
13																			
14																			
15																			
16																			
17																			
18																			
19																			

In Zeile 2 befinden sich automatisch die von Ihnen bei der Anpassung vergebenen Kategoriennamen »Softwareverkäufe«, »Buchverkäufe« etc.

In Spalte B der Tabelle tragen Sie nun entsprechend Ihre Einnahmen aus Softwareverkäufen ein, in Spalte C die Einnahmen aus Buchverkäufen etc. (Abbildung 38.52).

	A	B	C	D	E	F	G
1	Januar						Einnahmen
2	Kategorie	Softwareverkäufe	Buchverkäufe	3	4	5	6
3		745,00	860,00				
4		2689,00	2895,00				
5			379,00				
6							
7							

Abbildung 38.52:

Erfassung der Einnahmen

Rechts neben den zehn Einnahmekategorien befinden sich die Ausgabekategorien. In Spalte L geben Sie Ihre Postwertzeichenquittungen ein, in M die Telefonrechnungen etc. (Abbildung 38.53).

	K	L	M	N	O	P	Q
1	10	Post	Telefon	Bücher	14	15	16
2							
3		2,00	279,00	22,90			
4		5,00		79,00			
5		7,50					
6		12,00					
7		16,00					
8		2,00					
9		4,50					
10							

Abbildung 38.53:

Erfassung der Ausgaben

Ihnen stehen pro Monat in jeder Kategorie 56 Zeilen zum Eintragen ebenso vieler Belege zur Verfügung, was für den Hausgebrauch ausreichen sollte.

Tragen Sie jeweils die Bruttobeträge ein, also den Gesamtbetrag inklusive Umsatzsteuer! Um die Ermittlung der darin enthaltenen Umsatzsteuerbeträge kümmert sich die Anwendung selbst und zwar in Zeile 61 (Abbildung 38.54).

**!!
STOP**

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
58	Brutto	3434,00	4134,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	49,00	279,00	101,90
59	MWSt-Satz	16%	7%	16%	16%	16%	16%	16%	16%	16%	16%	0%	16%	7%
60	MWSt	473,66	270,45	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	38,48	6,67
61	Netto	2960,34	3863,55	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	49,00	240,52	95,23
62														
63														

Abbildung 38.54:

Ermittlung der Brutto-, Netto- und USt-Summen

Die Zeilen 59 bis 62 enthalten folgende Auswertungen:

- ➔ Zeile 59, »Brutto«: Summe der in der betreffenden Spalte eingegebenen Bruttobeträge
- ➔ Zeile 60, »USt-Sätze«: Ermittlung der für die verschiedenen Kategorien gültigen Umsatzsteuersätze.
- ➔ Zeile 61, »USt«: Summe der Umsatzsteuer in den Beträgen, die in der betreffenden Spalte eingetragen wurden
- ➔ Zeile 62, »Netto«: Summe der in der betreffenden Spalte eingegebenen Bruttobeträge minus Summe der darin enthaltenen Umsatzsteuer, also die Summe der Nettobeträge

Ob die Berechnungen korrekt sind, sehen Sie am Vergleich der vorhergehenden Bilder. Die beiden Einnahmen aus Softwareverkäufen von 745 DM und 2689 DM ergeben auch mit dem Taschenrechner zusammen 3434 DM.

Bei einem Umsatzsteuersatz von 16% sind demnach 473,66 DM an Umsatzsteuer in diesem Bruttobetrag enthalten und der Nettobetrag ist somit $3434 - 473,66 = 2960,34$ DM.



Übrigens sind die Formeln recht interessant, mit der in Zeile 60 die Umsatzsteuersätze ermittelt werden. B60 enthält beispielsweise die folgende Formel:

`=SVERWEIS(B2;USt_SÄTZE;2;FALSCH)`

Sie sucht im Bereich »USt_Sätze« den in B2 enthaltenen Text und liefert als Ergebnis den Wert, der sich in der gleichen Zeile, aber der zweiten Spalte des Bereichs, befindet.

B2 enthält im Beispiel den Text »Softwareverkäufe« (siehe Abbildung 38.51). Dieser Text wird im Bereich »USt_Sätze« gesucht, das heißt im so von mir benannten Bereich D4:E33 der Tabelle »Einstellungen«.

Die erste Spalte dieses Bereichs enthält die Kategorienbezeichnungen mit dem gesuchten Text »Softwareverkäufe« in Zelle D4 (siehe Abbildung 38.49). Die zweite Spalte enthält den zugehörigen Umsatzsteuersatz »16%«, den diese Funktion nun als Ergebnis übergibt.

Die restlichen Formeln in Zeile 60 arbeiten analog und suchen ebenfalls jeweils im Bereich »USt_Sätze« den in der betreffenden Spalte als Überschrift verwendeten Kategoriennamen und geben den zugehörigen Umsatzsteuersatz zurück.

38.7.3 Monatliche Umsatzsteuer-Voranmeldungen

Unterhalb der in den Zeilen 59 bis 62 gebildeten Summen befindet sich eine praktisch vollständige Umsatzsteuer-Voranmeldung für den betreffenden Monat, die Sie nur noch ins entsprechende amtliche Formular übertragen müssen (Abbildung 38.55).

	A	B	C	D	E	F	G	H	I	J	K	L
58												
59	Brutto	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
60	MwSt-Satz	16%	7%	16%	16%	16%	16%	16%	16%	16%	16%	0%
61	MwSt	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
62	Netto	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
63												
64												
65	Anmeldung der Umsatzsteuer-Vorau											
66	Steuerfreie Umsätze						DM	Steuer		volle DM	Steuer	
67	Steuerfreie Umsätze						0,00			48	0	
68	zum Steuersatz vor 16%						0,00	0,00		50	0	0,00
69	zum Steuersatz vor 7%						0,00	0,00		86	0	0,00
70	Umsätze, die anderen Steuersätzen unterliegen						0,00	0,00		35	0	0,00
71	Gesamt						0,00	0,00		0	0,00	
72												
73	Vorsteuerbeträge									66	0,00	
74												
75	Steuerpfl. Umsätze - Vorsteuerbeträge									83	0,00	
76												

Abbildung 38.55:
Monatliche
Umsatzsteuer-
Vor Anmeldung

In den Zeilen 66 bis 71 werden die Einnahmen behandelt, unterteilt in

- ➔ Steuerfreie Umsätze: J66 enthält die Summe der Beträge aus steuerfreien Umsätzen (das klappt nur, wenn Sie bei umsatzsteuerfreien Kategorien wie »Postwertzeichen« in der Tabelle »Einstellungen« »0%« als Umsatzsteuersatz eintragen!).
- ➔ Umsätze zum vollen Steuersatz (16%): J68 enthält die Nettobeträge, K68 die darauf anfallende Umsatzsteuer.
- ➔ Umsätze zum ermäßigten Steuersatz (7%): J69 enthält die Nettobeträge, K69 die darauf anfallende Umsatzsteuer.
- ➔ Umsätze, die anderen Steuersätzen unterliegen: J70 enthält die Nettobeträge aller Umsätze, die weder umsatzsteuerfrei sind noch den beiden Standardumsatzsteuersätzen unterliegen, und K70 enthält die darauf anfallende Umsatzsteuer.

J71 enthält die Gesamtsumme der auf diese Weise aufgesplitteten Nettobeträge und K71 die Summe der zugehörigen Umsatzsteuerbeträge.

In Zeile 73 geht es genau umgekehrt um die Umsatzsteuer, die in Ihren Ausgaben enthalten ist. K73 enthält die Summe dieser abziehbaren Vorsteuerbeträge; im Beispiel sind das 43,06 DM, die sich aus 36,39 DM Umsatzsteuer in Ihren Telefonkosten und 6,67 DM Umsatzsteuer in Ihren Ausgaben für Bücher zusammensetzen.

K75 enthält den ans Finanzamt zu überweisenden Überschuss Ihrer Umsatzsteuereinnahmen gegenüber den in Ihren Ausgaben enthaltenen Vorsteuerbeträgen, im Beispiel 675,25 DM.



Die in Spalte I enthaltenen Zahlen 48, 50 etc. entsprechen den zugehörigen Zeilennummern im Formular zur Vorsteueranmeldung und sollen eine Hilfe beim Ausfüllen dieser Formulare darstellen. Ändern sich die Formulare, sind diese Zeilennummern natürlich hinfällig und müssen von Ihnen korrigiert werden, wenn Sie auf diese Hilfestellung weiterhin Wert legen.

38.7.4 Quartals-Umsatzsteuer-Voranmeldungen

Die Anwendung erleichtert auch vierteljährliche Umsatzsteuer-Voranmeldungen. Nehmen wir an, Sie haben Belege für die Monate Januar, Februar und März erfasst (Abbildung 38.56).

Abbildung 38.56:
Erfasste Daten
eines Quartals

Buchfrgr.xlt:3

1

A

B

C

D

E

F

G

H

I

J

K

L

M

N

1

Januar

Einnahmen

2

Kategorie

Software

Buchve

3

4

5

6

7

8

9

10

Post

Telefon

Bücher

3

745,00

860,00

2,00

279,00

22,90

4

2689,00

2895,00

5,00

79,00

5

379,00

7,50

6

12,00

7

16,00

8

2,00

9

10

H <

>

H |

Überschuß

|

Steuer

|

Summe

|

Quartale

|

Jan

|

Feb

|

Mär

|

Apr

|

Mai

|

Jun

|

Jul

|

Aug

|

Sep

|

Okt

|

Nov

|

Dez

|

Buchfrgr.xlt:2

1

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

1

Februar

Einnahmen

2

Kategorie

Software

Buchve

3

4

5

6

7

8

9

10

Post

Telefon

Bücher

3

453,00

680,00

10,00

198,00

4

1290,00

12,00

H <

>

H |

Überschuß

|

Steuer

|

Summe

|

Quartale

|

Jan

|

Feb

|

Mär

|

Apr

|

Mai

|

Jun

|

Jul

|

Aug

|

Sep

|

Okt

|

Nov

|

Dez

|

Buchfrgr.xlt:1

1

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

1

März

Einnahmen

2

Kategorie

Software

Buchve

3

4

5

6

7

8

9

10

Post

Telefon

Bücher

3

2800,00

4

2,00

5

6,00

6

8,00

H <

>

H |

Überschuß

|

Steuer

|

Summe

|

Quartale

|

Jan

|

Feb

|

Mär

|

Apr

|

Mai

|

Jun

|

Jul

|

Aug

|

Sep

|

Okt

|

Nov

|

Dez

|

Um nun die Umsatzsteuer-Voranmeldung für dieses Quartal vorzunehmen, können Sie die in Abbildung 38.54 gezeigten monatlichen Auswertungen für die drei Monate addieren. Wesentlich einfacher: Aktivieren Sie die Tabelle »Quartale«, in der genau diese Additionen automatisch vorgenommen werden (Abbildung 38.57).

»Quartale« enthält einen Block mit den benötigten Zahlen fürs erste Quartal, darunter einen Block, in dem die Zahlen des zweiten Quartals addiert werden und so weiter.



Übrigens: Wie Abbildung 38.60 zeigt, enthält die Tabelle »Überschuss« zusätzlich eine Jahresauswertung der Umsatzsteuereinnahmen und Vorsteuer Ausgaben!

Buchfrgr.xlt

	A	B	C	D	E	F	G	H	I	J	K	
1		1. Quartal										
2		Anmeldung der Umsatzsteuer-Vorauszahlung						DM	Steuer	volle DM		Steuer
3		Steuerfreie Umsätze						0,00		48	0	
4		Steuerpflichtige Umsätze										
5		zum Steuersatz von 16%						4462.93	714.07	50	4462	713.92
6		zum Steuersatz von 7%						7115.89	498.11	86	7114	497.98
7		Umsätze, die anderen Steuersätzen unterliegen						0,00	0,00	35	0	0,00
8		Gesamt						11578.82	1212.18		11576	1211.90
9		Vorsteuerbeträge								66	72.46	
10		Steuerpfl. Umsätze - Vorsteuerbeträge								83	1139.44	
11												
12												
13												
14												
15												
16												
17		2. Quartal										
18		Anmeldung der Umsatzsteuer-Vorauszahlung						DM	Steuer	volle DM		Steuer
19		Steuerfreie Umsätze						0,00		48	0	
20		Steuerpflichtige Umsätze										
21		zum Steuersatz von 16%						0,00	0,00	50	0	0,00
22		zum Steuersatz von 7%						0,00	0,00	86	0	0,00
23		Umsätze, die anderen Steuersätzen unterliegen						0,00	0,00	35	0	0,00
24		Gesamt						0,00	0,00		0	0,00
25		Vorsteuerbeträge								66	0,00	
26		Steuerpfl. Umsätze - Vorsteuerbeträge								83	0,00	
27												
28												
29												

Überschub Steuer Summe Quartale Jan Feb Mar Apr Mai Jun Jul Aug Sep Okt Nov Dez

Abbildung 38.57:
Tabelle »Quartale«

38.7.5 Überschussrechnung

Auch die jährliche Überschussrechnung, die die Grundlage Ihrer Einkommensteuererklärung bildet, wird durch die Anwendung erleichtert, zunächst einmal durch die Tabelle »Summe«, die eine übersichtliche Aufstellung der monatlichen Einnahmen und Ausgaben enthält, inklusive der benötigten Summierung dieser monatlichen Zahlen (Abbildung 38.58).

Buchfrgr.xlt													
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Einnahmen											Ausgaben	
2	Kategorie	Softwareverkäufe	Buchverkäufe	3	4	5	6	7	8	9	10	Post	Telefon
3	Jan	2960,34	3863,55	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	49,00	240,52
4	Feb	1502,59	635,51	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	22,00	170,89
5	Mär	0,00	2616,82	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	16,00	0,00
6	Apr	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
7	Mai	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
8	Jun	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
9	Jul	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
10	Aug	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
11	Sep	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
12	Okt	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
13	Nov	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
14	Dez	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
15	Netto	4462,93	7115,89	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	87,00	411,21
16	Jan	473,66	270,45	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	36,48
17	Feb	240,41	44,49	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	27,31
18	Mär	0,00	183,18	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
19	Apr	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
20	Mai	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
21	Jun	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
22	Jul	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
23	Aug	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
24	Sep	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
25	Okt	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
26	Nov	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
27	Dez	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
28	MWS	714,07	498,11	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	65,79
29	Jan	3434,00	4134,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	49,00	279,00

Abbildung 38.58:
Tabelle »Summe«

Die Tabelle enthält drei untereinander angeordnete Blöcke: Der oberste reicht von Zeile 3 bis Zeile 15. Die Zeilen 3 bis 14 enthalten für jeden Monat die Nettobeträge der verschiedenen Einnahme-/Ausgabearten. In Zeile 15 werden daraus die Gesamtsummen für das gesamte Jahr gebildet.

Der zweite, von Zeile 16 bis 28 reichende Block, enthält die zugehörigen Umsatzsteuerbeträge, und der von Zeile 29 bis 41 reichende dritte Block enthält die Bruttobeträge.

Das war's im Wesentlichen. Die in »Quartale« und in »Summen« enthaltenen Auswertungen bilden die Grundlage Ihrer jährlichen Einkommensteuererklärung.

Möglicherweise – das ist jedoch individuell unterschiedlich – helfen Ihnen dabei zusätzlich noch die beiden Tabellen »Steuer« und »Überschuss«. In »Steuer« erfassen Sie alle Zahlungen ans bzw. Erstattungen vom Finanzamt (Abbildung 38.59).

Abbildung 38.59:
Tabelle »Steuer«

Buchfrgr.xlt												
	A	B	C	D	E	F	G	H	I			
1	Steuerzahlungen/erstattungen											
2												
3												
4	Einkommenssteuer				7598,00	Solidaritätszuschlag				531,86		
5	Datum Zahlung Erstattung Bemerkung					Datum Zahlung Erstattung Bemerkung						
6	23.01.06	3799,00		EKSt-Vorauszahlung für's 1.Quartal 2006	23.01.06	265,93		SolSt-Vorauszahlung für's 1.Quartal 2006				
7	13.04.06	3799,00		EKSt-Vorauszahlung für's 2.Quartal 2006	13.04.06	265,93		SolSt-Vorauszahlung für's 2.Quartal 2006				
8												
9												
10												
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												
21												
22												
23												
24	Summe				7598,00	0,00	Summe				531,86	0,00
25												
26												
27	Umsatzsteuer				0,00	Lohn-Kirchensteuer				0,00		
28	Datum Zahlung Erstattung Bemerkung					Datum Zahlung Erstattung Bemerkung						
29												
30												
31												
32												
33												
34												
H	Einstellungen Überschub Steuer Summe Quartale Jan Feb Mar Apr Mai Jun Jul Aug Sep Okt Nov Dez / 1 4											

Im oberen linken Block erfassen Sie Zahlungen/Erstattungen zur Einkommensteuer und im rechten Block darauf fällige »Solidaritäts«-Zuschläge. Im linken unteren Block erfassen Sie Zahlungen/Erstattungen zur Umsatzsteuer und im rechten unteren Block Zahlungen/Erstattungen zu Lohn- und Kirchensteuer.

D4, I4, D27 und I27 ermitteln daraufhin die resultierenden Summen aller Zahlungen bzw. Erstattungen in der jeweiligen Kategorie.

Die Tabelle »Überschuss« ermittelt den Jahresüberschuss der Betriebseinnahmen gegenüber den Betriebsausgaben (Abbildung 38.60).

Der linke Teil enthält eine aus der Tabelle »Summe« übernommene Aufstellung der Jahresnettoeinnahmen in den verschiedenen Kategorien und der fälligen Umsatzsteuer.

Buchfrgr.xlt									
A	B	C	D	E	F	G	H	I	J
2	Überschub der Betriebseinnahmen über die Betriebsausgaben								12125,10
3	Betriebseinnahmen				Betriebsausgaben				
4	Kategorie		Betrag		Kategorie	Betriebsanteil	Betrag		
5	Softwareverkäufe	Netto	4462,93		Post	100%	87,00		
6		Fällige Umsatzsteuer	714,07		Telefon	100%	411,21		
7	Buchverkäufe	Netto	7115,89		Bücher	100%	95,23		
8		Fällige Umsatzsteuer	498,11		14	100%	0,00		
9	3	Netto	0,00		15	100%	0,00		
10		Fällige Umsatzsteuer	0,00		16	100%	0,00		
11	4	Netto	0,00		17	100%	0,00		
12		Fällige Umsatzsteuer	0,00		18	100%	0,00		
13	5	Netto	0,00		19	100%	0,00		
14		Fällige Umsatzsteuer	0,00		20	100%	0,00		
15	6	Netto	0,00		21	100%	0,00		
16		Fällige Umsatzsteuer	0,00		22	100%	0,00		
17	7	Netto	0,00		23	100%	0,00		
18		Fällige Umsatzsteuer	0,00		24	100%	0,00		
19	8	Netto	0,00		25	100%	0,00		
20		Fällige Umsatzsteuer	0,00		26	100%	0,00		
21	9	Netto	0,00		27	100%	0,00		
22		Fällige Umsatzsteuer	0,00		28	100%	0,00		
23	10	Netto	0,00		29	100%	0,00		
24		Fällige Umsatzsteuer	0,00		30	100%	0,00		
25	USt-Erstattungen			0,00	Zugehörige USt			72,46	
26	Lohnsteuer-Erstattungen			0,00	USt-Zahlungen			0,00	
27	Summe			12791,00	Lohnsteuer-Zahlungen			0,00	
28								665,90	

Abbildung 38.60:

Tabelle
»Überschuss«

Zusätzlich werden diesen Betriebseinnahmen in B25:E27 die in der Tabelle »Steuer« von Ihnen erfassten Umsatz- und Lohnsteuererstattungen hinzuge-rechnet und in E27 die Gesamtsumme all dieser Einnahmen gebildet.

Analog dazu enthält der rechte Teil eine Aufstellung der jährlichen Nettoausgaben. Prinzipiell werden natürlich auch diese Zahlen aus der Tabelle »Summe« übernommen. Allerdings wird hier nur der unter »Betriebsanteil« von Ihnen eingegebene Prozentsatz ausgewiesen!



Beispielsweise zeigt Abbildung 38.58, dass die gesamte Nettoausgabe für »Post« 87 DM betrug (Zelle L15). Würde die Vorgabe »100%« für den Betriebsanteil der Kategorie »Post« durch »80%« ersetzt, so würden in der Tabelle »Überschuss« dementsprechend nur 80% dieser 87 DM als Betriebsanteil ausgewiesen, also 69,60 DM.

Am unteren Ende des Ausgabenblocks werden die in der Tabelle »Summe« enthaltenen Umsatzsteuerausgaben addiert und die in der Tabelle »Steuer« von Ihnen erfassten Umsatzsteuer- und Lohnsteuerzahlungen (siehe J25;J27). In J28 wird die Gesamtsumme der Betriebsausgaben gebildet.

J2 wiederum enthält den Überschuss der Einnahmen gegenüber den Ausgaben, im Beispiel 12.125,10 DM.

Noch ein Stück weiter unten befindet sich ein vertrauter Anblick (Abbildung 38.61).

Nach dem gewohnten Schema werden hier die jährlichen Umsatzsteuereinnahmen und Vorsteuerausgaben miteinander verglichen.

Abbildung 38.61:
Jahres-Umsatz-
steuerermittlung

Wie gesagt, ob Sie die beiden Tabellen »Steuer« und vor allem die doch recht stark auf mich selbst zugeschnittene Tabelle »Überschuss« gebrauchen können, muss sich herausstellen. Was jedoch die zuvor erläuterten Tabellen angeht, besteht wohl kein Zweifel daran, dass sie Ihre Buchführung erheblich vereinfachen.

38.8 Mit VBA auf Dateien beliebiger Art zugreifen

Das folgende Kapitel erläutert den programmgesteuerten Umgang mit Dateien. Sie erfahren, wie Sie beliebige Dateien (nicht nur Excel-Dateien) programmgesteuert erzeugen, löschen und manipulieren können.

Einige Anwendungsbeispiele: Die VBA-Bibliothek enthält verschiedenste Anweisungen und Funktionen zum Manipulieren von Dateien und Verzeichnissen. Beispielsweise löscht die Anweisung

Kill Pfadname

die Datei mit dem angegebenen Pfadnamen. Entsprechend löscht

Kill "c:\test.xls"

die Datei TEST.XLS, die sich im Stammverzeichnis von C: befindet.

Mit der Funktion *MkDir Pfad* können Sie ein neues Verzeichnis namens »Pfad« erstellen, mit *Rmdir Pfad* das Verzeichnis »Pfad« löschen, mit *ChDir Pfad* das aktuelle Verzeichnis wechseln und so weiter.

Oft sind jedoch die Anweisungen und Funktionen interessanter, die sich mit einzelnen Dateien und deren Manipulation beschäftigen. Daher erläutere ich zunächst, wie Dateien vor einer Manipulation geöffnet und danach wieder geschlossen werden. Danach geht es um die verschiedenen zur Verfügung stehenden Manipulationstechniken.

Die einfachste ist der sequentielle Zugriff, bei dem die zu verarbeitenden Daten unmittelbar hintereinander gespeichert werden, ähnlich wie Lieder auf einem Tonband. Umgekehrt können die Daten auch nur nacheinander gelesen werden.

Erheblich flexibler sind Direktzugriffsdateien, bei denen jeder Datensatz in einem eigenen Block fester Länge gespeichert wird. Jeder Satz besitzt eine eigene Satz- oder Recordnummer, über die ein direkter Zugriff auf den betreffenden Block möglich ist.

Binärdateien gestatten ebenfalls einen unmittelbaren Zugriff auf alle in einer Datei enthaltenen Daten. Binärer Zugriff behandelt eine Datei jedoch weder als Folge unstrukturierter Zeilen (sequentielle Datei) noch als strukturierte Sätze (Direktzugriffsdatei), sondern als Folge einzelner Bytes – und jedes Byte kann gezielt manipuliert werden. Daher eignet sich der binäre Zugriff beispielsweise zur Manipulation von Programmdateien.

All diese Dateiformen werden in der gleichen Weise erläutert: Zunächst lernen Sie die Zugriffsprinzipien und die dazu verwendeten Befehle anhand kleiner Zwei- oder Dreizeiler kennen, danach folgt der praxigerechte Abschluss in Form eines kleinen Demoprogramms, beispielsweise anhand eines Filterprogramms, das ein bestimmtes Zeichen in einer Datei sucht und es überall in der Datei durch ein anzugebendes anderes Zeichen ersetzt.

Alle im Folgenden vorgestellten Prozeduren befinden sich übrigens im Modul »Dateimanipulation« der Mappe VB.XLS.



38.8.1 Dateiformen

VBA kennt drei Arten von Dateien:

- ➔ In sequentiellen Dateien werden beliebige Daten unmittelbar hintereinander gespeichert, ähnlich wie Lieder auf einem Tonband. Der Vergleich stimmt allerdings nur, wenn wir von einem Tonband ohne Zählwerk ausgehen. Da wir uns die Position eines bestimmten Liedes ohne Zählwerk nicht merken können, müssen wir sequentiell suchen, uns Lied für Lied vortasten, bis das gewünschte gefunden wurde.
- ➔ Binärdateien behandeln Daten nicht als Zahlen oder Zeichenketten, sondern als einzelne Bytes. Auf jedes dieser Bytes können Sie unter Angabe der Bytenummer direkt zugreifen, um es zu lesen oder zu manipulieren. Ein Beispiel: Sie können unmittelbar auf das Byte Nummer 3168 einer Datei lesend zugreifen. Sie stellen fest, dass es das Zeichen »a« enthält. Wenn Sie wollen, können Sie dieses Zeichen nun durch ein »x« ersetzen, ohne die restlichen Bytes der Datei zu verändern.

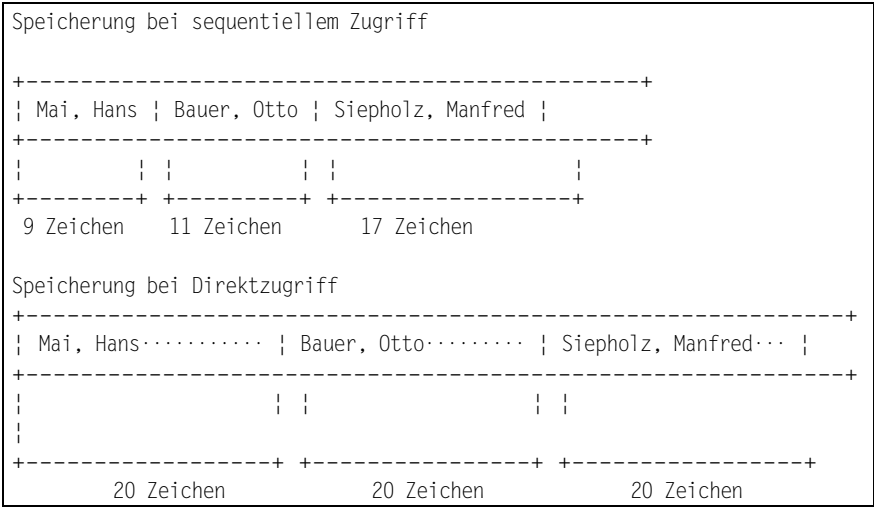
Dank dieser Möglichkeit, direkt auf beliebige Bytes lesend und schreibend zugreifen zu können, eignet sich der binäre Zugriff theoretisch hervorragend zur Dateiverwaltung. Allerdings wäre der praktische Umgang mit Binärdateien ziemlich umständlich. Um die Adresse Nummer 10 zu lesen, müssten Sie zuerst einmal errechnen, ab dem wievielten Byte diese Adresse in Ihrer Datei gespeichert ist.

➔ Einfacher ist die Datenverwaltung mit Direktzugriffsdateien, die ebenso wie Binärdateien einen unmittelbaren Lese- und Schreibzugriff auf Daten ermöglichen, allerdings nur auf einheitlich strukturierte Daten, auf Daten des gleichen Typs, zum Beispiel auf Adressen. Direktzugriffsdateien dürfen keinen völlig zufälligen Mischmasch von Daten enthalten wie sequentielle oder Binärdateien. Dafür haben sie gegenüber Binärdateien den Vorteil, dass zum Zugriff auf eine bestimmte Adresse keine Bytenummer benötigt, sondern einfach die Nummer der Adresse angegeben wird. Sie können jederzeit auf die Adresse Nummer 5 oder 10 der Datei zugreifen. VBA ermittelt für Sie, wo sich die betreffende Adresse befindet.

Möglich wird das, da die Daten sauberlich getrennt in Blöcken immer gleicher Länge gespeichert werden. Jede Adresse befindet sich in einem eigenen Block, Adresse 1 im ersten Block, Adresse 2 im zweiten Block und so weiter. Da alle Blöcke gleich lang sind, kann VBA unter Angabe der Blocknummer problemlos errechnen, wo sich ein bestimmter Block innerhalb der Datei befindet und direkt auf diesen zugreifen.

Die folgende Abbildung zeigt den Unterschied zwischen einer sequentiellen und einer Direktzugriffsdatei (Abbildung 38.62):

Abbildung 38.62:
Datenspeicherung



Bei der sequentiellen Speicherung von Daten befinden sich alle Daten unmittelbar nebeneinander, die Datei ist extrem komprimiert. Angenommen, Sie wollen auf die dritte Adresse zugreifen, auf »Siepholz, Manfred«. In der Abbildung sehen Sie, wie lang die beiden vorhergehenden Adressen sind und wissen daher, ab dem wievielten Zeichen der Datei die gesuchte Adresse beginnt.

In der Praxis weiß jedoch weder VBA noch Sie, welche Adressen vor der dritten gespeichert sind und wie lang diese Adressen sind. Um an die dritte Adresse heranzukommen, sind daher drei Leseanweisungen nötig. Die erste liest »Mai, Hans«. Die nächste »Bauer, Otto« und die dritte die gesuchte Adresse.

Das zweite Beispiel zeigt, wie die gleichen Adressen per Direktzugriff gespeichert werden. Die Datei wird in Sätze gleicher Länge unterteilt. Die Länge bestimmen Sie beim Öffnen mit der Angabe einer Satzlänge. Hier wurde als Satzlänge 20 Zeichen gewählt.

Die erste Adresse umfasst nur neun Zeichen. Dennoch wird für diese Adresse ein kompletter, 20 Zeichen langer Block verwendet. Die Zeichen Nummer 10 bis 20 sind Leerzeichen, hier durch Punkte angedeutet. Ebenso wird zur Speicherung der zweiten und der dritten Adresse jeweils ein kompletter Satz oder Record verwendet.

Das Resultat ist auf der einen Seite eine ziemliche Platzverschwendung. Die Datei belegt auf dem Speichermedium wesentlich mehr Platz, als angesichts der darin enthaltenen Daten eigentlich notwendig wäre.

Dafür kann VBA jedoch auf jede Adresse direkt zugreifen, ohne sich vom Dateianfang aus vortasten zu müssen: Da alle Blöcke exakt gleich lang sind, ergibt sich die Entfernung der dritten Adresse vom Dateianfang aus der Satznummer 3 der gesuchten Adresse und der Blocklänge von 20 Zeichen.

VBA kann den Schreib-/Lesekopf direkt zur ermittelten Position bewegen und die Adresse einlesen, ohne die echte Länge der beiden vorhergehenden Adressen zu kennen.

Eigentlich ist es falsch, von Dateiformen zu sprechen. Es handelt sich hier um Zugriffsformen. Eine beliebige vorhandene Datei können Sie mit sequentiellem, direktem oder binärem Zugriff behandeln, je nachdem was Sie gerade bezwecken.

Der Unterschied: Beim sequentiellen Zugriff wird die Datei zeilenweise behandelt, ähnlich einem Text mit unterschiedlich langen Textzeilen, beim Direktzugriff satzweise, wie ein Karteikasten mit immer gleich strukturierten Datensätzen, und beim binären Zugriff als Folge einzelner Bytes.

38.8.2 Dateien öffnen und schließen



Die Verwaltung von Dateien erfolgt immer in drei Schritten:

1. *Die Datei wird mit der Open-Anweisung geöffnet.*
2. *Daten werden geschrieben oder aus der Datei gelesen.*
3. *Die Datei wird mit der Close-Anweisung geschlossen.*

Open-Anweisung

Das Öffnen einer Datei hat den Sinn, VBA den Dateinamen und die Art des Zugriffs bekanntzugeben.

Open Datei [For Modus] [Access Zugriff] [Sperrtyp] As [#]Dateinummer [Len = Satzlänge]

Datei ist ein Stringausdruck, der den DOS-Konventionen für Pfadangaben und Dateinamen entspricht (maximal acht Zeichen plus – optional – eine maximal drei Zeichen lange Erweiterung), zum Beispiel »test.dat« oder »c:\daten\test.dat«.

Modus ist der Dateityp.

Zugriff spezifiziert die gewünschte Zugriffsform in einem Netzwerk, ebenso wie *Sperrtyp*.

Dateinummer ist eine Zahl zwischen 1 und 255. Das Zeichen »#« ist das Nummernzeichen (nicht mit dem identischen Datentypkennzeichen zu verwechseln) und hat keine weiteren Auswirkungen.

Satzlänge ist die gewünschte Blocklänge einer Direktzugriffsdatei.

Lässt man alle optionalen Argumente weg, kommt man zu

Open Datei As Dateinummer

als Minimalsyntax. Mindestens ein Name und eine Dateinummer sind anzugeben, zum Beispiel so:

Open "test.dat" As 1

Diese Anweisung öffnet eine Datei unter dem Namen TEST.DAT und gibt ihr die Nummer 1. Die Dateinummer ist wichtig, da mehrere Dateien gleichzeitig offen sein können und Sie daher bei Lese- oder Schreibanweisungen an VBA mitteilen müssen, welche der geöffneten Dateien Sie meinen, indem Sie jeweils die betreffende Dateinummer angeben.

Wenn Sie den Zugriffsmodus wie hier nicht angeben, öffnet VBA die Datei im Standardmodus *Direktzugriff*. Wenn Ihnen das nicht passt, müssen Sie die gewünschte Zugriffsart mit dem optionalen Argument *Modus* näher bestimmen. Fünf Modi stehen Ihnen zur Verfügung:

Modus	Zugriffsart	Ein-/Ausgabe
Input	Sequentiell	Lesen
Output	Sequentiell	Schreiben
Append	Sequentiell	Schreiben (am Dateiende)
Random	Direktzugriff	Lesen und Schreiben
Binary	Binär	Lesen und Schreiben

Tabelle 38.1:
Zugriffsmodi

Die drei ersten Modi, *Input*, *Output* und *Append*, beziehen sich ausschließlich auf sequentielle Dateien. Bei dieser Zugriffsart ist es leider nicht möglich, in eine geöffnete Datei abwechselnd Daten zu schreiben und zu lesen. Bereits beim Öffnen muss die gewünschte Operation angegeben werden. Zum Beispiel öffnet

```
Open "test.dat" For Input As 1
```

eine unter dem Namen TEST.DAT bereits existierende Datei und ermöglicht Lesezugriffe darauf. Und

```
Open "test.dat" For Output As 1
```

öffnet eine Datei unter dem Namen TEST.DAT und ermöglicht es, Daten in dieser Datei zu speichern. Dabei werden – falls TEST.DAT bereits existiert – schon vorhandene Daten einfach überschrieben. Im Gegensatz zum Öffnen der gleichen sequentiellen Datei mit

```
Open "test.dat" For Append As 1.
```

Im Modus *Append* können Sie ebenfalls nur Daten schreiben, nicht lesen. Die gespeicherten Daten überschreiben jedoch keine bereits vorhandenen Daten, sondern werden an das Dateiende angehängt.

Die beiden letzten Modi beziehen sich auf Direktzugriffs- und Binärdateien. Beide Zugriffsarten erlauben sowohl Schreib- als auch Lesezugriffe auf eine geöffnete Datei. Die Anweisung

```
Open "test.dat" For Binary As 1
```

öffnet TEST.DAT als Binärdatei mit der Dateinummer 1 und

```
Open "test.dat" For Random As 1
```

öffnet TEST.DAT als Direktzugriffsdatei.

Da eine Direktzugriffsdatei in Blöcke gleicher Länge unterteilt ist, die so genannte Satzlänge oder Recordlänge, besteht die Frage, welche Satzlänge hier benutzt wird. Die Standardsatzlänge ist 128 Byte. Das heißt, die in TEST.DAT gespeicherten einheitlichen Daten – zum Beispiel Adressen – dürfen nicht mehr als 128 Zeichen enthalten.

Außer Sie bestimmen mit der optionalen Angabe *Satzlänge* die Blocklänge selbst. Zum Beispiel öffnet

```
Open "test.dat" For Random As 1 Len = 256
```

eine Direktzugriffsdatei TEST.DAT mit einer Satzlänge von 256 Zeichen. Die maximale Satzlänge beträgt übrigens 32767 Byte.

Die optionale Satzlänge besitzt übrigens auch für sequentielle Dateien eine Bedeutung, obwohl es dort keine gleich langen Sätze im Sinne einer Direktzugriffsdatei gibt. Wenn Sie Daten in eine Datei schreiben, werden diese zuerst in einem Puffer gesammelt, der normalerweise 512 Byte (Zeichen) umfasst, und erst auf die Diskette/Festplatte übertragen, wenn der Puffer voll ist oder die Datei geschlossen wird.

Bei der Arbeit mit sequentiellen Dateien können Sie mit der Angabe *Satzlänge* die Größe dieses Puffers selbst bestimmen. Je größer er ist, desto seltener werden Sie bei der Eingabe vieler Adressen durch das Anlaufen der Festplatte unterbrochen, umso größer ist allerdings auch die Gefahr, Daten, die sich noch im Puffer befinden und noch nicht auf die Festplatte übertragen wurden, durch einen Stromausfall zu verlieren.

Sperrung definiert das Verhalten der geöffneten Datei im Netzwerk und ist eines der Schlüsselwörter *Shared* (gemeinsamer Zugriff mehrerer Terminals möglich), *Lock Read* (Lesezugriff ist von keinem anderen Terminal aus möglich), *Lock Write* (Schreibzugriff ist von keinem anderen Terminal aus möglich) oder *Lock Read Write* (weder Lese- noch Schreibzugriff durch andere Anwendungen möglich).

Close-Anweisung

Nach der Ausführung der Schreib-/Leseaktionen wird eine Datei mit *Close* wieder geschlossen:

```
Close [[#]Dateinummer [, [#]Dateinummer]...]
```

Close schließt die Datei mit der angegebenen Nummer bzw. die Dateien, deren Nummern Sie – mit Kommata als Trennzeichen – angeben. Das Nummernzeichen »#« hat wie bei *Open* keine weiteren Auswirkungen. Entsprechend schließt

`Close 1`

die unter der Nummer 1 geöffnete Datei. *Close 1, 2* würde die beiden Dateien mit den Dateinummern 1 und 2 schließen. *Close* ohne Angabe einer Dateinummer schließt alle geöffneten Dateien.

Das Schließen einer Datei sollten Sie niemals vergessen, vor allem nicht nach Schreibzugriffen. Wie erläutert, werden die zu speichernden Daten in einem Puffer gesammelt und der Puffer immer dann, wenn er voll ist, auf den Massenspeicher übertragen.



Es ist sehr unwahrscheinlich, dass der Puffer zufällig gerade voll ist, wenn Sie mit dem Schreiben von Daten fertig sind. Also enthält der Puffer noch einen Rest nicht gespeicherter Daten und die werden erst beim Schließen der Datei gespeichert.

38.8.3 Sequentielle Dateien

Sie wissen inzwischen, was eine sequentielle Datei ist und wie man eine Datei öffnet und schließt. Dann fehlen nur noch die Lese- und Schreib- anweisungen:

Anweisung/Funktion	Funktion
<code>Print</code>	Ausgabe analog der Bildschirmausgabe
<code>Write</code>	Ausgabe mit Zeichenketten in Anführungszeichen
<code>Input</code>	Lesebefehl zu mit <i>Print</i> geschriebenen Daten
<code>Line Input</code>	Lesebefehl zu mit <i>Write</i> geschriebenen Daten
<code>Input-Funktion</code>	Begrenzte Zeichenanzahl aus Datei lesen

Tabelle 38.2:
Lese- und Schreib-
anweisungen

Die gebräuchlichsten Schreib- und Leseanweisungen sind *Print* und *Input*:

```
Print #Dateinummer, Ausdrucksliste[,|;]]
Input #Dateinummer, Variable
```

**Print- und Input-
Anweisung**

Die Syntax von *Print* entspricht bis auf die Angabe *#Dateinummer* der *Print*-Methode zur Bildschirmausgabe. *#Dateinummer* ist jene Dateinummer, die Sie in der *Open*-Anweisung benutzten. *Input* liest gespeicherte

Daten wieder ein und speichert sie in der angegebenen Variablen, die einen passenden Typ besitzen muss.

Wir können nun bereits eine erste sequentielle Datei erstellen, die in einer Datei namens TEST.DAT zwei Zeichenketten speichert und beide Strings zur Kontrolle wieder einliest (Abbildung 38.63).

Abbildung 38.63:

Vb.xls, Modul
»Dateimanipulation«, Prozedur
»Seqfile«



Der erste Programmteil öffnet im aktuellen Verzeichnis (da kein Pfad angegeben ist) eine Datei TEST.DAT im Modus *Output* mit der Dateinummer 1. Das heißt in die Datei werden Daten geschrieben.

Print #1, »Hans Maier« speichert die Zeichenkette »Hans Maier« am Anfang der Datei. Die zweite *Print*-Anweisung speichert dahinter den String »Otto Müller«, den Inhalt der Variablen *a*. Die Schreibenweisungen sind beendet und die Datei wird geschlossen.

Der zweite Teil öffnet die Datei im Lesemodus *Input*. Die erste *Input*-Anweisung liest die erste in der Datei enthaltene Zeichenkette ein, die nächste die zweite Zeichenkette. Beide Zeichenketten werden zur Kontrolle im Direktfenster ausgegeben und die Datei geschlossen.

Sie können mehrere zu speichernde Daten durch Kommata oder Semikola voneinander trennen und mit einer einzigen *Print*-Anweisung in der Datei speichern. Dabei schreibt *Print* die Daten exakt so in die Datei, wie Sie es von Bildschirmausgaben im Direktfenster mit *Print.Debug* gewohnt sind.

Enthalten die Variablen *a*, *b* und *c* die Zeichenketten »Test1«, »Test2« und »Test3«, so speichert *Print #1, a; b; c* in der Datei die Zeichenkette

```
Test1Test2Test3.
```

Die alternative Schreibweise *Write* schließt dagegen jede Zeichenkette in Anführungszeichen und trennt alle Ausgaben durch je ein Komma voneinander. Als Trennzeichen dürfen im Gegensatz zu *Print* nur Kommata und keine Semikola verwendet werden. *Write #1, a, b, c* speichert in der Datei daher folgende Daten:

Write-Anweisung

```
"Test1", "Test2", "Test3"
```

Eines ist beiden Schreibweisen gemeinsam: Den Abschluss einer Ausgabeanweisung bildet ein Zeilenvorschub und ein Wagenrücklauf. Auf dem Bildschirm bedeutet das, dass der Cursor auf den Anfang der folgenden Zeile positioniert wird.

Bei der Ausgabe auf Diskette/Festplatte wird stattdessen eine Zeilenende-Markierung <CR><LF> in die Datei geschrieben, eine Kombination aus zwei ASCII-Codes, dem Code 13 (Wagenrücklauf) und dem Code 10 (Zeilenvorschub).

Merken Sie sich bitte, dass diese beiden ASCII-Codes jede *Print* oder *Write*-Anweisung abschließen, gewissermaßen das Zeilenende bilden.

Eine Leseanweisung wie *Input #1, a* liest bis zum Trennzeichen »,« oder bis zum Erreichen des Zeilenendes, das durch die Kombination <CR><LF> gekennzeichnet wird. Entsprechend würde im Beispiel (*Print #1, a; b; c*) die Anweisung *Input #1, a* der Variablen *a* die gesamte in der Datei gespeicherte Zeile »Test1Test2Test3« zuweisen.

Es ist keine Unterscheidung zwischen »Test1«, »Test2« und »Test3« möglich. Für *Input #* handelt es sich mangels Trennzeichen um eine einzige einzulesende Datenmenge.

Ganz anders dagegen ist es bei der mit *Write* erstellten Datei. Die vier gespeicherten Daten sind eindeutig voneinander getrennt, durch ein Komma bzw. durch das Zeilenende <CR><LF>.

Jede *Input*-Anweisung liest völlig korrekt immer nur bis zu einem dieser Trennzeichen und überliest übrigens ebenfalls korrekt die Anführungszeichen, die eine Zeichenkette einschließen.

Drei aufeinander folgende *Input*-Anweisungen lesen daher die ursprünglichen Daten wieder ein:

```
Input #1, a 'liest die Zeichenkette Test1  
Input #1, b 'liest die Zeichenkette Test2  
Input #1, c 'liest die Zeichenkette Test3
```

Die Kombination von *Write* und *Input* eignet sich somit recht gut zur Behandlung von sequentiellen Dateien, vor allem, da auch Zeichenketten, die selbst Kommata enthalten, problemlos gelesen werden. Daher wundern Sie sich bestimmt darüber, dass ich dennoch zum Speichern von Zeichenketten *Print* vorziehe.

Dafür gibt es zwei Gründe. Zum einen verschwenden die von *Write* verwendeten Anführungszeichen und Kommata Platz. Und vor allem wird es problematisch, wenn die in Anführungszeichen gesetzte Zeichenkette selbst ein Anführungszeichen enthält.

Darüber hinaus ist es einfach unüblich, dass Texte auf der Diskette/Platte in Anführungszeichen gespeichert werden. (Sehen Sie sich mit dem DOS-Befehl *TYPE* mal einige ASCII-Dateien auf Ihrer Festplatte an, speziell jene mit einer Endung wie *DOC* oder ähnlich.)

Wenn die Verwendung von *Print* unproblematisch sein soll, müssen allerdings bestimmte Voraussetzungen erfüllt sein: Zunächst muss für jede Ausgabe eine eigene *Print*-Anweisung verwendet werden, statt alle auszugebenden Daten durch Komma/Semikolon getrennt in einer einzigen *Print*-Anweisung aufzulisten. Da dann alle Daten eindeutig durch <CR><LF> getrennt sind, ist das Einlesen völlig unproblematisch. Jedenfalls dann, wenn zugleich die zweite Voraussetzung erfüllt ist: Zum Einlesen wird nicht *Input*, sondern *Line Input* verwendet.

**Line Input-
Anweisung**

Line Input behandelt ein in einer Zeichenkette enthaltenes Komma nicht als Stringendezeichen, sondern liest die Zeichenkette auf alle Fälle vollständig ein, bis zum durch <CR><LF> gekennzeichneten Zeilenende. Daher sind auch Zeichenketten mit Kommata völlig unproblematisch.



Input-Funktion

*Fassen wir zusammen: Sie speichern Daten in sequentiellen Dateien am besten mit *Print* und lesen sie mit *Line Input* wieder ein. Numerische Daten speichern Sie ebenfalls mit *Print* und lesen sie mit *Input*.*

Die Anweisungen (!) *Input* und *Line Input* lesen eine Datei zeilenweise, bis zum Zeilenende <CR><LF> beziehungsweise – nur *Input* – einem vorausgehenden Komma. Die *Input*-Funktion(!) liest dagegen eine festgelegte Zeichenanzahl aus einer Datei:

```
v$ = Input(Anzahl, [#]Dateinummer)
```


Anzahl ist die gewünschte Anzahl zu lesender Zeichen, *Dateinummer* die beim Öffnen angegebene Dateinummer. Diese Funktion ist für Sonderaufgaben recht brauchbar.

Ein Beispiel: Sie wollen eine Textdatei kopieren. Sie öffnen diese Datei und gleichzeitig eine zweite neue Datei. Dann lesen Sie jeweils einen String der Textdatei und schreiben ihn unverändert in die zweite geöffnete Datei.

Angenommen, die Textdatei enthält eine einzige 80 Kbyte lange Zeichenkette. Erst am Dateiende markiert eine <CR><LF>-Kombination das Zeilenende. Dann führt die Anweisung

```
Input #1, a$
```

zu einer Fehlermeldung, denn VBA betrachtet die gesamte Programmdatei als einen einzigen 80 Kbyte langen String und versucht diesen der Stringvariablen *a\$* zuzuweisen. Der Versuch schlägt fehl, da die maximale Länge eines VBA-Strings 64 Kbyte beträgt.

Mit *Input* vermeiden Sie diesen Fehler. Denn *Input* kümmert sich nicht weiter um Endemarkierungen, sondern liest einfach so viele Zeichen wie angegeben und übergibt diese Zeichen als Funktionswert.

```
Sub Kopieren
    Dim a As String

    Open "original" For Input As 1
    Open "kopie" For Output As 2

    '1.Zeichen
    a = Input(1, #1)
    Print #2, a;

    '2.Zeichen
    a = Input$(1, #1)
    Print #2, a;

    ...
    ...
End Sub
```

Listing 38.1:
Textdatei kopieren

Die Textdatei ORIGINAL wird im Lesemodus *Input* unter der Nummer 1 geöffnet, die neue Datei KOPIE, die die Kopie aufnehmen soll, im Schreibmodus *Output* unter der Nummer 2.

Die Anweisung *a = Input(1, #1)* liest genau ein Zeichen aus der Datei Nummer 1 ein, also das erste Zeichen des darin enthaltenen Textes, mit *Print #2, a;* wird es in die neue Datei KOPIE geschrieben.

Anschließend wird das zweite Zeichen aus Datei 1 gelesen und in Datei 2 geschrieben und so weiter.

Beachten Sie unbedingt das abschließende Semikolon! Dieses Semikolon ist enorm wichtig. Es unterdrückt den üblichen Abschluss einer *Print*-Anweisung mit der <CR><LF>-Kombination, denn dadurch würde folgende Datei entstehen:

Original: Test<CR><LF>

Kopie: T<CR><LF>e<CR><LF>s<CR><LF>t<CR><LF><CR><CR><LF><LF><CR><LF>

Jedem Zeichen würde eine <CR><LF>-Kombination folgen, auch den abschließenden Zeichen <CR> und <LF> selbst, denn diese werden von *Input* im Gegensatz zu den anderen Leseanweisungen wie ganz normale Zeichen behandelt.

Eof-Funktion

Das zeichenweise Lesen und Speichern soll solange wiederholt werden, bis das Ende der Originaldatei erreicht ist. Nur, wie erkennt das Programm das Dateieinde? Das Programm tut das mit der *Eof*-Funktion (*Eof* = End of File = Dateieinde).

```
v = Eof(Dateinummer)
```

Die *Eof*-Funktion testet, ob das Ende einer sequentiellen Datei erreicht ist. Dieses Dateieinde erkennt *Eof* an einem speziellen Kennzeichen, dem ASCII-Code 26, der sich am Ende einer sequentiellen Datei befindet. Erkennt die *Eof*-Funktion dieses Zeichen, übergibt sie als Funktionswert *True*, sonst *False*.

Um bis zum Erreichen des Dateieindes zu lesen, verwendet man üblicherweise folgende Konstruktion:

Listing 38.2: Dateieinde- Erkennung

```
While Eof(Dateinummer) = False
    Input #Dateinummer, Variable
    Debug.Print Variable
Wend
```

Die in der Schleife enthaltenen Leseanweisungen werden wiederholt, solange *Eof* den Wert *False* übergibt und das Dateieinde noch nicht erreicht ist. Das heißt, die Datei wird zeilenweise gelesen und im Direktfenster ausgegeben.

Wenn es sich bei der Datei nicht um eine ASCII-Textdatei handelt und mangels <CR><LF>-Kombinationen das Lesen von Textzeilen versagt, verwenden Sie einfach *Input* und lesen die Datei Zeichen für Zeichen (Abbildung 38.64).

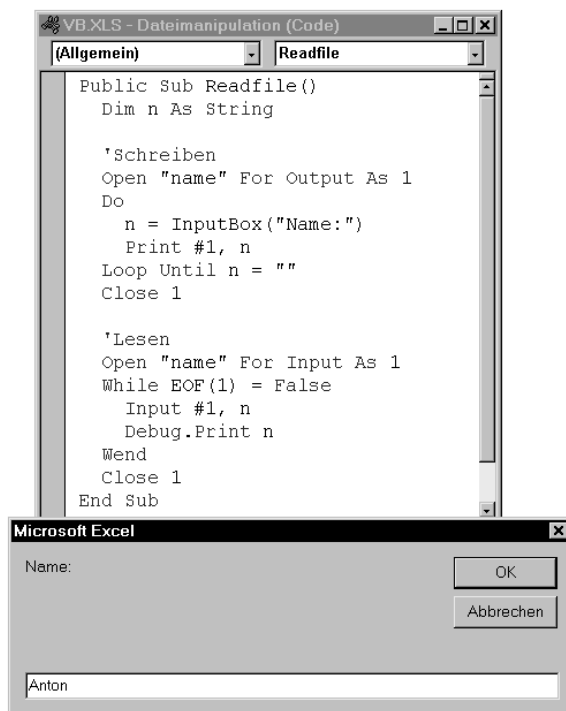


Abbildung 38.64:

Vb.xls, Modul
»Dateimanipulation«, Prozedur
»Readfile«

Rufen Sie sie vom Direktfenster aus mit *Call Readfile* auf, öffnet die Prozedur *Readfile* zunächst die Datei NAME im aktuellen Verzeichnis im Schreibmodus. Anschließend fordert sie Sie mit der Anweisung *InputBox* entsprechend der Abbildung immer wieder auf, einen Namen einzugeben und speichert Ihre Eingaben in der Datei.

Sie beenden die Eingabe, indem Sie einfach drücken, ohne etwas einzugeben. Danach öffnet das Programm die Datei im Lesemodus *Input*, liest die enthaltenen Namen bis zum Erreichen des Dateiendes und gibt sie im Direktfenster aus (Abbildung 38.65).

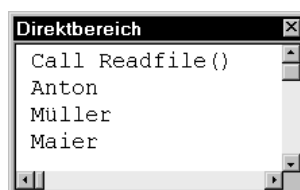


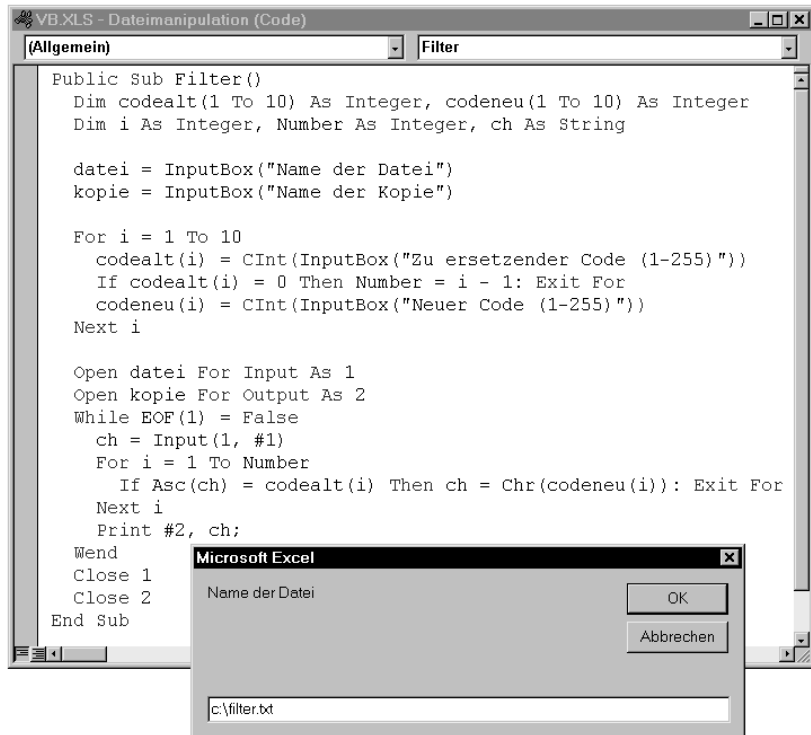
Abbildung 38.65:

Ausgabe der
Textdatei

Input und *Eof* sind Voraussetzungen für das folgende etwas anspruchsvollere Demoprogramm, das die Behandlung sequentieller Dateien abschließt. Es kopiert eine Datei zeichenweise und bietet darüber hinaus die Option, einzelne Zeichen gezielt durch andere zu ersetzen (Abbildung 38.66).

Abbildung 38.66:

Vb.xls, Modul
»Dateimanipulation«, Prozedur
»Filter«



Die Prozedur *Filter* fragt Sie zunächst nach dem Namen der zu kopierenden Datei und dem Namen der Dateikopie. Danach werden die zu ersetzenden Codes in einer *For*-Schleife abgefragt, maximal zehn Stück.

Um Ihnen die Möglichkeit zu geben, wirklich beliebige Codes zu ersetzen, nicht nur die auf der Tastatur enthaltenen Zeichen, sollen Sie nun nicht etwa ein Zeichen wie zum Beispiel »a« eingeben, sondern den zugehörigen ASCII-Code (»A«: 97). Da *InputBox* eine Zeichenkette übergibt, werden Ihre Eingaben anschließend mit der *Val*-Funktion in die benötigten numerischen Werte umgewandelt.

Wenn Sie einen Code ungleich 0 eingeben (*If codealt(i) = 0*), werden Sie nach dem Code jenes Zeichens gefragt, das das ursprüngliche Zeichen ersetzen soll. Geben Sie zum Beispiel 65 (ASCII-Code von »A«) ein, wenn Sie ein kleines »a« durch ein großes »A« ersetzen wollen.

Sie können die Codeabfrage jederzeit beenden, indem Sie auf die erste Frage 0 eingeben. Dann wird in *number* die Anzahl zu ersetzender Codes gespeichert (Schleifenvariable *i* minus 1) und die Schleife mit *Exit For* verlassen. Ein Beispiel für mögliche Eingaben sehen Sie hier:

Name der Datei: c:\filter.txt
Name der Kopie: c:\test.txt
Zu ersetzender Code (1-255): 97
Neuer Code (1-255): 65
Zu ersetzender Code (1-255): 101
Neuer Code (1-255): 69
Zu ersetzender Code (1-255): 0

Diese Testeingaben bedeuten, dass die in C:\ vorhandene Datei FILTER.TXT kopiert und ebenfalls in C:\ die Kopie mit dem Namen TEST.TXT erstellt werden soll.

Das Zeichen mit dem Code 97 (»a«) soll dabei durch das Zeichen mit dem Code 65 (»A«) ersetzt werden und das Zeichen mit dem Code 101 (»e«) durch 69 (»E«).

Die Eingabe »0« beendet die Schleife mit der Codeabfrage. Danach enthalten die einzelnen Variablen folgende Inhalte:

Variable	Inhalt
datei\$	"filter.txt"
kopie\$	"test.txt"
number%	2
codealt%(1)	97
codeneu%(1)	65
codealt%(2)	101
codeneu%(2)	69

Tabelle 38.3:
Variableninhalte

Die Datei mit dem Namen *datei* wird unter der Nummer 1 geöffnet, die neue Datei *kopie* erhält die Nummer 2. Die folgende *While*-Schleife testet mit *Eof(1)*, ob das Ende der Datei Nummer 1 erreicht ist. Wenn ja, wird die Schleife verlassen.

In der Schleife wird mit *ch = Input(1, #1)* das jeweils nächste Zeichen aus der Originaldatei gelesen. Eine *For*-Schleife prüft, ob der ASCII-Code dieses Zeichens mit einem der in dem Array *codealt()* gespeicherten Codes übereinstimmt (*If Asc(ch) = codealt(i)*).

Wenn ja, wird *ch* das Zeichen zugewiesen, das dem Code zugeordnet ist, den das zugehörige Element des Arrays *codeneu()* enthält, und die Schleife wird mit *Exit For* verlassen.

Anschließend wird das in *ch* enthaltene Zeichen in der Datei Nummer 2 gespeichert und zwar – abschließendes Semikolon nach *Print* – nur dieses eine Zeichen, ohne folgende <CR><LF>-Kombination!

Werden die Testeingaben auf die mit DATEI|TEXT SPEICHERN... als Textdatei gespeicherte Prozedur *Filter* selbst angewendet, ergibt sich folgende Datei-kopie (Abbildung 38.67):

Abbildung 38.67:
Erzeugte Dateikopie

```
test.txt - Editor
Datei Bearbeiten Suchen ?

Public Sub Filter()
Dim codEAlt(1 To 10) As Integer, codEnEu(1 To 10) As Integer
Dim i As Integer, NumbEr As Integer, ch As String

dAtEi = InputBox("Name dEr dAtEi")
kopIE = InputBox("Name dEr KopIE")

For i = 1 To 10
codEAlt(i) = CInt(InputBox("Zu ErsEtzEndEr CodE (1-255)"))
If codEAlt(i) = 0 Then NumbEr = i - 1: Exit For
codEnEu(i) = CInt(InputBox("NEuEr CodE (1-255)"))
Next i

OpEn dAtEi For Input As 1
OpEn kopIE For Output As 2
While EOF(1) = False
ch = Input(1, #1)
For i = 1 To NumbEr
If Asc(ch) = codEAlt(i) Then ch = Chr(codEnEu(i)): Exit For
Next i
Print #2, ch;
WEnd
Close 1
Close 2
End Sub
```

38.8.4 Direktzugriffsdateien

Bei der sequentiellen Speicherung werden Daten unterschiedlichster Art und Länge unmittelbar nebeneinander in einer Datei gespeichert. Die Unterscheidung erfolgt über ein Trennzeichen, ein Komma oder eine <CR><LF>-Kombination.

Eine solche Trennung ist bei einer Direktzugriffsdatei nicht nötig, da jeder Datensatz in einem eigenen Block fester Länge gespeichert wird. Jeder Satz besitzt eine eigene Satz- oder Recordnummer, über die ein direkter Zugriff auf den betreffenden Block der Datei möglich ist (Abbildung 38.68).

Abbildung 38.68:
Direktzugriffs-dateien

+-----+-----+-----+		
Mai, Hans.....	Bauer, Otto.....	Siepholz, Manfred...
+-----+-----+-----+		
+-----+-----+-----+		
Record 1	Record 2	Record 3

Eine Direktzugriffsdatei geht übrigens nicht ganz so verschwenderisch mit dem verfügbaren Speicherplatz um, wie es scheint, denn Zahlen werden im rechnerinternen komprimierten Format gespeichert. Das heißt jede Zahl vom Typ *Integer* belegt genau zwei Byte, jede Fließkommazahl einfacher Genauigkeit (*Single*) vier Byte. Auch eine Zahl wie 1304.7125 mit neun Ziffern würde in einer sequentiellen Datei neun Byte belegen.

Auch eine Direktzugriffsdatei muss zunächst geöffnet werden und zwar unter Angabe der Satzlänge (Recordlänge):

Datei öffnen

```
Open "test.dat" For Random As 1 Len=256
```

Diese Anweisung öffnet eine Direktzugriffsdatei (Modus *Random*) namens TEST.DAT mit einer Blocklänge von 256 Byte. Die Anweisung *Put* speichert Daten in der Datei:

Daten schreiben

```
Put [#]Dateinummer [, [Satznummer] [, Variable]]
```

Dateinummer kennen Sie. *Satznummer* ist eine Zahl zwischen 1 und 2.147.483.647, die die Nummer des Blocks der Datei angibt, in dem die Daten gespeichert werden sollen, zum Beispiel die Adressen einer Adressdatei. Bei einer neuen Datei verwendet man normalerweise die Satznummer 1 für die erste zu speichernde Adresse, 2 für die zweite und so weiter.

Variable ist eine beliebige Variable. Beachten Sie, dass Sie tatsächlich eine Variable angeben müssen, nicht wie sonst einen beliebigen Ausdruck, etwa *Put #1, »Hallo«*. Die Anweisung *Get* liest Daten aus einem Block einer Direktzugriffsdatei und speichert sie in der angegebenen Variablen:

Daten lesen

```
Get [#]Dateinummer [, [Satznummer] [, Variable]]
```

Ist eine Direktzugriffsdatei geöffnet, können Sie abwechselnd mit *Get* und *Put* Daten lesen und schreiben, ohne wie bei sequentiellen Dateien auf eine Richtung beschränkt zu sein. Nach der Ausführung der Lese-/Schreibanweisungen schließen Sie die Datei mit *Close* (Abbildung 38.69).

Datei schließen

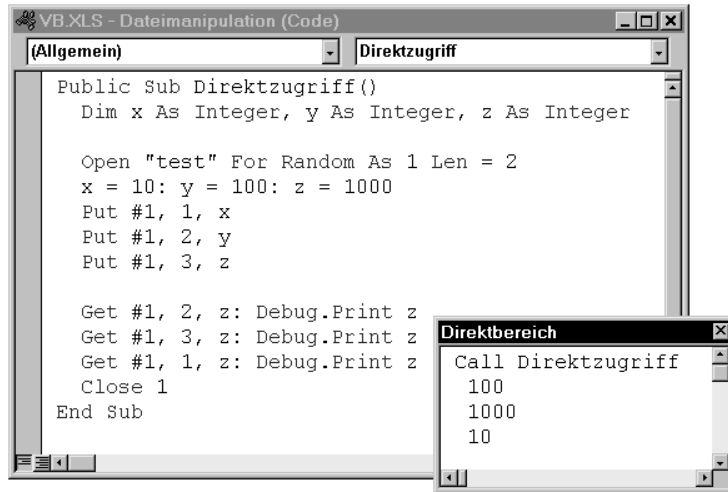
Direktzugriff schreibt die Inhalte dreier *Integer*-Variablen in die Direktzugriffsdatei TEST.

Jede wird in einem eigenen Record gespeichert. Die Recordlänge muss daher mindestens zwei Byte betragen (*Len* = 2). Längere Records sind zwar möglich, führen jedoch zur Verschwendung von Speicherplatz auf der Diskette/Platte.

x wird in Record 1 gespeichert, *y* in Record 2 und *z* in Record 3.

Abbildung 38.69:

Vb.xls, Modul
»Dateimanipulation«, Prozedur
»Direktzugriff«



Um zu zeigen, dass wirklich ein direkter Zugriff möglich ist, werden die Daten anschließend nicht vom Dateibeginn an gelesen, sondern in sehr wahlloser Reihenfolge.

Mit *Get #1, 2, z* wird zuerst die in Record 2 gespeicherte Integerzahl gelesen, in *z* gespeichert und anschließend mit *Debug.Print* ausgegeben. Anschließend greift *Get #1, 3, z* auf den dritten Record zu und erst zuletzt wird die im ersten Record enthaltene Zahl gelesen.

Bei umfangreichen Datenmengen haben Direktzugriffsdateien enorme Vorteile gegenüber sequentiellen Dateien, weshalb auch VBA diesen Dateityp benutzt. Wenn Sie in einer Datei mit 1000 Adressen die Recordnummer von Herrn Müller kennen, kann auf dessen Adresse direkt zugegriffen werden, ohne wie bei einer sequentiellen Datei erst alle vorhergehenden 999 Adressen lesen zu müssen. Und wenn sich seine Telefonnummer ändert, überschreiben Sie einfach die in Record 1000 enthaltene alte Adresse durch die geänderte.

Ob Sie nun eine Adressdatei verwalten, eine Artikeldatei oder Ähnliches, auf alle Fälle handelt es sich immer um strukturierte Daten, die den gleichen Aufbau haben. Eine Adresse (=Datensatz) besteht immer aus den gleichen Teilen (=Felder).

Ähnlichkeiten mit Verbundvariablen sind offensichtlich. Denn auch Verbundvariablen bestehen aus einzelnen Komponenten, die für jede Variable des gleichen Typs identisch sind und tatsächlich sind Verbundvariablen untrennbar mit Direktzugriffsdateien verbunden.

In der Praxis deklariert man für die Datensorte, die gespeichert werden soll, mit *Type* einen eigenen Datentyp, zum Beispiel den Typ *Adresstyp*, der die einzelnen Komponenten einer Adresse aufnimmt:

```
Type Adresstyp
    vorname As String * 10
    nachname As String * 15
    plz      As Integer
    wohnort  As String * 20
End Type
```

Listing 38.3:

Type

```
Dim adresse As Adresstyp
```

Um eine in der Variablen *adresse* enthaltene Adresse zu speichern, genügt eine *Put*-Anweisung, in der Sie die gewünschte Recordnummer und die Variable angeben. Da in jedem Record genau eine Variable dieses Typs gespeichert wird, ergibt sich die geforderte Recordlänge der Datei aus dem Speicherbedarf der Variablen.

Das Addieren der einzelnen Komponenten können Sie sich allerdings ersparen. Die *Len*-Funktion ermittelt nicht nur die Länge einer Stringvariablen, sondern auch den Speicherbedarf eines mit *Type* deklarierten Datentyps. Mit der Anweisung

Len-Funktion

```
Open "test" For Random As 1 Len = Len(adresse)
```

wird die Recordlänge der Datei TEST so gewählt, dass sie genau der Länge der Variablen *adresse* entspricht. Nun können die ersten Adressen in der Datei gespeichert werden:

```
adresse.vorname = "Willi"
adresse.nachname = "Maier"
adresse.plz = 68000
adresse.wohnort = "Mannheim"
Put #1, 1, adresse          '1.Adresse speichern

adresse.vorname = "Otto"
adresse.nachname = "Müller"
adresse.plz = 80000
adresse.wohnort = "München"
Put #1, 2, adresse          '2.Adresse speichern
```

Listing 38.4:

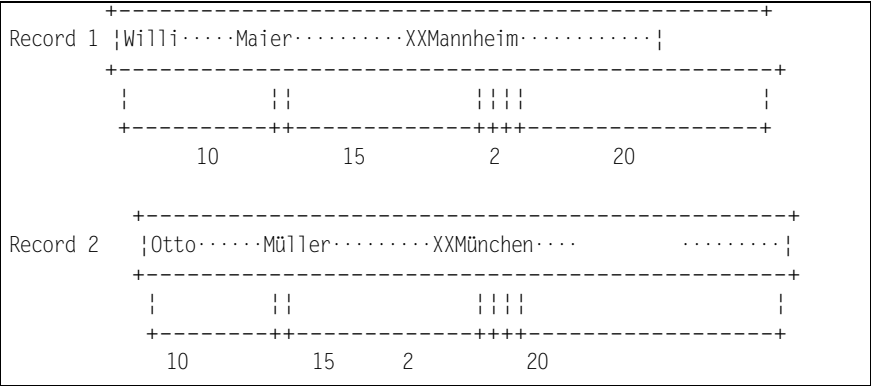
Adresskomponenten definieren

Die folgende Abbildung zeigt, in welcher Form die beiden Adressen gespeichert werden.

Beachten Sie, dass die Stringkomponenten von *adresse* Variablen fester Länge sind und die darin enthaltenen Zeichenketten daher mit Leerzeichen aufgefüllt werden.

Die Zeichen »XX« sollen übrigens die Integerzahlen 68000 bzw. 80000 darstellen, die wie erwähnt im komprimierten 2-Byte-Format gespeichert wird, was schlecht darstellbar ist (Abbildung 38.70).

Abbildung 38.70:
Speicherungs-
schema



Sicher wollen Sie nun ein praktisches Beispiel zum Umgang mit Direktzugriffsdateien, ein Demoprogramm. Die weiter unten abgebildete Prozedur *Direktzugriffsdatei* fordert Sie auf, eine Adresse einzugeben, bestehend aus einem vollständigen Namen und der zugehörigen Anschrift.

Zuerst geben Sie in einem Dialogfeld den Namen ein und drücken (Abbildung 38.71).

Abbildung 38.71:
Abfrage des
Namens



Das Dialogfeld verschwindet und ein weiteres Dialogfeld erscheint, das nach dem zugehörigen Wohnort fragt. Geben Sie ihn ein und drücken Sie wieder . Das erste Dialogfeld erscheint wieder, Sie geben einen weiteren Namen ein, danach erneut den zugehörigen Wohnort und so weiter.

Haben Sie genug Adressen eingegeben, beantworten Sie entweder die Frage nach dem Namen oder die folgende Frage nach dem Wohnort, indem Sie einfach nur drücken.

Sie könnten beispielsweise folgende Adressen eingeben:

Hans Maier
67000 Ludwigshafen
Otto Müller
80000 München

In diesem Beispiel werden zwei Adressen verwendet, die die Prozedur in den Records Nummer 1 und 2 der Direktzugriffsdatei XYZ speichert und die Eingabe anschließend beendet.

Danach fragt die Prozedur nach einer Recordnummer. Geben Sie eine 1 ein, wird der erste Datensatz eingelesen und angezeigt, bei Eingabe einer 2 entsprechend der zweite Datensatz und so weiter.

Durch Eingabe der Zahl 0 wird das Programm beendet.

Sie können direkt auf jeden der zuvor gespeicherten Datensätze zugreifen, ohne sich an eine bestimmte Reihenfolge halten zu müssen, wie die folgende Abbildung zeigt, in der zuerst der zweite und danach der erste Datensatz eingelesen wurde (Abbildung 38.72).

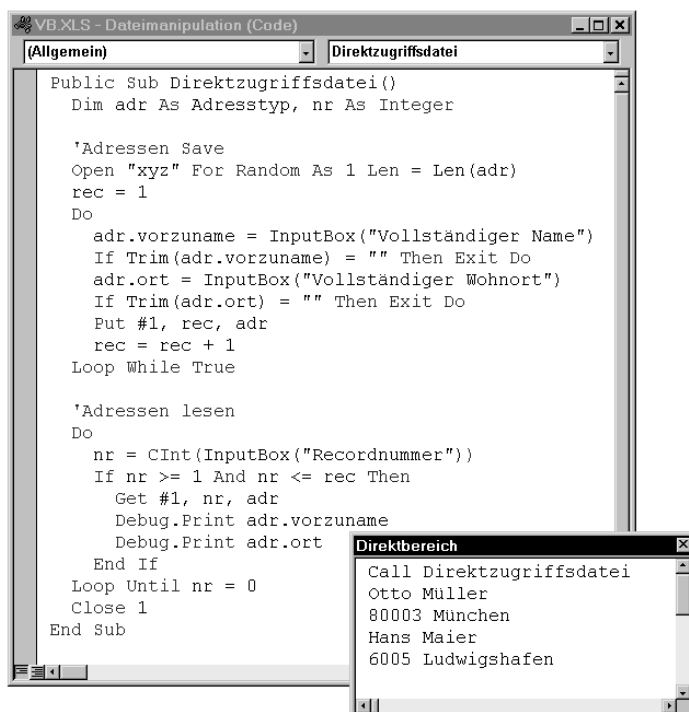


Abbildung 38.72:
Vb.xls, Modul
»Dateimanipulation«, Prozedur
»Direktzugriffsdatei«

Die Prozedur selbst enthält mehrere Kleinigkeiten, die Sie beachten sollten: zunächst die Variable *rec*, die bestimmt, in welchem Record eine neue Adresse gespeichert wird.

rec wird vor der Schleife mit 1 initialisiert und nach dem Speichern einer neuen Adresse jeweils um 1 erhöht, da jede neue Adresse im jeweils nächsten Record gespeichert wird. Nach dem Verlassen der Eingabeschleife wird *rec* um 1 vermindert und enthält nun die Gesamtzahl der in der Datei enthaltenen Records (Adressen).

Wenn Sie den Eingabeteil ohne Eingabe verlassen, beginnt eine *Do*-Schleife, in der Sie zunächst nach einer Recordnummer gefragt werden.

Danach wird geprüft, ob diese in *nr* gespeicherte Recordnummer in einem sinnvollen Bereich liegt, also zwischen 1 und *rec*, dem letzten in der Datei enthaltenen Record, um zu verhindern, dass eventuell über das Dateiende hinaus zuzugriffen wird. Ist diese Prüfung positiv, wird die im betreffenden Record enthaltene Adresse gelesen und auf dem Bildschirm angezeigt.

Dieser Vorgang wiederholt sich, bis Sie als Recordnummer 0 eingeben. Sie können sich daher – im Gegensatz zu sequentiellen Dateien – jede in der Datei enthaltene Adresse anzeigen lassen, ohne eine bestimmte Reihenfolge einhalten zu müssen.

**Logische Daten-
strukturen**

Direktzugriffsdatei demonstriert zwar sehr schön den direkten Zugriff auf jeden Datensatz einer Direktzugriffsdatei, aber wahrscheinlich fragen Sie sich, ob es nicht doch ein wenig komfortabler geht, denn wahrscheinlich wollen Sie beim Umgang mit einer Dateiverwaltung nicht ständig eine Liste mit Recordnummern und zugehörigen Adressen neben dem Rechner liegen haben.

Komfortablere Suchmöglichkeiten, zum Beispiel über einen Vor- oder Zunamen, hängen von der gewählten logischen Struktur Ihrer Datei ab, weniger vom Dateityp. Die Direktzugriffsdatei ist nur das grundlegende Handwerkszeug, mit dessen Hilfe komfortable logische Datenstrukturen verwirklicht werden. Darauf einzugehen würde allerdings den Rahmen dieses Buches sprengen.

Beim Umgang mit Direktzugriffsdateien sind einige Besonderheiten zu berücksichtigen. Sicher fragen Sie sich, was passiert, wenn man in einer neuen Datei die Records nicht der Reihe nach beschreibt (1, 2, 3, ...), sondern zum Beispiel zuerst Record Nummer 100, etwa so:

Listing 38.5: Open "test" For Random As 1 Len = 200
100 Records Put #1, 100, x%
anlegen Close 1

Dieses Programm öffnet eine Direktzugriffsdatei namens TEST mit der Recordlänge 200 und schreibt in Record Nummer 100 die *Integer*-Variable *x%* (Inhalt 0).

Lof-Funktion

Das Ergebnis ist recht interessant. Die Datei TEST wird angelegt, genauer: 100 Records á 200 Byte werden angelegt, die bis auf Record Nummer 100 leer sind. Und auf der Diskette/Platte befindet sich nun eine rund 20 Kbyte große leere Datei TEST. Die Funktion *Lof* (*Lof* = Length of File = Dateilänge) übergibt die Länge einer Datei in Byte:

```
v = Lof(Dateinummer)
```

Diese Funktion können Sie auf jede Datei anwenden. Wirklich nützlich ist sie jedoch vor allem bei der Behandlung von Direktzugriffsdateien, um das Dateiende zu erkennen. *Eof* kann nur auf sequentielle Dateien angewendet werden. *Lof* bietet dann folgenden Ausweg: Oft wissen Sie zwar nicht, wie viele Records in einer Datei enthalten sind, kennen dafür jedoch die Recordlänge. Dann ergibt sich die Recordanzahl aus der mit *Lof* ermittelten Dateilänge geteilt durch die Recordlänge:

```
Recordanzahl = Lof(Dateinummer) / Recordlänge
```

Zur praktischen Anwendung nehmen wir an, dass sich auf der Diskette eine Adressdatei namens ADRESSEN.DAT befindet, die Sie irgendwann einmal erstellt haben.

Sie wissen nicht mehr, wie viele Records die Datei enthält, nur noch, dass sie Adressen vom Typ *Adresstyp* enthält. Nun wollen Sie den Inhalt der gesamten Datei in ein Array einlesen:

```
Type Adresstyp
    vorzuname As String * 40
    ort        As String * 40
End Type
```

Listing 38.6:
Datei unbekannter
Größe einlesen

```
Dim adr(1 To 1000) As Adresstyp
Dim RecLength As Integer, i As Integer

RecLength = Len(adr)

Open "adressen.dat" For Random As 1 Len = RecLength%
For i = 1 To Lof(1) / RecLength
    Get #1, i, adr(i)
Next i
Close 1
```

Das Array *adr()* ist vom Typ *Adresstyp*: In der Schleife wird der Reihe nach auf die Records 1, 2, 3 usw. positioniert. Die Nummer des letzten Records ergibt sich aus dem Ausdruck $Lof(1) / RecLength$, Dateilänge geteilt durch die Recordlänge.

Zur Positionierung auf einen Record benutzt VBA einen Satzzeiger, den Sie sich als Variable vorstellen können, die die aktuelle Recordnummer enthält, also die Nummer des Records, auf den gerade positioniert wird.

Den Satzzeiger können Sie auf unterschiedliche Weise beeinflussen, zum einen durch die Angabe der gewünschten Recordnummer in einer *Put*- oder *Get*-Anweisung. Der aktuelle Stand des Satzzeigers ändert sich jedoch auch ohne Angabe der Recordnummer, die ja optional ist, denn jede *Put*- oder *Get*-Anweisung erhöht den Satzzeiger um 1:

Listing 38.7:
Sequentielle Satz-
zeigerpositionierung

```
Get #1, 1, adr 'Record 1 lesen
Get #1, , adr 'Record 2 lesen
Get #1, , adr 'Record 3 lesen
```

Um nacheinander die Records 1, 2, 3 usw. einer Direktzugriffsdatei zu lesen oder zu beschreiben, genügt es, beim ersten Zugriff die Recordnummer 1 anzugeben und bei allen folgenden Zugriffen auf die Recordnummer zu verzichten.

Seek-Anweisung

Mit der Anweisung *Seek* können Sie den Satzzeiger jedoch auch auf einen gewünschten Record setzen:

```
Seek [#]Dateinummer, Position
```

Position ist die Nummer des gewünschten Records. Um zum Beispiel Record 5 zu lesen, können Sie folgende Kombination aus *Seek* und *Get* verwenden:

```
Seek #1, 5
Get #1, , adr
```

In der *Get*-Anweisung wird auf die Angabe der Recordnummer verzichtet, da ja bereits mit *Seek* auf den gewünschten Record positioniert wurde. Interessanter als bei Direktzugriffsdateien ist *Seek* bei sequentiellen und binären Dateien, wo *Position* keine Recordnummer ist, sondern die Nummer eines beliebigen Bytes innerhalb der Datei, auf das Sie lesend oder schreibend zugreifen wollen.

Seek-Funktion

Im Gegensatz zur Anweisung *Seek* beeinflusst die Funktion (!) *Seek* den Satzzeiger nicht, sondern gibt seinen aktuellen Inhalt aus, die Nummer des Records, auf den gerade positioniert ist und auf den nun mit einer *Put*- oder *Get*-Anweisung ohne Angabe der Recordnummer zugegriffen würde.

Die Funktion *Loc* übergibt die Nummer des Records, auf den zuletzt lesend oder schreibend zugegriffen wurde:

Loc-Funktion

```
v = Seek(Dateinummer) 'aktuelle Recordnummer  
v = Loc(Dateinummer) 'Record, auf den zuletzt zugegriffen wurde
```

Ein Beispiel:

```
Put #1, 5, adr      'Adresse in Record 5 speichern  
Debug.Print Loc(1) 'gibt 5 aus  
Debug.Print Seek(1) 'gibt 6 aus  
Put #1, , adr       'Adresse in Record 6 speichern
```

Listing 38.8:
Seek und Loc

Zuerst wird mit *Put* eine Adresse in Record Nummer 5 gespeichert. Verständlicherweise übergibt *Loc* anschließend 5 als Nummer des Records, auf den zuletzt zugegriffen wurde, und *Seek* übergibt 6 als Nummer des aktuellen Records, denn jede *Put*- oder *Get*-Anweisung erhöht den Satzzeiger um 1. Die folgende *Put#*-Anweisung ohne Recordnummer wird daher auf Record Nummer 6 zugreifen.

Übrigens können Sie nicht nur die *Seek*-Anweisung, sondern auch die *Seek*- und die *Loc*-Funktion auf *alle* Dateitypen anwenden, auf sequentielle und Binärdateien dann eben bytebezogen statt recordbezogen. Das heißt, *Seek* und *Loc* übergeben die Nummer des zuletzt beschriebenen Bytes (*Loc*) bzw. die Nummer des Bytes, auf das momentan positioniert ist (*Seek*).

38.8.5 Binärdateien

Binärdateien gestatten wie Direktzugriffsdateien einen unmittelbaren Zugriff auf alle in einer Datei enthaltenen Daten unter Angabe der Position.

Der Unterschied zur Direktzugriffsdatei ist die Art der gespeicherten Daten. Binärer Zugriff behandelt eine Datei nicht als Folge unstrukturierter Zeilen (sequentielle Datei) oder strukturierter Sätze (Direktzugriffsdatei), sondern als Folge einzelner Bytes. Und jedes Byte kann gezielt manipuliert werden.

Daher eignet sich der binäre Zugriff beispielsweise zur Manipulation von Programmdateien. Zum Beispiel können Sie in der Datei EXCEL.EXE mit binärem Zugriff einzelne Bytes verändern, darin enthaltene Maschinensprachebefehle – was ich Ihnen jedoch nicht empfehlen würde, wenn Sie Excel weiterhin benutzen wollen!

Ein anderes Beispiel: Gehen wir davon aus, Sie haben eine Direktzugriffsdatei ADRESSEN.DAT mit einer Recordlänge von 100 Byte. Und Sie wollen das dritte Zeichen in Record Nummer 31 verändern. Wenn Sie ADRESSEN.DAT als Binärdatei öffnen, können Sie dieses Byte mit der Nummer 3003 (30

Records á 100 Byte plus 3 Byte) beliebig manipulieren, ohne den restlichen Inhalt des Records zu beeinflussen.

Der Umgang mit Binärdateien ähnelt dem Umgang mit Direktzugriffsdateien.

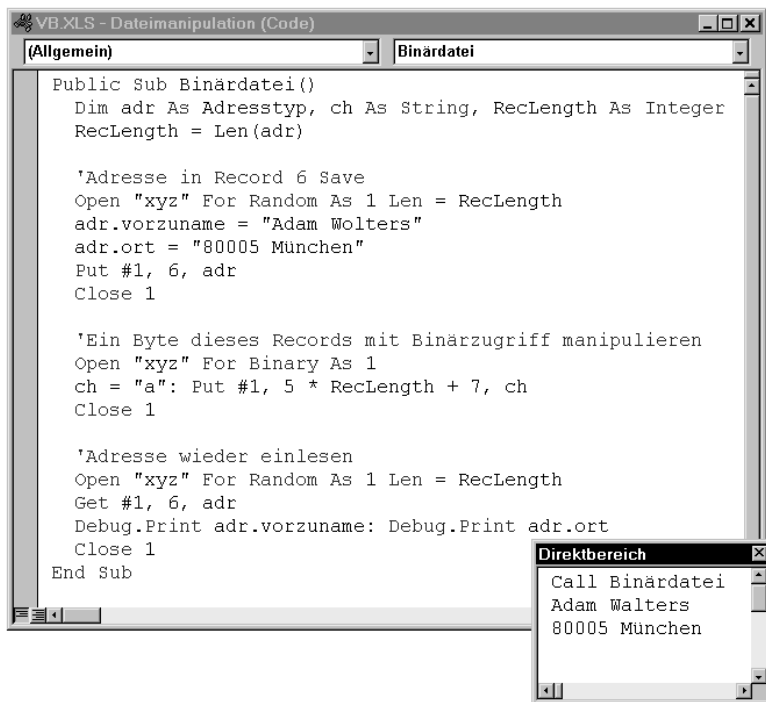
Mit *Get* oder *Put* werden Daten gelesen und geschrieben, allerdings nicht unter Angabe der Record-, sondern der Bytenummer. Die Länge der angegebenen Variablen bestimmt, wie viele Bytes *Get* aus der Datei liest (*Integer*: 2 Byte, *String* * 1: 1 Byte usw.) bzw. wie viele Bytes *Put* in die Datei schreibt.

Zusätzlich können Sie *Input* zum Lesen von Daten verwenden, mit der *Eof*-Funktion das Dateende und mit *Lof* die Dateilänge abfragen.

Seek und *Loc* besitzen die gleichen Funktionen wie beim Direktzugriff, übergeben jedoch Bytenummern statt Recordnummern (Abbildung 38.73).

Abbildung 38.73:

Vb.xls, Modul
»Dateimanipulation«, Prozedur
»Binärdatei«



Diese Prozedur öffnet eine Datei namens XYZ im Direktzugriff und speichert in Record 6 eine Adresse. Anschließend wird mit binärem Zugriff ein Zeichen der Adresse manipuliert. Aus »Wolters« soll »Walters« werden. Zuerst wird die Direktzugriffsdatei geschlossen und die Datei im Binärmodus geöffnet:


```
Open "xyz" For Binary As 1
```

Die Angabe einer Recordlänge ist überflüssig, da es bei dieser Zugriffsform keine Records gibt. Nun wird in *ch* ein Zeichen gespeichert, »a«, und dieses Zeichen an der Position $5 * \text{RecLength} + 7$ in die Datei geschrieben.

Die folgende Abbildung zeigt, wie dieser Ausdruck zustande kommt (Abbildung 38.74):

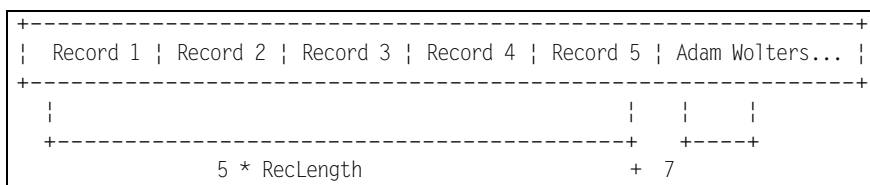


Abbildung 38.74:
Record gezielt
manipulieren

Das Zeichen »a« überschreibt das »o« in »Wolters«. Anschließend wird die Datei geschlossen, im Direktmodus geöffnet und der manipulierte Record 6 zur Kontrolle eingelesen, der nun den Namen »Walters« enthält.

Um den nicht allzu großen Unterschied zwischen dem binären und dem Direktzugriff zu verdeutlichen: Der binäre Zugriff befindet sich eine Ebene unter dem Direktzugriff, ist primitiver. Sie können theoretisch Folgen strukturierter Daten auch mit binärem Zugriff verwalten, zum Beispiel eine Adresstyp. Wenn der Datentyp *Adresstyp* eine Länge von 80 Byte hat, speichern Sie die erste Adresse mit

```
Put #1, 1, adr(1)
```

ab Byte Nummer 1 in der Datei. Mit

```
Put #1, 81, adr(2)
```

die zweite Adresse ab Byte Nummer 81, eine dritte entsprechend ab Byte Nummer 162 und so weiter.

Sie sehen, der Direktzugriff ist eigentlich nur eine komfortablere Version des Binärzugriffs, der den Umgang mit strukturierten Daten erleichtert. Denn zweifellos ist es sehr angenehm, dass jedem Datensatz eine eigene Recordnummer zugeordnet ist, jedenfalls angenehmer, als die ständige Positionsrechnung per Hand über eine Bytenummer, die der binäre Zugriff erfordert.

Teil 7 Anhang

A.1:	Die Demos auf der CD	1065
A.2:	Dateisuche	1066
A.3:	Der Druck-Manager	1068
A.4:	Rechtschreibprüfung	1070
A.5:	AutoKorrektur	1073
A.6:	Add-Ins	1074
A.7:	Excel individuell konfigurieren	1076

A Anhang

A.1 Die Demos auf der CD

Die CD zum Buch enthält praktisch alle Demos, die in diesem Buch entwickelt werden. Die verschiedenen Entwicklungsstufen werden dabei im betreffenden Kapitel anhand von Abbildungen dokumentiert.

Lautet die Bildunterschrift einer solchen Abbildung beispielsweise »EDIT.XLS, Tabelle5«, dann erkennen Sie daran, dass sich auf der Begleit-CD eine Arbeitsmappe EDIT.XLS befindet und »Tabelle5« dieser Mappe den abgebildeten Zustand enthält, sodass Sie diese Mappe jederzeit öffnen und darin die betreffende Tabelle aktivieren können, um sich den abgebildeten Tabellenzustand näher anzuschauen.

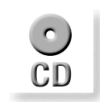
Dazu müssen Sie die Demos auf Ihrer Festplatte installieren. Die Dateien wurden mit einem Komprimierungsprogramm gepackt. Als Resultat befindet sich auf der CD eine einzige Datei DEMOS.EXE, mit der Sie zunächst nichts anfangen können. Zuerst müssen die darin enthaltenen Dateien entkomprimiert werden und zwar so:

Installieren

- ➔ Vergewissern Sie sich, dass auf Ihrer Festplatte noch gut *ein Megabyte* an freiem Speicherplatz vorhanden ist.
- ➔ Kopieren Sie DEMOS.EXE in irgendein Verzeichnis Ihrer Festplatte.
- ➔ Rufen Sie DEMOS.EXE auf.

Die Datei entkomprimiert sich nach dem Aufruf selbstständig. Anschließend befinden sich im betreffenden Verzeichnis die Original-Demodateien. DEMOS.EXE wird nicht mehr benötigt und kann gelöscht werden.

Eventuell finden Sie die Demodateien bereits in entpackter Form auf der Begleit-CD vor, statt DEMOS.EXE also bereits die einzelnen darin enthaltenen Dateien. Das Entpacken haben Sie sich in diesem Fall gespart und müssen nur noch die jeweilige interessierende Datei auf Ihre Festplatte kopieren.



Dateien auf CD sind jedoch schreibgeschützt – und daran ändert sich auch nach dem Kopieren auf die Festplatte nichts und daher sollten Sie den Schreibschutz der betreffenden Datei aufheben:

- ➔ Klicken Sie mit der rechten Maustaste auf die (auf die Festplatte kopierte) Datei.
- ➔ Wählen Sie im Kontextmenü den Befehl **EIGENSCHAFTEN**.
- ➔ Deaktivieren Sie im Register *Allgemein* das Kontrollkästchen *Schreibgeschützt* und klicken Sie auf **OK**.

Sie können mehrere Dateien auf einmal selektieren und auf die beschriebene Art und Weise den Schreibschutz all dieser Dateien gleichzeitig aufheben.



Während Sie dieses Buch durcharbeiten, werden Sie beim Experimentieren ziemlich sicher auch die eine oder andere Demodatei verändern oder gar löschen. Durch erneutes Ausführen des beschriebenen Vorgangs können Sie jederzeit wieder die Originaldatei auf Ihre Festplatte zaubern.



Für so manchen Leser dürfte mein C-Programm DM.EXE (»DatenManipulation«) nützlich sein, das ich ursprünglich für eigene Bedürfnisse entwickelte (Aktienanalyse). Es eignet sich hervorragend dazu, riesige und ständig wechselnde Datenmengen (Beispiel: Dutzende von über Jahre hinweg verfolgten Aktienkursverläufen), die sich mit Excel nicht mehr vernünftig manuell oder per VBA-Programm aufbereiten lassen, mit einer eigenen Befehlssprache batchgesteuert zu manipulieren und zur Weiterverarbeitung für Excel oder – mein eigener Spezialfall – für das neuronale Netzwerkprogramm BrainMaker aufzubereiten.

DM.EXE wird nirgendwo in diesem Buch erläutert, da es nun mal nicht direkt zu Excel gehört. Dafür befindet sich in der gepackten Datei DEMOS.EXE jedoch eine ASCII-Datei DM.DOC, die alle Features des Programms bis ins Detail erläutert.



Ich entwickle in diesem Buch Schritt für Schritt eine kleine Buchführung, die beispielsweise für Selbstständige oder Kleinstbetriebe interessant sein dürfte. Die Datei BUCHFRGR.XLT stellt eine Weiterentwicklung dieser Buchführung dar. Der Umgang damit wird in einem eigenen Kapitel im Anhang erläutert.

A.2 Dateisuche



Der Befehl **DATEI|SUCHEN...** öffnet die Suchabteilung des Aufgabenbereichs. Es gibt sie in zwei Versionen: einer einfachen Suche nach Texten, die in Dokumenten enthalten sind, bei der Sie auswählen können, wo nach Dateien gesucht werden soll, die diesen Text enthalten und welche Dateity-

pen dabei berücksichtigt werden sollen, und einer komplexeren Version, die ich nun besprechen werde und die das Gleiche ermöglicht und darüber hinaus zusätzliche Möglichkeiten bietet. Zwischen den beiden Versionen schalten Sie mit *Einfache Suche* bzw. *Weitere Suchoptionen* um (Abbildung A.1).

Weitere Suchoptionen

Suchen nach:

Eigenschaft: Text oder Eigenschaft

Bedingung: enthält

Wert:

☐ Und ☐ Oder

Hinzufügen Entfernen Alle entfernen

Suchen Wiederherstellen

Weitere Suchoptionen:

Suchen in: Ausgewählte Speicherorte

Ergebnisse entsprechen: Ausgewählte Dateitypen

Siehe auch

Einfache Suche

In diesem Dokument suchen...

Abbildung A.1:
Dateisuche

Im *Dateiauswahl*-Dialogfeld öffnet der Befehl *Suchen...* des *Extras*-Symbols ein Dialogfeld mit zwei Registern *Grundlegend* und *Erweiterte Optionen*, die exakt die gleichen Möglichkeiten bieten. Der einzige Unterschied besteht in der optischen Darstellung als normales Dialogfeld statt als Aufgabenbereich.

Extras

Gehen wir davon aus, dass Sie im Aufgabenbereich auf *Weitere Suchoptionen* klicken, um die komplexere Suchvariante zu aktivieren. Zur Definition der Kriterien, die die anzuzeigenden Dateien erfüllen müssen, verwenden Sie vor allem die Felder *Eigenschaft*, *Bedingung* und *Wert*.

Zunächst selektieren Sie im Feld *Eigenschaft*, um welche Dateieigenschaft es beim aktuellen Suchkriterium geht: um einen in der Datei enthaltenen Text, um den Dateinamen, das Erstellungsdatum der Datei, die Seitenanzahl (v.a. in Word-Dokumenten) etc.

Anschließend wählen Sie im Feld *Bedingung* eine Vergleichsbedingung für dieses Kriterium und geben danach im Feld *Wert* den Vergleichswert ein.

Angenommen, Sie suchen alle Dateien, deren Dateiname unter anderem die Zeichenkette »test« enthält (»Testen«, »Ein Test«, »Test5.xls« etc.). Dazu wählen Sie unter *Eigenschaft* das Kriterium »Dateiname«, unter *Bedingung* die Vergleichsbedingung »enthält« und geben im Feld *Wert* den Vergleichswert »test« ein.

Danach klicken Sie auf *Hinzufügen*, um diese Suchbedingung in die Kriterienliste einzufügen.

Nachdem die Definition der Suchbedingung abgeschlossen ist, können Sie unter *Suchen in* angeben, welche Ordner durchsucht werden sollen, wobei Sie auch Webordner einbeziehen können, und unter *Ergebnisse entsprechen* die Dateitypen auswählen, die durchsucht werden sollen.

Suchen führt die Suche aus und zeigt die gefundenen Dateien an. Klicken Sie auf eine davon, erscheint ein Listenpfeil, der ein Kontextmenü öffnet, dessen Befehle Ihnen unter anderem ermöglichen, die betreffende Datei im zugehörigen Programm zu öffnen.

Ändern blendet wieder die Suchoptionen ein, sodass Sie Ihre Suchkriterien editieren und danach eine weitere Suche starten können.



Sie können mehrere Suchkriterien definieren und miteinander kombinieren, um so im Extremfall für jede Eigenschaft ein eigenes Kriterium festzulegen.

Nachdem Sie ein weiteres Kriterium definiert haben und bevor Sie es mit *Hinzufügen* zusätzlich in die Kriterienliste aufnehmen, müssen Sie sich allerdings entscheiden, ob es mit *Und* oder mit *Oder* mit den restlichen Kriterien verknüpft werden soll.

Und bedeutet, dass nur Dateien gefunden werden, die allen Kriterien genügen. *Oder* bedeutet, dass Dateien gefunden werden, die mindestens eines der verwendeten Kriterien erfüllen.

Entfernen entfernt die momentan in der Kriterienliste selektierte Bedingung aus dieser, *Alle entfernen* entfernt alle Kriterien und *Wiederherstellen* stellt zuvor gelöschte Bedingungen wieder her.

A.3 Der Druck-Manager

Für den Ausdruck ist der Druck-Manager zuständig, ein Windows-Programm, das den Druck im Hintergrund ermöglicht, sich die auszudrucken-

den Daten merkt und dann eigenständig – ohne Mitwirkung von Excel – Zeile für Zeile an den Drucker schickt.

Sie können sogar mehrere Druckaufträge nacheinander erteilen, die eingesammelt und der Reihe nach abgearbeitet werden.

Excel selbst ist anschließend während des Ausdrucks unbeschäftigt und Sie können an der aktuellen oder einer anderen Tabelle weiterarbeiten, während Windows die Druckaufträge erledigt.

Da Ihr Rechner jedoch nicht wirklich zwei Dinge gleichzeitig erledigen kann – Ausdruck von Daten und gleichzeitiges Weiterarbeiten von Excel – gibt es einen Haken: Der Ausdruck wird immer dann kurzzeitig unterbrochen, wenn Sie Excel intensiv beschäftigen (zum Beispiel mit dem Neuberechnen einer Tabelle, dem Laden oder dem Speichern einer Datei).

Er wird jedoch sofort fortgesetzt, wenn Sie für einen Moment untätig vor dem Bildschirm sitzen. Diese beschäftigungslose Zeit nutzt der Druck-Manager, um wieder einige Zeilen an den Drucker zu senden.

Im Zweifelsfall wird immer Excel vorgezogen und der Druckvorgang für einen Moment unterbrochen. Sie können diese Prioritäten jedoch ändern: Drucken Sie bitte irgend etwas aus. In der Taskleiste erscheint ganz rechts ein Drucker-Symbol. Doppelklicken darauf öffnet den Druck-Manager und zeigt eine Liste aller momentan bearbeiteten Druckaufträge (Abbildung A.2).

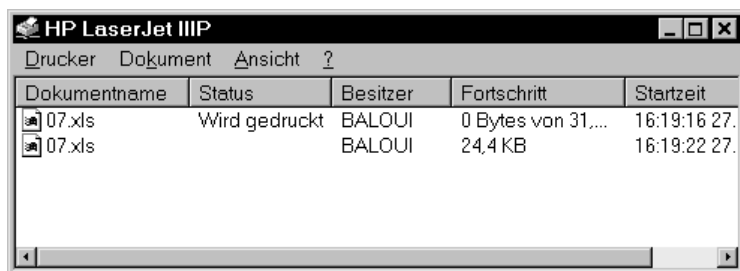


Abbildung A.2:
Der Druck-Manager

Die wichtigsten Möglichkeiten:

- ➔ Selektieren Sie einen Druckauftrag durch Anklicken des Dokumentnamens und drücken Sie **[Entf]**, so wird der betreffende Druckauftrag gelöscht.
- ➔ **DRUCKER|DRUCKAUFTRÄGE LÖSCHEN** löscht alle Druckaufträge.
- ➔ **DOKUMENT|DRUCKER ANHALTEN** unterbricht den Ausdruck der Dokumente bzw. setzt ihn nach erneuter Anwahl dieses Befehls fort.



Damit Sie nicht ebenso wie ich selbst kürzlich den Drucker völlig grundlos für defekt halten: Ich unterbrach den Ausdruck eines Dokuments mit DOKUMENT|DRUCKER ANHALTEN, setzte ihn jedoch nicht fort, sondern löschte kurz darauf alle Druckaufträge. Am nächsten Tag klappte kein Ausdruck mehr.

Warum wohl? Windows merkt sich dummerweise auf ewig, dass der Drucker angehalten wurde; auch ein Neustart ändert nichts an diesem Zustand! Sie müssen diesen Befehl erneut wählen, um jemals wieder drucken zu können.

A.4 Rechtschreibprüfung



EXTRAS|RECHTSCHREIBUNG... führt ebenso wie das zugehörige Symbol (Kategorie *Extras* des *Anpassen*-Dialogfelds) eine Rechtschreibprüfung in Tabellen, Grafiken oder Makro-Arbeitsblättern durch.

Dabei werden die in Zellen enthaltenen Texte überprüft, zugehörige Zellkommentare und an Grafikobjekte gebundene Texte, Kopf- und Fußzeilentexte ebenso wie Texte in ausgeblendeten Zeilen oder Spalten.

Sind zuvor mehrere Tabellen selektiert, erstreckt sich die Prüfung auf all diese Tabellen. Ist eine einzelne Zelle aktiv, wird der Inhalt der betreffenden Tabelle geprüft. Ist ein Bereich einer Tabelle selektiert, beschränkt sich die Rechtschreibprüfung auf diesen Bereich.

Bemängelt Excel ein Wort, das sich nicht in seinem Wörterbuch befindet, setzt es den Zellcursor auf die Zelle, in der sich das Wort befindet und öffnet ein Dialogfeld mit allen möglichen Optionen (Abbildung A.3).

Ist das Wort tatsächlich falsch, tippen Sie im Eingabefeld das Wort ein, durch das es ersetzt werden soll.

Eventuell schlägt Ihnen Excel unter *Vorschläge* einige Alternativen vor, wie das Wort ersetzt werden könnte. Um einen dieser Vorschläge anzunehmen, selektieren Sie einfach den betreffenden Listeneintrag, der danach im Eingabefeld erscheint.

In beiden Fällen aktivieren Sie anschließend *Ändern*, um das fehlerhaft geschriebene Wort durch Ihre manuelle Eingabe oder den von Ihnen selektierten Vorschlag ersetzen zu lassen.

Folgende Schaltflächen ermöglichen eine detailliertere Steuerung der Rechtschreibprüfung:

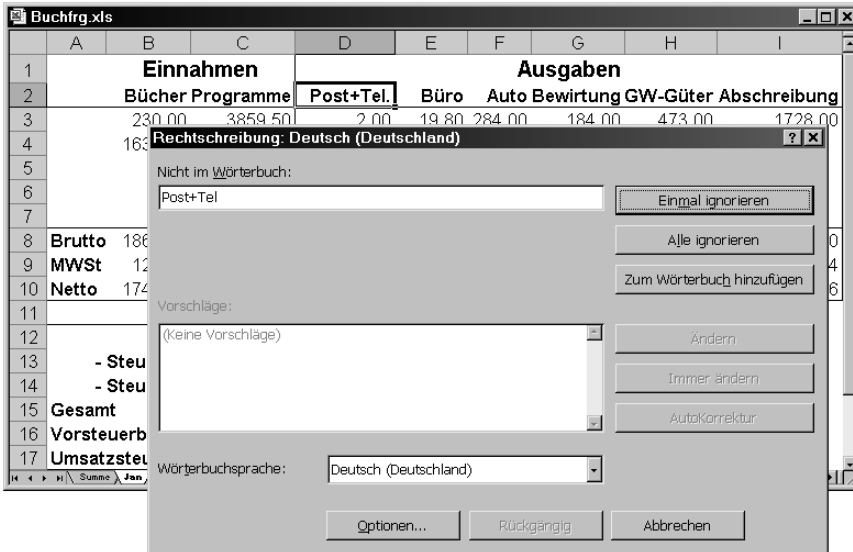


Abbildung A.3:
Rechtschreib-
prüfung

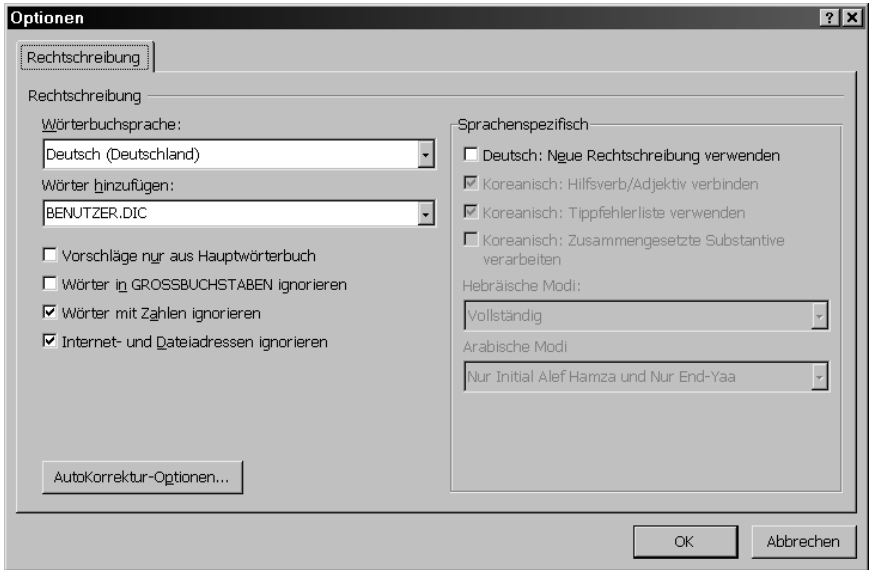
- ➔ **Einmal ignorieren:** Das angeblich falsch geschriebene Wort wird ignoriert und Excel setzt die Überprüfung fort.
- ➔ **Alle ignorieren:** Excel wird bei der weiteren Überprüfung das betreffende Wort immer ignorieren, was sinnvoll ist, wenn Sie völlig sicher sind, dass die Schreibweise korrekt ist.
- ➔ **Ändern:** Excel ersetzt das Wort durch das im Eingabefeld.
- ➔ **Immer ändern:** Excel ersetzt das betreffende Wort in allen Feldern, in denen es vorkommt, durch das Wort im Eingabefeld.
- ➔ **Zum Wörterbuch hinzufügen:** Excel nimmt das Wort in sein Wörterbuch auf und akzeptiert es bei folgenden Rechtschreibprüfungen als korrekt.
- ➔ **Rückgängig:** Excel macht die letzte Änderung rückgängig.
- ➔ Im Listenfeld **Wörterbuchsprache** können Sie die für die Rechtschreibprüfung verwendete Sprache auswählen.

AutoKorrektur: Excel fügt das Wort zusammen mit der gewählten Ersetzung in die AutoKorrektur-Liste ein, sodass es ab jetzt bereits bei der Texteingabe automatisch durch das korrekte Wort ersetzt wird (siehe Anhang A.5, »AutoKorrektur«).



Das von der Schaltfläche **Optionen...** geöffnete Dialogfeld legt verschiedene Einstellungen für die Rechtschreibprüfung fest (Abbildung A.4).

Abbildung A.4:
Rechtschreib-
prüfung



➔ Sind Wörterbücher in verschiedenen Sprachen installiert (nur dann verfügbar), können Sie im Listenfeld *Wörterbuchsprache* zwischen Wörterbüchern in verschiedenen Sprachen wie Englisch oder Französisch wählen.

➔ Unter *Wörter hinzufügen* können Sie bestimmen, in welches Wörterbuch der Eintrag aufgenommen wird: ob in das Standardwörterbuch von Excel oder in ein zusätzliches Wörterbuch, das angelegt wird, wenn Sie im Eingabefeld einen entsprechenden Namen eingeben.

Sie können auf diese Weise beliebig viele benutzerdefinierte Wörterbücher mit Namen wie »EDV« oder »Geographie« anlegen und das zu benutzende jeweils aus der Liste auswählen.

➔ *Vorschläge nur aus Hauptwörterbuch*: Ist dieses Kontrollkästchen aktiviert, verwendet Excel für seine Vorschläge nur sein Hauptwörterbuch, aber keine weiteren von Ihnen angelegten benutzerdefinierten Wörterbücher.

➔ *Wörter in GROSSBUCHSTABEN ignorieren*: Ist dieses Kontrollkästchen aktiviert, werden großgeschriebene Wörter wie »EDV« ignoriert.

➔ *Wörter mit Zahlen ignorieren*: Ist dieses Kontrollkästchen aktiviert, werden Wörter wie »2009er«, die Zahlen enthalten, ignoriert.

➔ *Internet- und Dateiadressen ignorieren*: Ist dieses Kontrollkästchen aktiviert, ignoriert Excel Wörter, die wie »C:\Test« Internet- bzw. Dateiadressen angeben.

- ➔ **Wörterbuchsprache:** Sind Wörterbücher in verschiedenen Sprachen installiert (nur dann verfügbar), können Sie im zugehörigen Listenfeld zwischen Wörterbüchern in verschiedenen Sprachen wie Englisch oder Französisch wählen.

Im rechten Dialogfeldabschnitt *Sprachenspezifisch* können Sie sprachspezifische Festlegungen treffen, im Falle von Deutsch beispielsweise wählen, ob alte oder neue Rechtschreibung verwendet werden soll.

- ➔ *AutoKorrektur-Optionen...* öffnet das im Anhang 5, »AutoKorrektur«, beschriebene Dialogfeld.



A.5 AutoKorrektur

Excel kann Texte bereits während der Eingabe korrigieren. Geben Sie beispielsweise den Text »Dies ist...« ein, wird nach Eingabe des Leerzeichens zwischen den beiden Wörtern das Wort »Dies« automatisch in »Dies« umgewandelt, die fehlerhafte Großschreibung also korrigiert.

Welche Korrekturen Excel ausführt, bestimmen Sie mit Hilfe des Befehls EXTRAS|AUTOKORREKTUROPTIONEN... (Abbildung A.5).

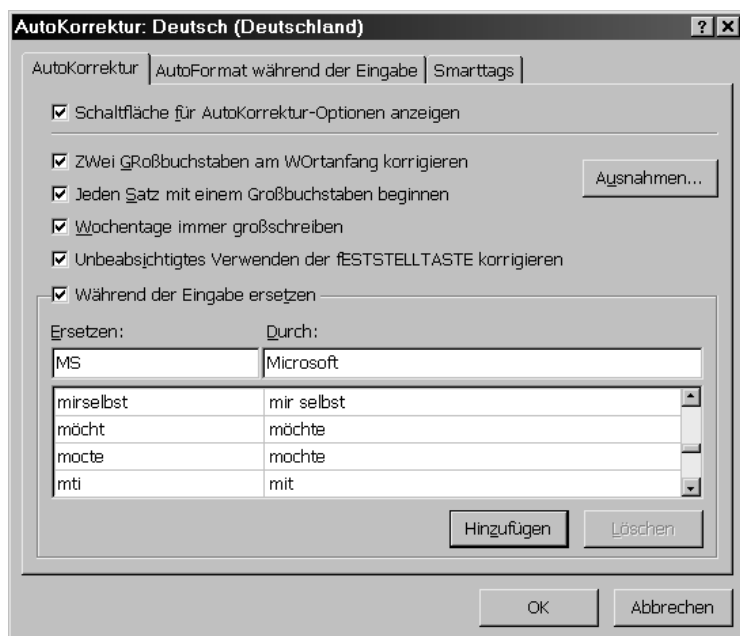


Abbildung A.5:
AutoKorrektur

Die Kontrollkästchen sind selbsterklärend. Das erste korrigiert beispielsweise zwei aufeinander folgende Großbuchstaben im Wort »TESt« automa-

tisch zu »Test«. Wichtig ist vor allem das untere Kontrollkästchen, das darüber entscheidet, ob die AutoKorrektur während der Eingabe überhaupt stattfinden soll.

Zusätzlich können Sie unter *Ersetzen* einen Text wie »MS« und unter *Durch* einen zweiten Text wie »Microsoft« eingeben, der den ersten automatisch ersetzen soll. Das Resultat: Geben Sie »Excel stammt von MS« ein, macht Excel daraus automatisch »Excel stammt von Microsoft«.

Hinzufügen fügt derartige Definitionen in das Listefeld ein und mit *Löschen* können Sie eine darin selektierte Ersetzungskombination genau umgekehrt jederzeit wieder aus dem Listefeld entfernen.

Ausnahmen... öffnet ein Dialogfeld, in dem Sie festlegen können, welche Ausnahmen bei der Option *Jeden Satz mit einem Großbuchstaben beginnen* berücksichtigt werden, damit beispielsweise nach einem Wort wie »ca.«, das mit einem Punkt endet, das folgende Wort nicht mit einem Großbuchstaben begonnen wird (»ca. drei« statt »ca. Drei«), und bei welchen Wörtern auch zwei aufeinander folgende Großbuchstaben nicht korrigiert werden sollen, z.B. »IBM«.



Im Register AutoFormat während der Eingabe können Sie wählen, ob Internet- und Netzwerkpfade durch Hyperlinks ersetzt werden sollen oder nicht.



Im Register Smart Tags können Sie wählen, ob Excel an bestimmte Daten, die es automatisch erkennt, wie E-Mail-Adressen, Smart Tags anheften soll, die schnellen Zugriff auf ein zu der betreffenden Datenart passendes Kontextmenü gestatten.

Je nach Auswahl im zugehörigen Listefeld wird dadurch eine Zelle mit einer E-Mail-Adresse wie *maier@ microsoft.com* zunächst durch den gewohnten kleinen Smart Tag-Indikator hervorgehoben oder aber dieser Indikator fehlt und nur beim Anklicken der Zelle erscheint das gewohnte Smart Tag-Listen-Symbol.

A.6 Add-Ins

Viele der in den verschiedenen Menüs enthaltenen Befehle basieren auf so genannten Add-Ins, Zusätzen, die sich in separaten Dateien befinden und Excel nach dem Einbinden um verschiedene Funktionen erweitern, zum Beispiel der Befehl EXTRAS|SOLVER..., der den Solver aufruft.

Add-Ins befinden sich im Ordner MAKRO, der sich wiederum im Ordner EXCEL befindet und in verschiedenen Unterordnern von MAKRO; darüber hinaus befinden sich zusätzliche Add-Ins auf der Microsoft Website, an die Sie mit EXTRAS|TOOLS IM WEB... herankommen.

*XLA/XLL/ DLL-
Dateien;*

Zu jedem Add-In gehört eine Datei mit der Erweiterung XLA, manchmal auch eine Datei mit der Endung XLL. Öffnen Sie die XLA-Datei, wird das Add-In geladen. Meist enthalten die Excel-Menüs danach einen zusätzlichen Menübefehl (fast immer im Menü EXTRAS), der den Zugriff auf die nun eingebaute Erweiterung ermöglicht.

Zu einigen Add-Ins gehören zusätzliche Dateien mit der Endung DLL. Dabei handelt es sich um Programm-Bibliotheken, die in einer normalen Programmiersprache wie zum Beispiel C geschrieben wurden.

Sie können sie zwar nicht wie andere Dateien einfach öffnen, müssen das jedoch auch gar nicht. Darum kümmert sich der in der zugehörigen XLA-Datei enthaltene Programmcode. Es reicht daher immer aus, die XLA-Datei des Add-Ins zu öffnen.

Möglicherweise finden Sie an einigen Add-Ins so viel Gefallen, dass Sie sie ständig verwenden wollen. Dazu können Sie sie prinzipiell in das Verzeichnis XLSTART kopieren, da Excel alle in diesem Verzeichnis enthaltenen Dateien beim Programmstart automatisch öffnet.

Der Befehl EXTRAS|ADD-INS... ermöglicht das Gleiche, jedoch wesentlich komfortabler (Abbildung A.6).

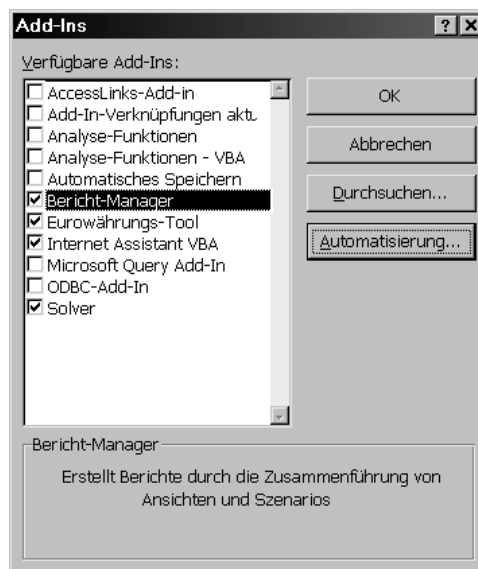


Abbildung A.6:
Der Add-In-
Manager

Add-Ins, deren Kontrollkästchen aktiviert sind, sind momentan eingebunden und werden bei jedem Aufruf von Excel automatisch geöffnet.

Da jedes Add-In mit dem Laden mindestens einer mehr oder weniger großen Datei verbunden ist, wird der Start von Excel durch die Einbindung vieler oft gar nicht benötigter Add-Ins entsprechend deutlich verlangsamt, sodass Sie alle überhaupt nicht oder nur sehr selten benutzten Add-Ins deaktivieren sollten.

Dazu deaktivieren Sie das zugehörige Kontrollkästchen und wählen *OK*. Excel entfernt das betreffende Add-In aus dem Speicher und lädt es bei zukünftigen Excel-Starts nicht mehr, sodass der Aufruf von Excel nun wesentlich schneller erfolgt.

Sie sollten versuchsweise einmal alle Kontrollkästchen aktivieren. Schauen Sie sich anschließend das Menü *EXTRAS* an, so werden Sie darin einige neue Befehle entdecken.



Durchsuchen... öffnet übrigens das Dialogfeld zum Öffnen von Dateien, mit dem Sie Add-Ins einbinden können, die Sie nachträglich von Microsoft oder von Fremdherstellern erhielten und die daraufhin der Liste hinzugefügt wurden.



Automatisierung öffnet ein Dialogfeld, in dem alle COM-Add-Ins aufgelistet werden, die eingebunden werden können, und das selbst wiederum eine Durchsuchen-Schaltfläche besitzt, mit der nach weiteren COM-Add-Ins gefahndet werden kann.

A.7 Excel individuell konfigurieren

Der Befehl *EXTRAS|OPTIONEN...* öffnet ein Dialogfeld, in dem Sie in verschiedensten Registern höchst unterschiedliche Einstellungen treffen können, die die Arbeit mit Excel beeinflussen. Im Folgenden werde ich nur einige wenige häufig sehr sinnvolle Optionen erläutern, da die meisten selbsterklärend sind.



Die meisten Einstellungen wie die Anzeige von Gitternetzlinien oder Spalten- und Zeilenköpfen sind nur für die aktive Tabelle gültig! Nur Optionen wie das Definieren eigener Listen für die AutoAusfüllen-Funktion, bei denen diese lokale Gültigkeit unsinnig wäre, sind global gültig, für alle Tabellen aller Arbeitsmappen!

A.7.1 Ansicht

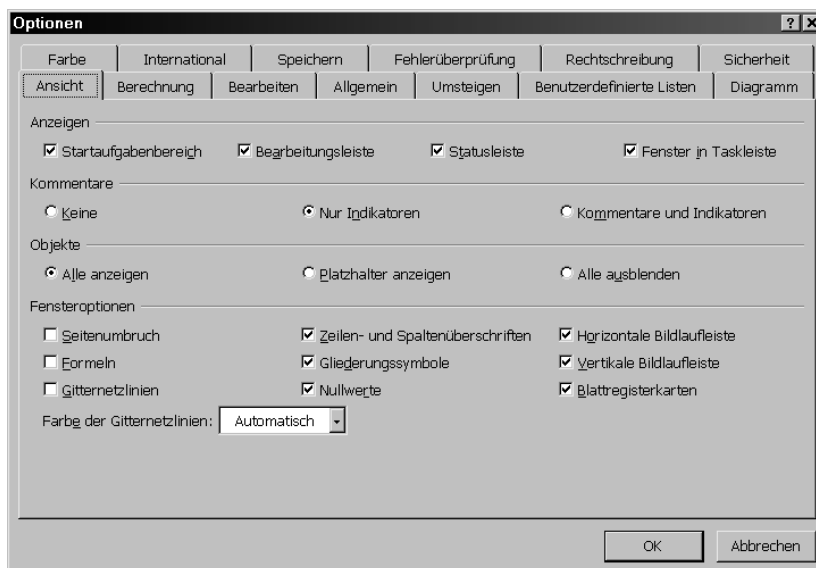


Abbildung A.7:
Register »Ansicht«

Im Register *Ansicht* (Abbildung A.7) können Sie verhindern, dass der Aufgabenbereich nach dem Öffnen automatisch erscheint und die Bearbeitungsleiste oder die Statusleiste ausblenden oder die Anzeige von Kommentaren und Grafikobjekten beeinflussen.

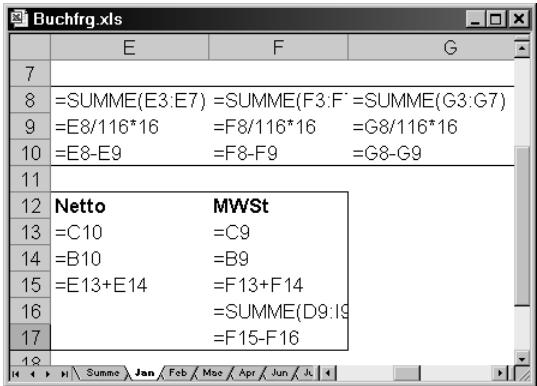
Ist *Fenster in Taskleiste* aktiviert, wird jedes in Excel geöffnete Fenster wie ein eigenes Anwendungsprogramm in der Taskleiste von Windows aufgeführt und Sie können darin jederzeit sehr bequem eine der geöffneten Mappen aktivieren.

Im Abschnitt *Fenster-Optionen* können Sie festlegen, ob spezielle Tabellen- und Arbeitsmappenelemente wie Gitternetzlinien, Reihen- und Spaltenköpfe, Gliederungs-Symbole und Bildlaufleisten angezeigt werden sollen.

Deaktivieren Sie beispielsweise *Nullwerte*, so wird in allen Zellen, die den Wert 0 enthalten, überhaupt nichts angezeigt. Aktivieren Sie *Formeln*, so werden die in den Zellen enthaltenen Formeln selbst statt der resultierenden Werte angezeigt (Abbildung A.8).

Die Aktivierung von *Seitenumbruch* führt zur Anzeige von Umbruchlinien, mit denen Excel in die Tabelle eingefügte Seitenumbrüche hervorhebt.

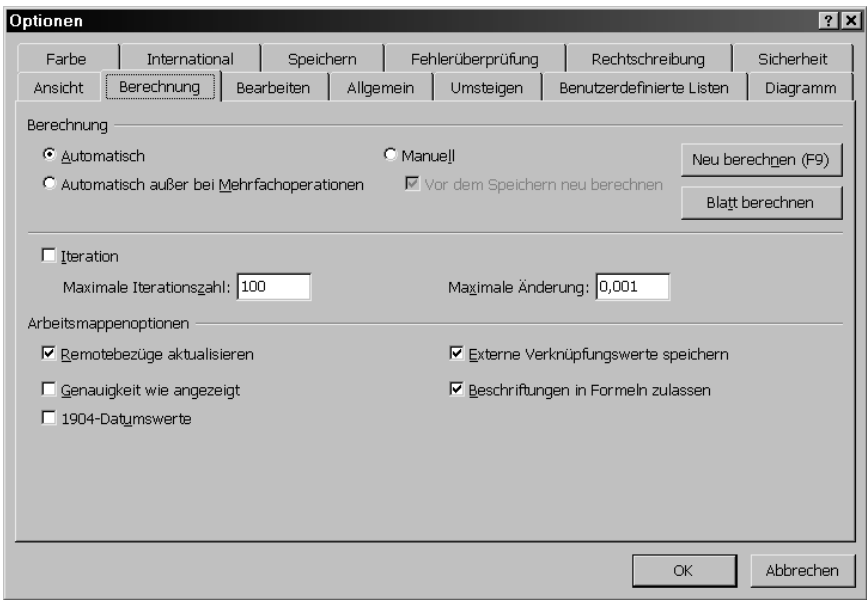
Abbildung A.8:
Formeln anzeigen



In der Rubrik *Objekte* zeigt *Alle anzeigen* alle in der Tabelle enthaltenen Grafikobjekte wie Linien, Kreise oder Diagramme an. *Platzhalter anzeigen* ersetzt diese Grafikobjekte durch graue Rechtecke. *Alle ausblenden* unterdrückt die Anzeige dieser Objekte völlig, wodurch bei langsamen Rechnern das Scrollen durch eine Tabelle, die Grafikobjekte enthält, beschleunigt werden kann.

A.7.2 Berechnen

Abbildung A.9:
Register »Berechnung«



Neu berechnen

Jede Änderung eines Zellinhalts führt zu einer Neukalkulation aller davon abhängigen Zellen. In großen Tabellen mit vielen Verknüpfungen kann eine Feldeditierung daher bedeuten, dass Dutzende oder Hunderte davon abhän-

giger Zellinhalte neu kalkuliert werden müssen, was unter Umständen zu Wartezeiten nach jeder einzelnen Editierung führt.

Um dennoch zügig arbeiten zu können, aktivieren Sie im Register *Berechnung* (Abbildung A.9) statt *Automatisch* die Option *Manuell*. Excel nimmt nun nach Änderungen keinerlei Neukalkulationen mehr vor und Sie können umfangreiche Datenmengen verzögerungsfrei eingeben.

Aufgrund der fehlenden Aktualisierung werden in Formelzellen zunächst teilweise falsche Werte angezeigt!

Das ignorieren Sie jedoch während der Dateneingabe und lassen einfach danach durch Aktivierung der Schaltfläche *Neu berechnen* () alle geöffneten Mappen oder aber mit *Blatt berechnen* nur die aktive Mappe *einmalig* neu durchkalkulieren.

Anschließend aktivieren Sie wieder *Automatisch*, um den üblichen Zustand wiederherzustellen.

Nur wenn *Manuell* aktiviert ist, lässt sich das Kontrollkästchen *Vor dem Speichern neu berechnen* umschalten. Ist es aktiviert, wird bei ausgeschalteter automatischer Neuberechnung die Tabelle vor dem Speichern aktualisiert und zeigt daher beim nächsten Öffnen korrekte Zellinhalte an.

Die Zielwertsuche und der Solver verwenden als Rechenmethode die Iteration. Dabei nähert sich die betreffende Funktion dem echten Ergebnis in einer Vielzahl von Rechenschritten – und erreicht es in ungünstigen Fällen niemals exakt.

Iterationen

Mit *Maximale Iterationszahl* bestimmen Sie, nach wie vielen Näherungsschritten die Annäherung spätestens aufgegeben wird, und mit *Maximale Änderung*, wie grob oder fein die einzelnen Näherungsschritte sein sollen. Je kleiner dieser Wert ist, desto genauer fällt das Ergebnis aus (und desto länger dauert die Berechnung).

Ist *Remotebezüge aktualisieren* aktiviert, aktualisiert Excel ständig alle Formeln, die Bezüge auf andere Anwendungen enthalten (DDE-Funktion), sodass diese jederzeit die aktuellen Werte der Fernverweise verwenden.

Remotebezüge

Bei deaktivierter Option werden stattdessen die zuletzt benutzten Werte weiterverwendet, auch wenn sich diese in der betreffenden Anwendung inzwischen geändert haben.

Genauigkeit wie angezeigt entscheidet darüber, ob Excel intern immer mit der vollen Genauigkeit von 15 Stellen rechnet oder aber – bei aktivierter

Genauigkeit

Option – nur mit der angezeigten Genauigkeit, bei mit dem Format »0,00« formatierten Zellen beispielsweise nur mit zwei Nachkommastellen.

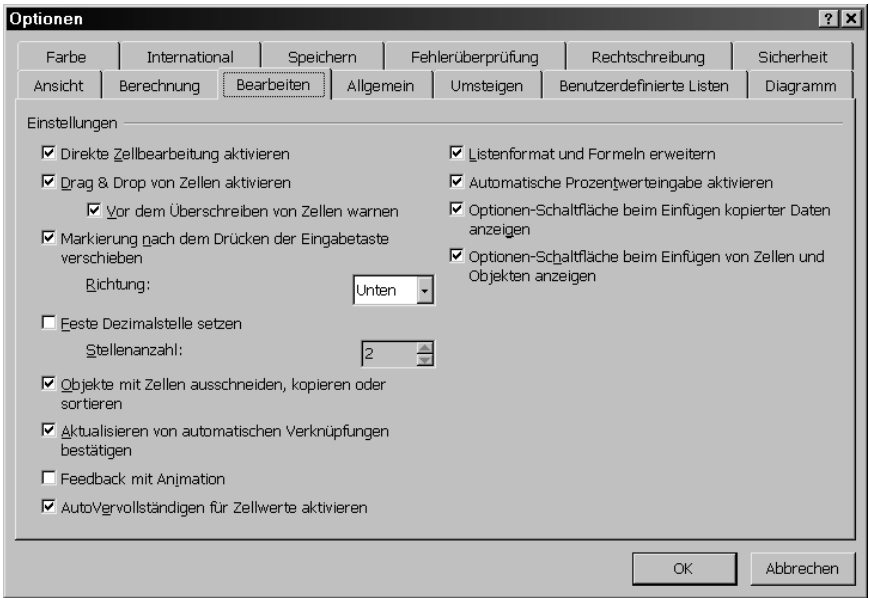
Datumswerte *1904-Datumswerte* ist nur für den Datenaustausch mit dem Apple Macintosh interessant. Die auf diesem Rechner laufende Excel-Version verwendet für Datumswerte serielle Zahlen, die beim 1. Jan 1904 beginnen – im Gegensatz zur PC-Version, bei der der seriellen Zahl 0 das Ausgangsdatum 1. Jan 1900 zugewiesen ist.

Um Excel-PC-Dateien mit Excel für den Macintosh weiterzubearbeiten (oder umgekehrt), müssen Sie diese Option aktivieren, da sonst beispielsweise das auf dem PC eingegebene Datum 1.1.1990 auf dem Mac als 1.1.1994 interpretiert wird.

Verknüpfungswerte *Externe Verknüpfungswerte speichern* speichert im Arbeitsblatt nicht nur externe Bezüge, sondern auch die resultierenden Verknüpfungswerte. Wird dadurch die auf der Platte gespeicherte Datei außerordentlich groß (Mappen mit sehr vielen externen Bezügen), deaktivieren Sie dieses Kontrollkästchen, um nur noch die externen Bezüge selbst in der Mappe zu speichern.

A.7.3 Bearbeiten

Abbildung A.10:
Register
»Bearbeiten«



Im Register *Bearbeiten* (Abbildung A.10) dürfte speziell *Feste Dezimalstelle setzen* interessant sein. Mit der Vorgabe 2 wird aus der eingetippten »21«

nach dem Beenden der Eingabe eine »0,21«, sodass Sie sich die Eingabe umfangreicher Zahlenkolonnen mit festen Dezimalstellen vereinfachen können.

Ist Listenformat und Formeln erweitern aktiviert, versucht Excel bei Eingaben im Listenformat, also untereinander oder nebeneinander angeordneten Text- oder Zahlenkolonnen, selbstständig zu ermitteln, ob die gesamte Kolonne eine einheitliche Formatierung aufweist. Falls ja, wird diese Formatierung automatisch auch der neuen Eingabe zugewiesen. Befinden sich beispielsweise in Spalte A im Bereich A1:A5 fett formatierte Zahlen und geben Sie in A6 eine weitere Zahl ein, wird diese ebenfalls automatisch fett formatiert.



Noch interessanter ist die automatische Formelerweiterung: Angenommen, rechts neben dieser Zahlenkolonne, also in B1:B5, befindet sich eine Kolonne mit Formeln, die beispielsweise einen bestimmten Prozentsatz der in Spalte A enthaltenen Werte ermitteln (in B1 die Formel $=20\% * A1$, in B2 die Formel $=20\% * A2$ etc.). Sie geben nun am Ende der Zahlenkolonne in A6 einen neuen Wert ein. Excel nimmt daraufhin an, dass in der Zelle daneben, in B6, analog die Formel $=20\% * A5$ benötigt wird – und trägt diese Formel selbstständig in B6 ein.

Optionen-Schaltfläche beim Einfügen kopierter Daten anzeigen und Optionen-Schaltfläche beim Einfügen von Zellen und Objekten anzeigen legen fest, ob nach dem Einfügen der Zwischenablage bzw. von Zellen die entsprechenden Smart Tags erscheinen.

Smart Tags

A.7.4 Umsteigen

Im Register *Umsteigen* (Abbildung A.11) können Sie unter *Excel-Dateien speichern unter* das Standard-Dateiformat wählen, das Excel beim Speichern verwendet, beispielsweise das Excel 5.0-Format, falls die meisten Ihrer Kollegen nur über diese Excel-Version verfügen und dennoch problemlos auf Ihre Mappen zugreifen können sollen.

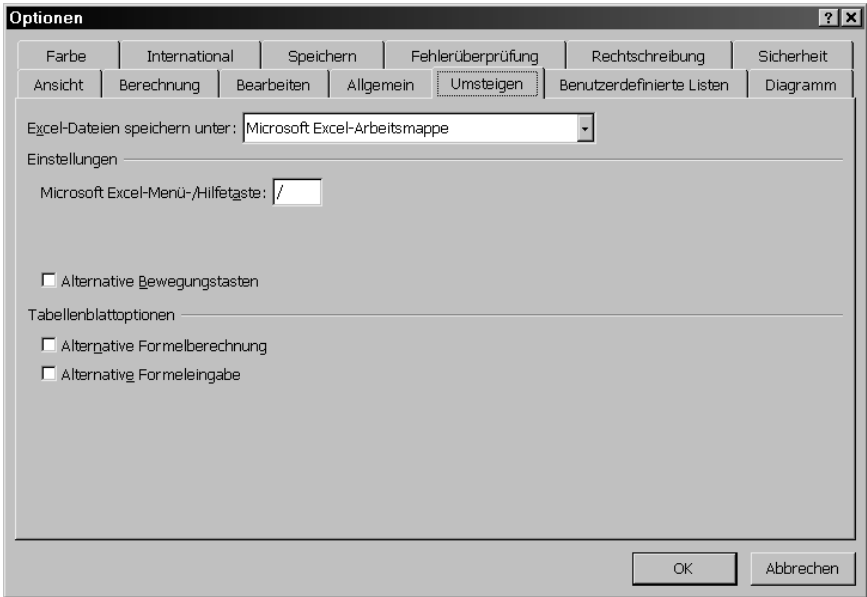
Standard-Speichertyp

Alternative Formelberechnung steuert die Auswertung von Ausdrücken wie $2+3 * \text{SUMME}(A3:B5) * (4-3)$. Ist dieses Kontrollkästchen aktiviert, erfolgt die Auswertung gemäß den Regeln von Lotus 1-2-3: Sie können ein Lotus-Arbeitsblatt öffnen, ohne dass die darin enthaltenen Ausdrücke eventuell falsch ausgewertet werden.

Ausdrucks-berechnung

Alternative Formeleingabe konvertiert nach dem Lotus-Schema eingegebene Formeln in die Excel-Syntax.

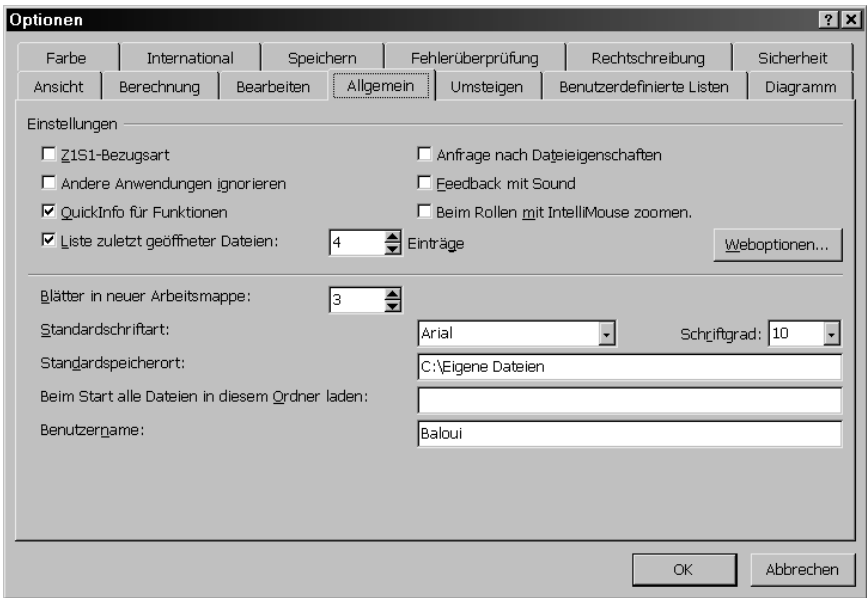
Abbildung A.11:
Register
»Umsteigen«



Unter *Einstellungen* können Sie die Excel-Menübedienung wählen, um beispielsweise wie in Lotus mit der vorgegebenen Taste »/« Menüs zu aktivieren, oder alternativ dazu eine beliebige andere Menüaktivierungstaste.

A.7.5 Allgemein

Abbildung A.12:
Register
»Allgemein«



Die meisten Optionen des Registers *Allgemein* (Abbildung A.12) habe ich bereits erläutert, zum Beispiel die beiden Bezugsarten, die Anzahl der Tabellen, die eine neu angelegte Mappe enthält, und den zusätzlichen Startordner, oder dass Sie eine Standardschriftart und -größe wählen können, die beim nächsten Start von Excel automatisch vorgegeben wird.

Mit *Standardspeicherort* können Sie *jenen Ordner angeben, der beim Speichern oder Öffnen einer Mappe automatisch vorgegeben werden soll, und mit Weboptionen...* (Abbildung A.12) unterschiedlichste Einstellungen für das Speichern/Öffnen von Dateien im Internet festlegen (siehe Kapitel 30.1, »Weboptionen«).



A.7.6 Benutzerdefinierte Listen

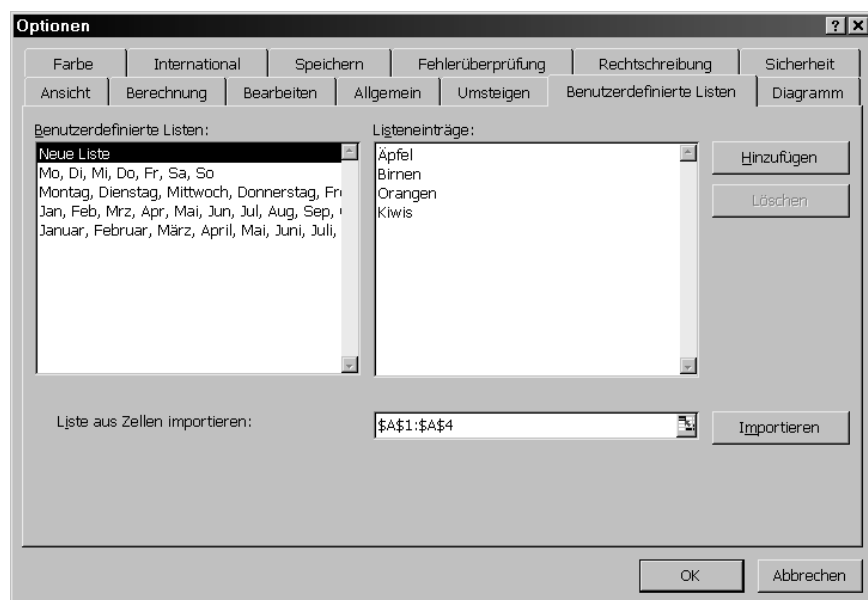


Abbildung A.13:
Register »Benutzer-
definierte Listen«

Die *AutoAusfüllen*-Funktion verlängert wie erläutert eine vorgegebene Reihe wie »Jan« und »Feb« um »Mrz«, »Apr« etc. Dabei erkennt sie ausschließlich die unter *Benutzerdefinierte Listen* enthaltenen Reihen.

Soll Excel eine mit »Äpfel« und »Birnen« begonnene Liste beim *AutoAusfüllen* mit »Orangen«, »Kiwis« und dann wieder von vorne mit »Äpfel«, »Birnen«, »Orangen« etc. ergänzen, definieren Sie im Register *Benutzerdefinierte Listen* eine entsprechende Benutzerliste (Abbildung A.13).

In der Liste *Benutzerdefinierte Listen* selektieren Sie »Neue Liste« und geben entsprechend der Abbildung unter »Listeneinträge« »Äpfel«, »Birnen«, »Orangen« und »Kiwis« ein.

Drücken Sie nach der Eingabe jedes Listenelements  und fügen Sie die Liste mit *Hinzufügen* den bereits vorhandenen Listen hinzu.

Löschen löscht die selektierte Liste.

Importieren fügt jene Liste hinzu, die im unter *Liste aus Zellen importieren* angegebenen Tabellenbereich definiert ist, zum Beispiel die abgebildete Obstliste, wenn der Bereich A1:A4 angegeben wird und dieser Bereich die vier Texte enthält.

Selbstdefinierte Listen können Sie auch beim Sortieren verwenden, um eigene Sortierreihenfolgen zu definieren (siehe Kapitel 13.1.2, »Sortieroptionen«).

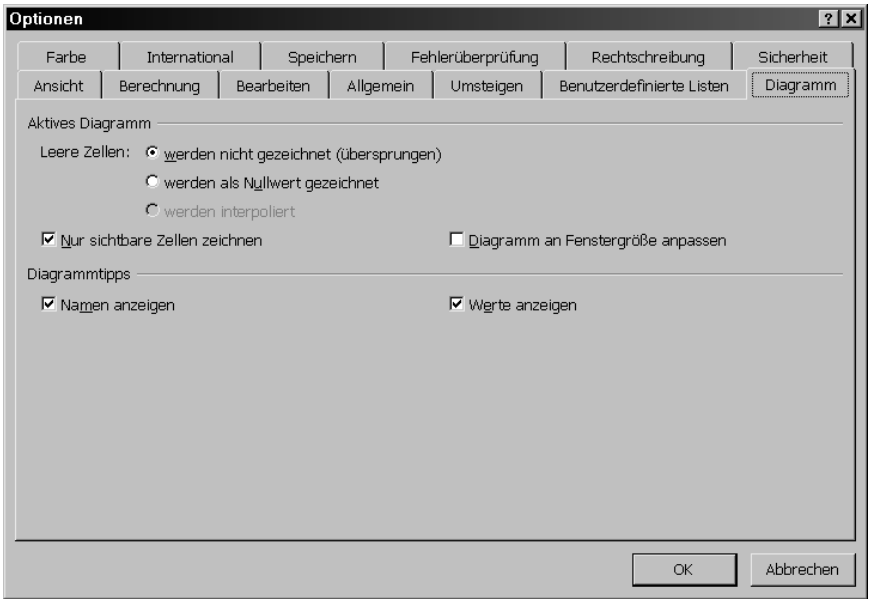


A.7.7 Diagramm

Bevor Sie das Register Diagramm aktivieren, muss ein Diagramm aktiv sein (Abbildung A.14).



Abbildung A.14:
Register
»Diagramm«



Dann können Sie darin Voreinstellungen für das aktive und alle neu erzeugten Diagramme festlegen.

Unter *Leere Zellen* definieren Sie, wie im Quellbereich enthaltene leere Zellen behandelt werden:

- ➔ ob Nullwerte einfach übergangen werden sollen, um derartige Lücken zu vermeiden,
- ➔ ob sie im Diagramm als Nullwert eingezeichnet werden, beispielsweise als Balken mit der Höhe Null (was sich als Lücke zwischen den benachbarten Balken bemerkbar macht)
- ➔ oder ob sie interpoliert, das heißt der Mittelwert der benachbarten Balken, eingezeichnet werden sollen.

Ist *Nur sichtbare Zellen zeichnen* aktiviert, werden beim Ausblenden von Werten in Tabellen mit Gliederungen alle zugehörigen Balken, Säulen etc. sofort aus dem Diagramm entfernt.

Ist *Diagramm an Fenstergröße anpassen* aktiviert, füllt das Diagramm das zugehörige Fenster immer aus und passt sich sofort jeder Veränderung der Fenstergröße an.

Ist *Namen anzeigen* aktiviert, zeigt Excel den Namen des Diagrammelements an, auf dem sich der Cursor befindet. *Werte anzeigen* zeigt analog dazu den Wert an, den der Datenpunkt unter dem Cursor darstellt.

A.7.8 Farbe

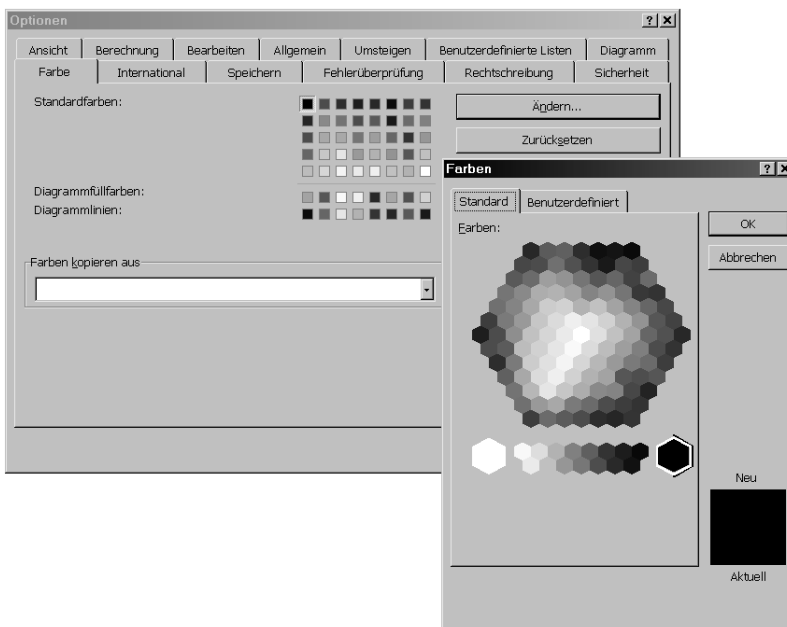
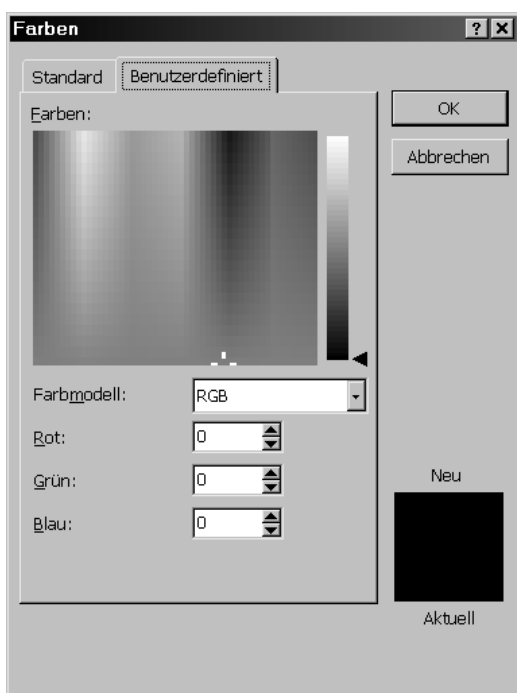


Abbildung A.15:
Register »Farbe«
und »Standard«

Im Register *Farbe* (Abbildung A.15) können Sie die für die verschiedenen Bildelemente verwendeten Standardfarben (und separat dazu die Diagrammfüllfarben und Diagrammlinien) durch andere Farben ersetzen: Sie selektieren die zu ersetzende Farbe durch Anklicken und wählen *Ändern...*, um das in der Abbildung sichtbare Register *Standard* des *Farben*-Dialogfelds zu öffnen und wählen darin die neue Farbe durch Anklicken aus.

Sie können jedoch auch das Register *Benutzerdefiniert* aufschlagen, das die Beeinflussung aller Parameter ermöglicht, die die selektierte Farbe auszeichnen (Abbildung A.16).

Abbildung A.16:
Register »Benutzer-
definiert«



Das aktuelle Ergebnis wird in der Musterbox angezeigt.

Ein Farbton besteht aus einer Mischung der Grundfarben Rot, Grün und Blau – zumindest im RGB-Farbmodell; unter »Farbmodell« können Sie bei Bedarf ein anderes Farbmodell wählen. Der jeweilige Grundfarbanteil ist eine Zahl zwischen 0 und 255. Beispielsweise definiert die Zahlenkombination 255/0/0 reines Rot (255 Anteile Rot, jeweils 0 Anteile Grün und Blau).

Für eine Farbmanipulation ziehen Sie den Schieber am rechten Bildschirmrand oder das Fadenkreuz, das nach dem Anklicken des Farbfelds erscheint, in Richtung des gewünschten Farbtons. Die Inhalte der Eingabefelder passen sich automatisch entsprechend an.

Im übergeordneten Dialogfeld können Sie mit *Farben kopieren aus* eine in einer anderen Tabelle auf diese Weise von Ihnen definierte Farbpalette in die gerade aktive Tabelle übernehmen.

Das Listenfeld zeigt alle Tabellen außer der aktiven an, die sich momentan im Speicher befinden. Klicken Sie eine davon an, wird die darin definierte Farbpalette in die aktive Tabelle übernommen.

Zurücksetzen macht Änderungen der Farbpalette für die aktive Tabelle rückgängig.

A.7.9 International

Im Register *International* können Sie unter anderem einstellen, welches Dezimaltrennzeichen Excel verwendet (Abbildung A.17).

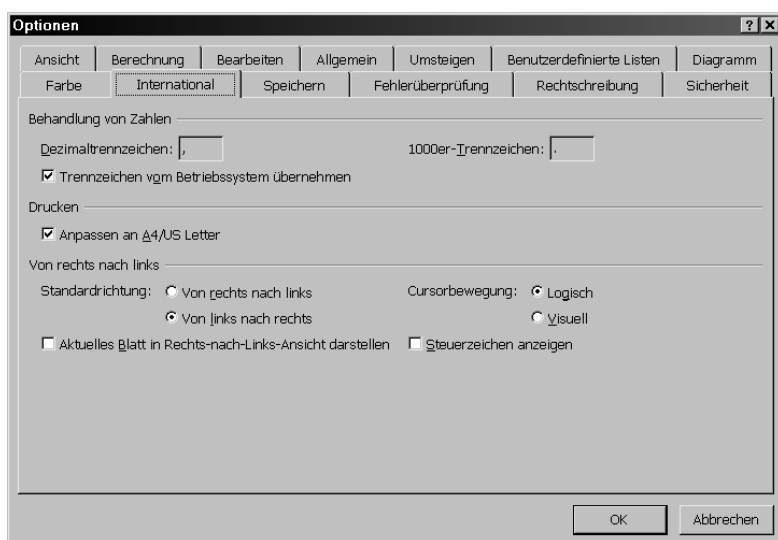


Abbildung A.17:
Register
»International«

Das ist äußerst nützlich, wenn Sie beispielsweise aus einer englischsprachigen Webseite Aktienkurse importieren, in denen dann ja das englische Dezimaltrennzeichen ».« verwendet wird.

Nur mit Umstellung dieses Dezimaltrennzeichen vor dem Import interpretiert Excel einen importierten Kurs wie »12.78« korrekt und zeigt nach erneuter Umstellung auf ».,« entsprechend »12,78« an.



Darüber hinaus können Sie Excel mit diesem Register an Sprachen anpassen, deren Schreibweise von rechts nach links orientiert ist (siehe Kapitel 6.7, »Text- und Zahlenausrichtung«).

A.7.10 Speichern

Im Register *Speichern* können Sie festlegen, in welchen Abständen Excel die Informationen zur – hoffentlich nicht notwendigen – Wiederherstellung von Dokumenten speichert (Abbildung A.18).

Abbildung A.18:
Register
»Speichern«



A.7.11 Fehlerüberprüfung



Im Register *Fehlerüberprüfung* können Sie wie bereits erläutert (siehe Kapitel 38.3, »Fehler in Tabellen aufdecken«, Abschnitt »Fehlerüberprüfung«) die Regeln festlegen, gemäß denen Excel Fehler erkennt (Abbildung A.19).

A.7.12 Rechtschreibung



Im Register *Rechtschreibung* legen Sie wie erläutert (siehe Anhang A.4, »Rechtschreibprüfung«) die Funktionsweise der Rechtschreibprüfung fest (Abbildung A.20).



Abbildung A.19:
Register »Fehler-
überprüfung«

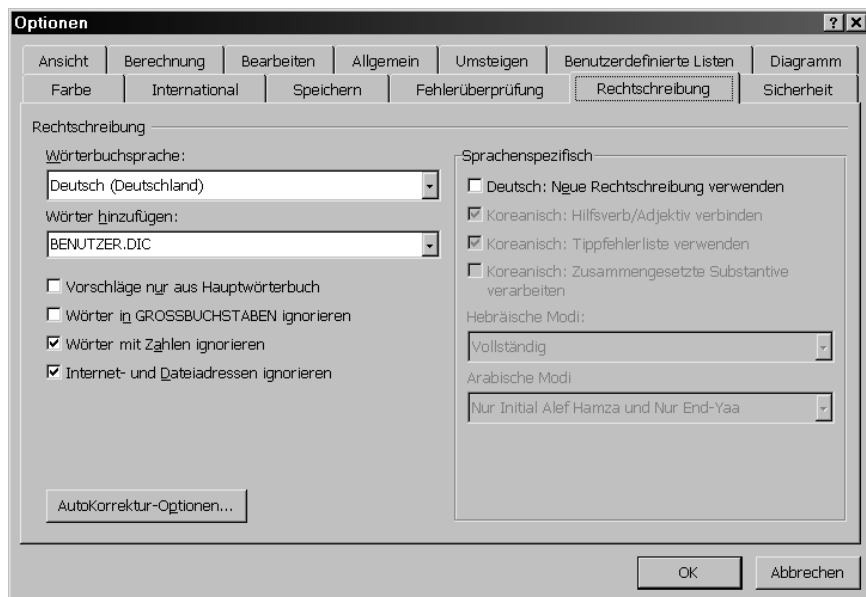
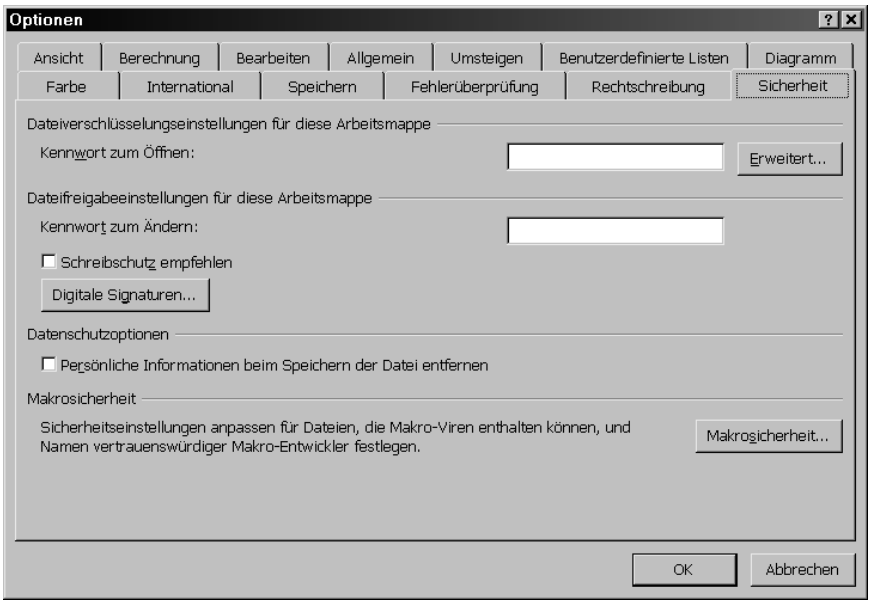


Abbildung A.20:
Register »Recht-
schreibung«

A.7.13 Sicherheit

Im Register *Sicherheit* legen Sie diverse Sicherheits-Optionen fest (Abbildung A.21).

Abbildung A.21:
Register
»Sicherheit«



Dabei handelt es sich zum einen um die bereits erläuterte Verschlüsselung von Mappen (siehe Kapitel 3.1.1, »Speicher-Optionen«), aber auch um zusätzliche Optionen wie das Entfernen persönlicher Informationen aus den gespeicherten Dokumenten (Autorenname etc.) oder die Festlegung des Makro-Sicherheitslevels, die darüber bestimmt, wie sich Excel beim Öffnen eines Dokuments verhält, das Makros enthält.

Stichwortverzeichnis

Häufig gibt es zu einem Stichwort mehrere Seitenverweise. Kursiv formatierte Seitenangaben weisen auf jene Seiten hin, auf denen das betreffende Stichwort besonders ausführlich erläutert wird.



!

#DIV/0 359
#NAME#? 332
#NAME? 117, 227
#WERT! 555
#ZAHL! 359
' 824
(Deklarationen) 784, 853
 81
(Name) 952

 Bildlauf 87
 Dialogfeld 81
* 431
+, 168
... 583
.XLT 327
<CR><LF>-Kombination 1046
? 431
?IDirekthilfe 96
?!Erkennen und Reparieren... 65
?!Microsoft Excel-Hilfe 98
?!Produkt aktivieren... 65
€ 215, 223
12-Stunden-Einteilung 232
1904 Datumswerte 1080
24-Stunden-Einteilung 232
3D-Diagramme 647
3D-Diagramm-Optionen 677
3D-Säulendiagramm-Varianten 647
3D-Verknüpfungen 319

A

A1 210
Abbrechen 79
Abfragedefinition speichern 454
Abkürzen 380
ABS 363

Absätze 578
Absolute Bezüge 121
Abstandsbreite 676, 678
Access 445
Achsen 681
Achsenannah 684
Achsenschnittpunkt 692
Achsenspezifisches Kontextmenü 660
Achsentexte ausrichten 682
Achsentitel 599
Activate-Methode 925
Active Server Pages 757
ActiveSheet 909
Add-In 447, 1009
 speichern 815
Add-In-Format 815
Add-In-Manager 815
Addition 115
Additionsoperator 114
Adressangabe, Hyperlink 147
Änderungs-Protokoll 305
Aktienexperten 621
Aktienkurse 634, 737
 Dezimaltrennzeichen 744
 internationale 744
Aktienkurs-Smart Tag 53, 743
Aktive Mappe 926
Aktive Zelle 157, 926
Aktives Blatt 926
Aktives Diagramm 926
Aktives Fenster 92, 926
Aktivierung 65
 Aktivierungs-Assistent 65
 Freischaltcode 66
 per Internet 66
 per Telefon 66
 Rechner-Hardware 66

- Aktualisierbare Webabfrage erstellen... 53, 743
- Aktueller Bereich 207, 525
- Aktuelles Array 207
- Aktuelles Datum und die Uhrzeit anzeigen 231
- Alias-Namen 756
- Alle Ebenen 211
- Allgemein 230
- Alphabetisch 382
-  Dialogfeld 80
- Menü 69
-  +  88
-  +  88
-  + Buchstabe 69
-  +  111, 285
-  +  +  352
-  +  +  351
- Alternative Formelberechnung 1081
- Alternative Formeleingabe 1081
- An eine andere Stelle kopieren 438
- An Markierung anpassen 319
- An Seite anpassen 613
- Analysen 408
- Anfasser 579
- Anführungszeichen 360
- Angepasste Formel 123
- Anpassen 282
- ANSI 547
- Ansicht|Aktivierreihenfolge 959
- Ansicht|Aufgabenbereich 83
- Ansicht|Benutzerdefinierte Ansichten... 415
- Ansicht|Definition 798
- Ansicht|Diagrammfenster 605
- Ansicht|Direktfenster 801
- Ansicht|Eigenschaftenfenster 780
- Ansicht|Ganzer Bildschirm 89
- Ansicht|Kommentare 137
- Ansicht|Kopf- und Fußzeile... 284
- Ansicht|Kriterien 465
- Ansicht|Letzte Position 797
- Ansicht|Lokal-Fenster 806
- Ansicht|Normal 295
- Ansicht|Objektkatalog 798
- Ansicht|Projekt-Explorer 780
- Ansicht|Seitenumbruch-Vorschau 293
- Ansicht|SQL... 465
- Ansicht|Symbolleisten 72, 72
- Ansicht|Tabellen 463
- Ansicht|Überwachungsfenster 811
- Ansicht|Zoom... 318, 463
- Ansichtenliste 415
- Ansichten-Manager 415
- ANSI-Zeichen-Kodierung 545
- Antwort 402
- Anweisungen 784
- Anweisungsblock 878
- Anzeige 968
- Anzeige-Einschränkung 505
- Append 1039
- Application 904
- Arbeitsbereich, speichern 151
- Arbeitsbereichsdatei 151, 542
- Arbeitsblatt 67
- Arbeitsblattgruppe 308
- Arbeitsmappe 94
 - freigeben 91
 - neu anlegen 301
 - schützen 1008
- Arbeitsmappenfunktionalität 730
- Arbeitsmappen-Registerkartenleiste 308
- Arbeitsmappen-Registerleiste 94
- Argument 357, 362
 - benannt 846
 - optional 846
- Argumentart 359
- Argumentliste 844
- Arithmetisch 200
- Array 857, 864
- Array-Grenzfunktionen 860
- Array-Untergrenze 859
- Art 583
- As Typ 844
- ASCII-Code 1043, 1048
- ASCII-Code 13 820
- ASCII-Dateien 1044
- ASCII-Zeichen-Kodierung 545
- Assistent auswählen... 98
- Assistenten-Galerie 99
- Aufeinanderfolgende Trennzeichen als ein Zeichen behandeln 552, 554

Aufgabenbereich 82, 1077
 Auflösung 274, 283
 Aufruf 64
 Ausblenden 246
 Ausdruck im Querformat 276
 Ausdruck löschen und bearbeiten 815
 Ausdrucken 277
 Ausdrucksberechnung 1081
 Ausführbare Anweisungen 784
 Ausführen 767
 Ausführen|Entwurfsmodus 957
 Ausführen|Fortsetzen 808
 Ausführen|Sub/UserForm ausführen 792, 956
 Ausführen|Unterbrechen 808
 Ausführen|Zurücksetzen 808, 853
 Ausfüllen 248
 Ausgabe in Datei umleiten 278
 Ausgangsbasis 365
 Ausgangswerte wiederherstellen 402
 Ausgeblendet 1008
 Ausgewählte Blätter 278
 Ausmaskieren 380
 Ausprägungen 485
 Ausprägungskombination 486, 492
 Ausrichtungsvarianten 247
 Aussage 380
 Aussagenprüfung 382
 Ausschnitte der gleichen Tabelle 316
 Auswahl mehrerer, rechteckiger Bereiche 159
 Auswahlliste... 113
 Auswahlmenüs 67
 Auswertungen 473
 Auswertungsfunktion 513
 Auto_Activate 769
 Auto_Close 769
 Auto_Deactivate 770
 Auto_Open 769
 AutoAusfüllen 190, 193, 197, 1083
 AutoAusfüllen-Kontextmenü 196
 AutoAusfüllen-Smart Tag 190, 196
 AutoBerechnung 135
 AutoFilter 426
 AutoFormatieren-Funktion 262
 AutoFormen 591

AutoGliederung 348
 AutoKorrektur 117, 1073
 Auto-Makros 769
 Auto-Makro-Typen 769
 Automatisch 583, 684
 Automatische Daten-Tipps 810
 Automatische Formatierung 351
 Automatische Gliederung 347
 Automatische Größe 584
 Automatische Mustervorlage 329
 AutoSkalieren 650
 AutoVervollständigen für Zellwerte 113

B

Balkendiagramme 633
 Balkendiagramm-Varianten 633
 Barwert 387
 BAS 779
 Basisverzeichnis 716, 754, 755
 Bearbeiten...Objekt 568
 Bearbeiten...Objekt|Bearbeiten 569
 Bearbeiten|Ausfüllen 190, 191
 Bearbeiten|Ausfüllen|Im Blocksatz ausrichten 192
 Bearbeiten|Ausfüllen|Rechts 192
 Bearbeiten|Ausfüllen|Reihe... 196
 Bearbeiten|Ausfüllen|Unten 192
 Bearbeiten|Ausschneiden 110, 170
 Bearbeiten|Blatt löschen 311
 Bearbeiten|Blatt verschieben/kopieren... 313
 Bearbeiten|Eigenschaften/Methoden anzeigen 797
 Bearbeiten|Einfügen 110, 171, 179
 Bearbeiten|Einzug vergrößern 789
 Bearbeiten|Einzug verkleinern 789
 Bearbeiten|Ersetzen 203, 798
 Bearbeiten|Gehe zu... 204, 247, 525
 Bearbeiten|Inhalte einfügen 185, 186, 534, 565
 Bearbeiten|Konstanten anzeigen 797
 Bearbeiten|Kopieren 110, 124, 179, 563
 Bearbeiten|Lesezeichen|Alle Lesezeichen löschen 795
 Bearbeiten|Lesezeichen|Lesezeichen setzen/zurücksetzen 794

Stichwortverzeichnis

- Bearbeiten|Lesezeichen|Nächstes Lesezeichen 795
- Bearbeiten|Lesezeichen|Vorheriges Lesezeichen 795
- Bearbeiten|Löschen|Formate 218
- Bearbeiten|Löschen|Kommentare 137
- Bearbeiten|Office-Zwischenablage 564
- Bearbeiten|ParameterInfo 796
- Bearbeiten|QuickInfo 795
- Bearbeiten|Rückgängig 164, 264
- Bearbeiten|Suchen... 201, 798
- Bearbeiten|Verknüpfungen... 567, 719
- Bearbeiten|Wiederholen... 164
- Bearbeiten|Wort vervollständigen 796
- Bearbeiten|Zellen löschen... 173, 178
- Bearbeitungsleiste 67, 108, 1077
- Bearbeitungsmodus 109
- Bedingte Formate 208
- Bedingte Kompilierung 881
- Bedingte Verzweigung 874
- Bedingung 876
- Beenden 89
- Befehls-Schaltfläche 953, 967
- Bei gleicher Spalte entfällt Spaltenname 345
- Bei gleicher Zeile entfällt Zeilenname 345
- Benannte Argumente 846
- Benutzerdefiniert 611
- Benutzerdefinierte Fußzeile... 285
- Benutzerdefinierte Kopfzeile... 285
- Benutzerdefinierte Sortierreihenfolge 425
- Benutzerliste 1083
- Benutzeroberfläche 59
- Berechnete Felder 440
- Berechnungsreihenfolge 115
- Bereich 359
 - entfernen 173
 - Selektion 157
- Bereichsbezug 940
- Bereichsname 334, 338
- Bereichsrand 343
- Bestimmtheitsmaß 623
- Bestimmtheitsmaß im Diagramm darstellen 623
- Betrachtungshöhe 649
- Bezeichnungsfeld 967
- Bezüge, externe 322
 - auf externe Bezüge 855
- Bezug auf ein Objekt übergeben 917
- Bezug mit implizitem Schnittpunkt 345
- Bezugseindeutigkeit 912, 920
- Bezugsformat 120, 121, 210
- Bezugslinien 678
- Bibliothek 855
- Bibliotheken 854
 - VBA und Excel 799
- Bild ab 88
- Bild auf 88
- Bilddatei 968
- Bilder
 - einfügen 287
 - Farbanzahl 590
 - formatieren 287
 - Helligkeit 590
 - komprimieren 760
 - Kontrast 590
 - manipulieren 590
 - Qualität 761
 - Transparente Farbe 590
 - Von Scanner oder Kamera 589
- Bildlauf 85
- Bildlauffeld 86
- Bildlauffeiste 88, 967, 1077
- Bildlaufpfeile 77, 77
- Binärdatei 1035, 1040
- Binärer Zugriff 1059
- Binärmodus 1060
- Bitmuster 242
- Blätter in neuer Arbeitsmappe 94
- Blätter löschen 311
- Blasendiagramme 641
- Blasendiagramm-Optionen 680
- Blasendiagramm-Varianten 641
- Blasengröße 618
- Blatt berechnen 1079
- Blatt kopieren 313
- Blatt schützen 1006
- Blatthintergrund 256, 257
- Blattschutz 1023
- Blocklänge 1037, 1040
- Blocklänge einer Direktzugriffsdatei 1038
- Blocksatz 249

Blockstrukturen 877
 Börsenanwendungen 678
 Börsenbeginn 679
 Börsenschluss 679
 Bogenmaß 388
 Boolean 827
 Breite-/Höhe-Verhältnis 613
 Bruchformate 225
 Buchfrg.xlT 1022
 Buchführung 1023
 Buchhaltungsformate 224
 Bw 387
 ByRef 849
 Byte 827
 Bytenummer 1060
 Bytenummern statt Recordnummern 1060

C

Call 841
 Caption 953
 Carriage Return 543
 Case 882
 Case Is 883
 Case-Ausdruckliste 883
 Cells-Methode zur Bereichsangabe
 verwenden 928
 Chart 542
 Chart-Objekt 944
 Click 959
 ClipArt 593
 Clipboard 561
 Clipper 561
 Close-Anweisung 1040
 Code editieren 788
 Codezeilen 824
 COM-Add-In 55, 1076
 Compiler-Direktive, #If 881
 Const 832
 ControlSource 969
 CSng-Funktion 938
 CSV 541, 543, 545
 CUB 515
 Cube 511
 Cube-Abfrage 516
 Currency 828

D

Dann_Wert 382
 Date 828
 Datei 1038
 binäre 1035
 Direktzugriff 1035, 1050
 öffnen 1038
 schließen 1038
 sequentielle 1035, 1041, 1041
 DateilAufgabenbereich speichern... 151
 DateilDatei exportieren... 779
 DateilDatei importieren... 779
 DateilDaten an Microsoft Excel zurückgeben
 508
 DateilDruckbereichIDruckbereich aufheben
 289
 DateilDruckbereichIDruckbereich festlegen
 289
 DateilDrucken... 277
 DateilEigenschaften 147
 DateilEntfernen von 779
 DateilNeu... 91, 327, 459
 DateilÖffnen... 152, 459, 463
 DateilOLAP-Cube erstellen... 512
 DateilSchliessen und Zurückkehren zu...
 570
 DateilSeite einrichten... 76, 281, 611
 DateilSeitenansicht 279
 DateilSenden an 298
 DateilSenden anIE-Mail-Empfänger 298
 DateilSenden anIE-Mail-Empfänger (als
 Anlage)... 299
 DateilSenden anIE-Mail-Empfänger (zur
 Überarbeitung) 45, 299
 DateilSenden anINächster Empfänger...
 300
 DateilSenden anVerteilerempfänger... 300
 DateilSenden anWeitere Empfänger... 300
 DateilSpeichern 139, 463
 DateilSpeichern unter... 151, 463, 541
 DateilSQL ausführen... 465
 DateilSuchen... 1066
 DateilTabellendefinition... 448, 455, 467
 DateilWebseitenvorschau 724
 Datei-Eigenschaften 147
 Dateiende 1046, 1057
 Dateiformat 541

- Dateiformen 1035
- Dateikopie 1050
- Dateilänge 1057
- Dateiname 154, 286
- Dateinummer 1038
- Dateireservierung 300
- Dateityp 143, 154, 327, 1038
- Dateiursprung 547
- Daten|Daten aktualisieren
 - Pivot-Tabellen 492
 - Query 454
 - Webabfragen 741
- Daten|Externe Daten importieren 454
- Daten|Externe Daten importieren|Abfrage bearbeiten... 740
 - Query 454
- Daten|Externe Daten importieren|Daten importieren... 455, 735
- Daten|Externe Daten importieren|Datenbereichs-Eigenschaften... 741
- Daten|Externe Daten importieren|Neue Abfrage erstellen... 447, 512
- Daten|Externe Daten importieren|Neue Webabfrage... 455
- Daten|Externe Daten importieren|Parameter... 742
- Daten|Externe Daten|Neue Webabfrage... 739
- Daten|Filter|Alle anzeigen 426
- Daten|Filter|AutoFilter 426
- Daten|Filter|Spezialfilter... 432
- Daten|Gruppierung und Gliederung|AutoGliederung 348
- Daten|Gruppierung und Gliederung|Detail anzeigen 356
- Daten|Gruppierung und Gliederung|Detail ausblenden 355
- Daten|Gruppierung und Gliederung|Einstellungen... 350
- Daten|Gruppierung und Gliederung|Gliederung entfernen 351
- Daten|Gruppierung und Gliederung|Gruppierung aufheben... 352
- Daten|Gruppierung und Gliederung|Gruppierung... 351, 352
- Daten|Gültigkeit 1003, 1004
- Daten|Maske... 439
- Daten|MS Access-Bericht... 995
- Daten|MS Access-Formular... 994
- Daten|PivotTable- und PivotChart-Bericht... 487
- Daten|Sortieren... 418
- Daten|Tabelle... 521
- Daten|Teilergebnisse... 441
- Daten|Text in Spalten... 546, 550
- Daten|Zu Microsoft Access konvertieren... 995
- Datenarten 206
- Datenaustausch 143
- Datenaustausch über die Zwischenablage 563
- Datenbank 390
 - Zugriff auf externe Datenbanken 445
- Datenbanken 445
- Datenbankfunktionen 389
- Datenbankprogramme, und ODBC 540
- Datenbereichs-Eigenschaften 454
- Datenbeschriftungen 618
- Datenblock 87
- Dateneingabe, in selektierte Bereiche 167
- Datenexport 539
- Datenfeld 490, 857
 - übergeben 860
- Datenfeld-Deklaration 858
- Datenfeldgrenzen 859
- Datenfeldgrenzfunktionen 860
- Datenimport 539
- Datenmasken 438
- Datenpunkt 598, 666
- Datenpunkt formatieren 667
- Datenpunkt selektieren 658
- Datenpunktbild 598
- Datenpunktreihe 598
- Datenquellen, Zugriff auf externe 507
- Datenreihe 598
- Datenreihe einfügen 626
- Datenreihe selektieren 650, 657
- Datenreihe-Formel 650
- Datenreihen 668

- Reihenfolge der 674
- Datenreihenanzordnung 606
- Datenreihenname 618
- Datensätze 417
- Datensätze eingeben 462
- Datensätze Bearbeiten zulassen 461
- Datensätze Gehe zu... 463
- Datensätze Sortieren... 463
- Datenstruktur 466
- Datentyp 381, 387, 548, 826
- Datentyp Object 915
- Datentyp Variant 915
- Datentypen 359, 866
- Datum 68, 286, 365, 366, 828
- Datum/Uhrzeiteingabe 828
- Datumsfilter 432
- Datumsformate 220, 228
- Datumsfunktion 366
- Datumsreihe 193, 194
- Datumstext 366, 367
- Datumswert 1080
- DATWERT 366
- dBase 445
 - Datenaustausch 560
- dBase-Datei direkt importieren 561
- DBFunktionsname 390
- DBFx (dBASE x) 542
- DBMAX 390
- DDE 540, 1079
- Debug.Print 803, 818
- Debuggen 895
- Debuggen|Aktuellen Wert anzeigen... 811
- Debuggen|Alle Haltepunkte löschen 809
- Debuggen|Ausführen bis Cursor-Position 809
- Debuggen|Einzelschritt 806
- Debuggen|Haltepunkt ein/aus 805, 809
- Debuggen|Kompilieren von VBAProjekt 790
- Debuggen|Nächste Anweisung anzeigen 809
- Debuggen|Nächste Anweisung festlegen 809
- Debuggen|Prozedur abschliessen 809
- Debuggen|Prozedurschritt 807
- Debuggen|Überwachung bearbeiten... 813, 815
- Debuggen|Überwachung hinzufügen... 811
- Debugger 803
- Default-Webseite 738
- Deklaration einer Prozedur mit Static 854
- Deklaration einer Stringvariablen 830
- Deklaration einer Variablen mit Static 854
- Deklaration selbstdefinierter Datentypen 867
- Deklarationen 829
- Deklarationsbereich eines Moduls 853, 854
- Deklarationsort 853
- Deklarationszwang 831
- DEM 223
- DEMOS.EXE 1065
- Deskriptive Statistik 369
- Desktop 142
- Detaildaten 348
- Determinante 389
- Dezimalkomma 555, 744
- Dezimalpunkt 555, 744
- Diagramm
 - formatieren 656
 - rotieren 649
- Diagramm|3D-Ansicht... 648
- Diagramm|Daten hinzufügen... 625
- Diagramm|Datenquelle... 611
- Diagramm|Diagramm-Optionen... 614
- Diagramm|Diagrammtyp... 630
- Diagramm|Speicherort... 611
- Diagramm|Trendlinie hinzufügen... 619
- Diagrammarten 630
- Diagramm-Assistent 599
- Diagramm-Beschriftungen 601
- Diagrammblätter 611
- Diagramm-Druckoptionen 612
- Diagramme
 - erzeugen 598
 - Titel 614
- Diagrammelemente 598
- Diagrammfenster 605
- Diagrammfläche formatieren 664
- Diagrammfläche selektieren 657
- Diagrammfunktionalität 730
- Diagrammgröße 611
- Diagramm-Symboleiste 630

- Diagrammtext, ungebundener 619
- Diagrammtiefe 678
- Diagramm-Tipps 656
- Diagrammtitel 599
- Diagrammtyp 600
- Diagrammvariante 600
- Dialogfeld 76
- Dialogfelder
 - objektbezogene 582
 - selbstdefinierte 955
- DIF 542
- Differenz in Tagen 367
- Differenz zwischen zwei Datumsangaben 367
- Differenz zwischen zwei Uhrzeiten 368
- Dim 829, 858
- Dim As 830
- Direktfenster 801, 818
- Direktzugriff 1039
- Direktzugriffs- und Binärdateien 1039
- Direktzugriffsdatei 1035, 1036, 1040, 1050, 1050
- Division 115
- Division durch Null 359, 894
- DLL 1075
- DM 215
- DNS 712, 754, 758
- Dokument, schützen 1006
- Dokument\Drucker anhalten 1069
- Dokumente einbetten 570
- Domänen-Namen-System 712, 754, 758
- Doppelklick 119, 582
- Doppelpfeil 243
- Do-Schleifen 889
- DOS-Konventionen für Pfadangaben und Dateinamen 1038
- Double 828
- DQY 463
- Drag & Drop 169
- Drahtmodell, rotieren 648
- Drehen-Schaltfläche 950
- Drehfeld 77, 967
- Drehung 649
- DropDown-Liste 427
- DropDown-Pfeil 75, 426
- Druck im Hintergrund 1068
- Druckaufträge 1069
- Druckbereich 288
- Drucken 271
- Drucker, einrichten 274
- Drucker\Druckaufträge löschen 1069
- Druckergebnis 278
- Druckerinstallation 271
- Druckerschriftarten 241
- Druckerspeicher 276
- Drucker-Symbol 277
- Druck-Manager 1068
- Druckqualität 282
- Drucktitel 289
- Druckvorgang 1069
- Dynamic data exchange 540
- Dynamische Datenaustausch 540
- Dynamische Datenfelder 861
- E**
- Ebenennummer 350
- Editieren 164, 534
- Editierpasswörter 1008
- Effekte 237
- Eigene Dateien 142
- Eigenschaft ActiveCell 926
- Eigenschaft ActiveChart 926
- Eigenschaft ActiveSheet 926
- Eigenschaft ActiveWindow 926
- Eigenschaft ActiveWorkbook 926
- Eigenschaft ActiveWorksheet 926
- Eigenschaft einen Wert zuweisen 917
- Eigenschaft Let 917
- Eigenschaft Next 909
- Eigenschaft Value 912
- Eigenschaften 785, 908
 - anzeigen/ändern 780
- Eigenschaftfenster 780, 951
- Ein- und Ausgabebereich definieren 1011
- Einbindung einer Grafik über die Zwischenablage 563
- Einblenden 246, 247
- Eindeutiger Index 458
- Einfache externe Bezüge 559
- Einfügen 171
- Einfügen einer Tabelle über die Zwischenablage 562

- Einfügen\Datei... 798
- Einfügen\Diagramm... 599
- Einfügen\Grafik 589
- Einfügen\Grafik\Aus Datei... 589
- Einfügen\Grafik\AutoFormen 591
- Einfügen\Grafik\ClipArt... 593
- Einfügen\Grafik\WordArt... 591
- Einfügen\Hyperlink... 700
- Einfügen\Klassenmodul 780
- Einfügen\Kommentar 136
- Einfügen\Kommentar bearbeiten 136
- Einfügen\Kopierte Zellen... 183
- Einfügen\Namen\Definieren... 331
- Einfügen\Namen\Einfügen... 332, 337
- Einfügen\Namen\Erstellen... 338
- Einfügen\Namen\Übernehmen... 339
- Einfügen\Objekt... 570
- Einfügen\Prozedur... 782, 786, 790, 838
- Einfügen\Seitenumbruch 295
- Einfügen\Seitenumbruch aufheben 295
- Einfügen\Spalten 177
- Einfügen\Symbol... 38, 137
- Einfügen\Tabellenblatt 311
- Einfügen\UserForm 955
- Einfügen\Zeilen 177, 178
- Einfügen\Zellen... 122, 171
- Einfügen-Smart Tag 84, 179, 562
- EingabeDlg 938
- Eingabefelder 77
- Einheiten anzeigen 686
- Einlesen einer Excel-Tabelle in Multiplan 559
- Einrücken, Zellinhalte 250
- Einstellungen 120
- Einzelblatt 274
- Einzelschritt 806
- Einzug 274
- Elemente, löschen 659
- Elementspezifisches Formatier-Dialogfeld 661
- Else 876
- Elseif 879
- Else-Zweig 876
- E-Mail
 - Mappe als Anlage versenden 299
 - Mappe im Textkörper versenden 299
- Mappe versenden 298
- Mappe weiterleiten 298
- Verteiler erstellen 300
- Empfängerliste 300
- End of File 1046
- Ende einer sequentiellen Datei 1046
- Endkapital 388
- Endlospapier 274
- Endlosschleifen 941
- Entf 81, 109
- Entfernen 173
- EntireColumn-Methode 938
- Entscheidung 382, 876
- Entweder...oder 876
- Entwurfsmodus 951
- Entwurfsqualität 290, 613
- Eof-Funktion 1046
- Ereignisprozedur 784, 959
 - erstellen 792
- Ergebnismatrix 533
- Err 897
- Err.Clear 901
- Err.Raise 899
- Error 899
- Error Err 896
- Error-Trapping 895
- Ersetzen 201, 203
- Erste Modellvariante definieren 410
- Erste Seitenzahl 282
- ERW 159
- Esc 69
- EUR 223
- Euro, Währungen umrechnen 1020
- Euro-Symbol 215, 223
- Exakte Textsuche 436
- Excel 4.0-Kompatibilität 1013
- Excel für den Macintosh 1080
- Excel-Arbeitsblatt 107
- Excel-Bibliothek 903
- Excel-Dateien im SYLK-Format 558
- Excel-Fenster, Komponenten 67
- Excel-Konstanten 833
- Excel-Makros 1013
- Excel-Objekte 912
- Exchange 298
- Exemplare 278

- Exit 892
 - Exit Do 892
 - Exit For 892
 - Exit Function 892, 893
 - Exit Sub 892
 - Exit-Anweisung 917
 - Exponent 111
 - Exponentialformat 111, 221
 - Exponentialschreibweise 111
 - Exponentialzahlen 226
 - Exportieren 779
 - Externe Daten 735
 - Externe Verknüpfungswerte speichern 1080
 - Extras\Add-Ins... 1075
 - Extras\Änderungen
 - nachverfolgen\Änderungen annehmen oder ablehnen... 304
 - Extras\Änderungen
 - nachverfolgen\Änderungen hervorheben... 303
 - Extras\Analyse-Funktionen 1010
 - Extras\Anpassen... 72
 - Extras\Arbeitsmappe freigeben... 301
 - Extras\Arbeitsmappen vergleichen und zusammenführen... 306
 - Extras\Assistent\Verweis... 1017
 - Extras\Auto-Korrektur-Optionen... 1073
 - Extras\Euroumrechnung... 1021
 - Extras\Fehlerüberprüfung... 45, 997
 - Extras\Formelüberwachung 1001
 - Extras\Formelüberwachung\Alle Spuren entfernen 1002
 - Extras\Formelüberwachung\Detektiv-Symboleiste anzeigen 996
 - Extras\Formelüberwachung\Formelauswertung 1000
 - Extras\Formelüberwachung\Formelüberwachungsmodus 999
 - Extras\Formelüberwachung\Spur zum Fehler 1001
 - Extras\Formelüberwachung\Spur zum Nachfolger 1003
 - Extras\Formelüberwachung\Spur zum Vorgänger 1002
 - Extras\Formelüberwachung\Überwachungs-fenster anzeigen 998
 - Extras\Makro\Aufzeichnen... 766
 - Extras\Makro\Aufzeichnung beenden 767
 - Extras\Makro\Makros... 767
 - Extras\Makro\Sicherheit... 773
 - Extras\Makro\Visual Basic-Editor 776
 - Extras\Optionen... 1076
 - Register Allgemein 94
 - Register Bearbeiten 109
 - Register Editor 788
 - Extras\Rechtschreibung... 1070
 - Extras\Schutz\Arbeitsmappe schützen und freigeben... 1009
 - Extras\Schutz\Arbeitsmappe schützen... 1008
 - Extras\Schutz\Arbeitsmappenschutz aufheben... 1009
 - Extras\Schutz\Benutzer dürfen Bereiche bearbeiten... 1008
 - Extras\Schutz\Blatt schützen... 1006
 - Extras\Schutz\Blattschutz aufheben... 1006
 - Extras\Schutz\Freigabeschutz aufheben... 1009
 - Extras\Solver... 398
 - Extras\Szenario-Manager 409
 - Extras\Teilsomme... 1013
 - Extras\Tools im Web... 994, 1075
 - Extras\Verweise... 855
 - Extras\Zielwertsuche... 397
 - Extremwertdarstellung 685
- F**
- F1 98
 - F2
 - Abfragen 462
 - Objektkatalog 798
 - F4 127
 - F5 204, 247
 - F9 135, 186
 - Berechnen 1079
 - F10 69
 - Fadenkreuz 578
 - FALSCH 359, 381, 387
 - Farben kopieren aus 1087

- Farbmodell 1086
- Favoriten 142
- Fehlerbehandlung 895
- Fehlerbetrag 671
- Fehlercode 898, 900, 901
- Fehlerdefinition 899
- Fehlerindikator Y 670
- Fehlerindikatoren-Kontextmenü 673
- Fehlermeldung 332, 1005
- Fehler-Objekt 897
- Fehlerroutine 895
- Fehler-Smart Tag 84, 117, 227, 997
- Fehlerüberprüfung 997
- Fehlerverfolgung 895
- Fehlerwert 117, 290, 359
- Feld 68, 417
- Feldnamen 417
 - einschließen 454
- Feldnummer 390
- Feldüberschriften 390
- Fenster 85
 - fixieren 317
- Fenster der aktiven Arbeitsmappe 93, 315
- Fenster|Anordnen... 92, 315
- Fenster|Ausblenden 315
- Fenster|Einblenden... 315
- Fenster|Fixierung aufheben 317
- Fenster|Neues Fenster 93, 314
- Fenster|Teilen 316
 - Modulfenster 797
- Fenster|Teilung aufheben 317
- Fensteranordnung
 - horizontal 93
 - vertikal 94
- Fenstergröße 90
- Fenstermodi 88
- Fensterstapel 92
- Fernverweise 1079
- Feste Breite 551, 554
- Feste Dezimalstelle setzen 1080
- Feste Spaltenbreiten 551
- Filterkriterien 429
- Filtern 426
- Finanzmathematik 387
- FINDEN 361
- Find-Methode 936
- Firmennetzwerk 746
- Flächendiagramme 635
- Flächendiagramm-Varianten 635
- Fließtext 578
- Fokus 970
- For 885
- For Each...Next-Schleife 887
- For...Next-Schleife 884
- Format »Standard« 230
- Format von Zelle wählen... 202
- Format|AutoFormat... 262
- Format|Bedingte Formatierung... 266, 269
- Format|Blatt|Ausblenden 310
- Format|Blatt|Einblenden... 310
- Format|Blatt|Hintergrund... 256
- Format|Blatt|Umbenennen 310
- Format|Formatvorlage... 257
- Format|Schriftart... 463
- Format|Spalte 245
- Format|Spalte|Ausblenden 246
- Format|Spalte|Breite... 245
- Format|Spalte|Einblenden 246
- Format|Spalte|Optimale Breite bestimmen... 245
- Format|Spalte|Standardbreite... 245
- Format|Spalten ausblenden 463
- Format|Spalten einblenden... 463
- Format|Spaltenbreite... 463
- Format|Zeile 245
- Format|Zeile|Höhe... 245
- Format|Zeilenhöhe... 463
- Format|Zellen... 216, 236, 247, 252
- Formate 559
 - landesspezifische 228
 - selbst definieren 232
- Formatier-Dialogfelder 659
- Formatieren 214
- Formatier-Symbole 216
- Formatierungs-Eigenschaften 942
- Formatierungs-Leiste 214
- Formatierungs-Schemata 262
- Formatierungs-Voreinstellungen, ändern 261
- Formatkomponenten 258
- Formatvorlage 257
 - enthält 258

Stichwortverzeichnis

- Standard 258
- zuweisen 351
- Formel 108
 - Auswertungen 1000
 - testen 134, 523
 - Überwachungsmodus 999
 - verbergen 1008
- Formel im Diagramm darstellen 623
- Formelauswertungsfenster 1000
- Formeleingabe 118
- Formulare 955
- Formularereignis 959
- FoxPro 445
- Freizugebende Mappe schützen 1009
- FRM 779
- Frontpage 98-CD 745
- Frontpage Server Extensions 714
- FTP-Adresse 712
- FTP-Server 705, 711, 712, 718
 - Anmeldekenntwort 711
 - Anmeldezugriffsvoraussetzungen 711
- FTP-Zugriff 727
- Funktion
 - Error 899
 - finanzmathematische 387
 - Hyperlink 701
 - logische 380
 - mathematische 362
 - numerische 834
 - selbstdefinierte 786
 - statistische 369
 - String 834
 - trigonometrische 388
- Funktionsargumente 357
- Funktions-Assistent 128
- Funktionsaufruf 358
 - verschachtelt 358
- Funktionsgruppen 362
- Funktionsname 789
- Funktionsprozeduren 784, 838
 - erstellen 785
- Funktionstyp 840
- Funktionswert 357, 360, 789, 840
- Fußzeile 283, 284

G

- Ganze Seite verwenden 613
- Ganzzahlig 363
- Gebietsschema 228
- Gehe zu 204
- Geklammerter Ausdruck 115
- Geltungsbereich von Konstanten 833, 854
- Geltungsbereiche 852
 - Konstanten 852
 - Prozeduren 851
 - Variablen 852
- Geltungsbereichsregeln 917
- Gemischte Bezüge 127
- Genauigkeit 1079
- Geometrisch 200
- Gesamtaussage 385, 386
- Gesamte Arbeitsmappe 278
- Gesamtergebnisse 492
- Gesperrt 1007
- Get 1051, 1060
- Gitternetzlinien 68, 253, 256, 290, 1077
- Glättung 372
- Gleitender Durchschnitt 621
- Gliederung
 - automatische 347
 - manuelle 351
- Gliederungen 347
- Gliederungsebene 348, 350
- Gliederungsfunktion 348
- Gliederungs-Symbole 351, 1077
- GoTo 874
- Grafik
 - einfügen 589
 - integrieren 563
 - Komprimierung 760
- Grafikdatei 589
- Grafik-Objekt 587
 - Register Abstände 584
 - Register Ausrichtung 584
 - Register Eigenschaften 586
 - Register Schrift 584
 - Register Schutz 585
 - Register Web 587
 - selektieren 580
 - Register Größe 585

Grenzen 402
 Grenzwert 400
 Größe ändern 91
 Größen in umgekehrter Reihenfolge 689
 Größenachse 599, 689
 Größenachse (Y) schneidet bei größter
 Rubrik 692
 Größenachse (Y) schneidet zwischen
 Rubriken 692
 Größenachse... 692
 GROSS 360
 Großbuchstaben 360
 Große Schaltflächen 986
 GroupName 974
 Gruppe 308, 347
 gruppieren 441
 Gruppierung beginnen 993
 Gruppierungs-Symbol 581
 Gültigkeitsmeldung 1004
 Gültigkeitsprüfung 208
 Gültigkeitsregel 1004
 Güte der Regression 373

H

Häufigkeitsauszählung 485
 Haltepunkt 806
 Hauptgitternetz 616
 Hauptintervall 686
 Hauptspalten rechts von Detaildaten 351
 Hauptstriche 683
 Hauptzeilen 348
 Hauptzeilen unter Detaildaten 351
 HEUTE() 231
 Hide 978
 Hilfgitternetz 616
 Hilfsintervall 686
 Hilfsstriche 683
 Hintergrund 581
 Hintergrund aller Zellen 256
 Hochformat 276, 281
 Höchstwert 685
 Horizontal 284
 Horizontaler Umbruch 293
 Hosts 749
 HTM 721
 HTML 721
 HTML-Format 543

HTML-Formatierung 740
 HTML-Tabellen 738
 Hyperlink 82, 147, 699
 Adresse 701
 auf aktuelle Excel-Mappe 700
 auf E-Mail-Adressen 708
 auf neue Dokumente 708
 auf Textmarke 705
 auf Webseiten 707
 Bearbeiten 702
 Einfügen 700
 Entfernen 702
 Felder 699
 Funktion 701
 Kontextmenü 702
 öffnen 702
 Ziel 700, 704, 707
 zuweisen... 994
 Hyperlink-Funktion 702
 Hypertext Markup Language 721

I

If 874, 876
 If-/Else-Blöcke 878
 If-Blöcke 877
 If-Then-Else 876
 If-Then-Else-Blockstruktur 878
 Im Web suchen 102
 Importieren 779
 Importierte Abfragedaten 454
 INDEX 374
 Index
 Array 857
 relative Positionsangabe 376
 Suchfunktionsresultat 375
 Zellen 927
 Index erstellen 458
 Indexfelder 458
 Indirekte Abhängigkeiten 211
 Indizes 859
 Infofeld 96
 Informationsfunktionen 387
 Inhalte
 auswählen 186
 einfügen 185
 löschen 174
 Inhalte... 206

Stichwortverzeichnis

Inkrement 198
Innenringgröße 680
Input 1039, 1041, 1043
InputBox 822
Input-Funktion 1044
Installation 60
 Installationsprogramm 60
InStr 837
INT 363
Int 835
Integer 827
Integrierte Menüs 988
Interaktives Layouten 280
Interaktivität 728
Interne Speicherung 369
Internet Explorer 751
Internet Information Server 745
Internetbrowser 723
Internet-Speicherort 713
Interpolation 370
Intranet 707, 724
Inverse 389
Investition 388
IP-Adresse 749
IPX/SPX 747
Iqy 735
Is 883
IsMissing 847
IST... 387
ISTTEXT 387
ISTZAHL 387
Iterationen 1079

J

JAHR 368
Jahresreihe 195
Jahresübergänge 194
Jahreszahlung 387
JETZT() 231
JPEG 761

K

Kassetten 276
Kategorienname 618
Kegeldiagramme 643

Kegelvarianten 643
Keine Linie 583
Klassenmodul 792, 857, 954, 959
 öffnen 792
Kleinstwert 685
Kollision 302
Kombination zweier Bereichsnamen 345
Kombinationsfeld 967
Kommentar 136, 206, 290, 824
Kommentar anzeigen 137
Kommentar ausblenden 137
Kompilieren einer Prozedur 790
Kompilierung, bedingte 881
Komponenten eines Verbunds 869
Konstante 359
 Geltungsbereich 852
 iterale 832
 symbolische 832
 vordefinierte 833
Konstantenname 832
Kontextmenü 70, 70, 164
Kontextsensitiv 96
Kontrollkästchen 78, 967
Konvertieren von Lotus-Tabellen 560
Konvertierung 827
Koordinaten 121
Koordinatensystem 121, 187, 373
Kopf- und Fußzeilen 280
Kopfzeile 283, 284, 417
Kopieren 179, 313
Kopieren absoluter Formeln 125
Kopieren relativer Formeln 124
Kreis- und Ringdiagramm-Optionen 679
Kreisdiagramme 635
Kreisdiagramm-Varianten 636
Kreissegment 636
Kreiszahl Pi 388
Kreuz-Tabelle 485
Kriterien 440
KriterienAlle Kriterien entfernen 465
KriterienKriterien hinzufügen... 464
Kriterienbereich 433
Kursdiagramm-Varianten 645
Kursdifferenzen 679
Kursverlauf 678

L

Label 874
 LÄNGE 361
 Länge der Zeichenkette 361
 LAN 724
 Landesspezifische Formate 228
 Laserdrucker 283
 Laufrahmen 119
 Laufzeitfehler 893
 Laufzeit-Fehlercode 897
 Layouten 280
 Lbound 860
 Leer 109
 Leerzeichenfolgen, variable 552
 Leerzeichen-Sequenzen 540, 556
 Leerzellen 206
 Left 836
 Legende 599
 Legende formatieren 666
 Legende frei verschieben 665
 Legendenspezifisches Kontextmenü 660
 Legenden-Symbol 618
 Legendentexte 665
 Len 835
 Len-Funktion 1053
 length of file 1057
 Lese-/Schreibanweisungen 1051
 Lese-/Schreibkennwort 145
 Leseanweisung 1043
 Lesezeichen 794
 Let-Anweisung 825
 Letzte Zelle 207
 Line Input 1041
 Linealzeile 552, 555
 Linear 687
 Lineare Regression 373, 621
 Linefeed 543
 Liniendiagramme 634
 Liniendiagramm-Optionen 678
 Liniendiagramm-Varianten 634
 Linke Spalte 479
 LinkedCell 952
 LINKS 364
 Linksbündig 247
 Liste einfügen 337
 Liste zuletzt geöffneter Dateien 152

Listenfeld 967
 Mehrfachselektion 78
 Listenformat und Formeln erweitern 1081
 Local Area Network 724
 Loc-Funktion 1059
 Lock Read 1040
 Lock Read Write 1040
 Lock Write 1040
 Löschen 439
 Format suchen 202
 Lösen 401
 Lösung verwenden 402
 Lösungsmodelle 407
 Lof-Funktion 1057
 Logarithmische Skalierung 688
 Logische Datenstrukturen 1056
 Lokale Ausdrücke 806
 Lokale Variablen 842
 Lokalfenster 806
 Long 827
 LOOKUP 559
 Lotus 559
 Lotus- und Symphony-Funktionen 559
 Lupen-Symbol 279

M

Macintosh 1080
 Macintosh-Datei 547
 Makro 765
 ausführen 768
 mit Objekten verknüpfen 770
 Sicherheit 773
 Sicherheitsstufe einstellen 773
 zuweisen... 770, 770, 993
 Makrobeschreibung 766
 Makroname 766
 Makro-Recorder 765, 767
 Manipulation von Programmdateien 1059
 Mantisie 111
 Manuell 1079
 Manuelle Gliederung 351
 Manuelle Zufuhr 274
 Mappe.xlt 262, 329
 Markiermodus, erweiterter 159
 Markierung 277

Stichwortverzeichnis

- Markierung nach dem Drücken der
 - Eingabetaste verschieben 109
- Maske 439
- Maskieren 380
- Matrix 359, 376, 377
- Matrixbereiche 534
- Matrixexpansion 535
- Matrixformel 372, 530
- Matrixfunktionen 389
- Matrixkonstante 359, 370, 535
- Matrizen 370, 389
- Max 399
- Maximale Änderung 1079
- Maximale Iterationszahl 1079
- Maximale Satzlänge 1040
- Maximieren 91
- Mehrdimensionale Arrays 864
- Mehrdimensionale Datenfelder 864
- Mehrfachoperationen 519
- Mehrfachoperationsbereich 524
- Mehrfachselektion 159
- Mehrfachtests 882
- Mehrfaktorielle lineare Regression 374
- Menü 70
- Menüleiste 67, 68
- Menüs, kontextsensitive 70
- Methode Activate 908
- Methode Cells 910, 927
- Methode Copy 944
- Methode End 946
- Methode Paste 944
- Methode Range 927
- Methode Select 928
- Methode Value 937
- Methoden 785, 908
- Microsoft Excel 4.0-Arbeitsmappe 542
- Microsoft Excel-Add-In (*.xla) 815
- Microsoft Excel-Arbeitsmappe 143
- Microsoft Exchange 298
- Microsoft Office Web Components 728
- Microsoft Query 445
- Microsoft Query-Add-In 447
- Mid 836
- Minimieren 91
- Minuszeichen 235
- Mit Quelldaten verknüpft 683
- Mit-Anweisung 914
- MITTELWERT 128, 129
- Mittelwerte bilden 134
- MMULT 389
- Modell laden... 406
- Modell speichern... 406
- Modul
 - Deklarationsbereich 853
 - Name ändern 780
- Modulebene 784
- Modulfenster 777
 - teilen 797
- Modus 1038
- MONAT 368
- Monatsreihe 195
- MS-DOS 547
- MsgBox 896
- Msqry32.exe 447, 455
- MSQuery 561
- Multiplan 542
 - Datenaustausch 558
- Multiplikation 115
- Multiseiten 967
- Muster 255
- Musterformatierungen 255
- Mustervorlage 240, 262, 327, 1022
- Mustervorlagenverzeichnis 1022
- N**
- Nachfolger-Zellen 211
- Name 334
 - ersetzen 341
 - erstellen 338
 - übernehmen 339
- Namenslistenfeld 336
- Nebenbedingungen festlegen 400
- Nebeneinander 93
- NetBEUI 747
- Netzdiagramme 638
- Netzdiagramm-Varianten 638
- Netzwerk 746, 1038, 1040
 - Adapter 748
 - Computernamen 746
 - Eigenschaften 746
 - IP-Adresse 749
 - IPX/SPX 747
 - NetBEUI 747
 - Protokolle 747

Rechnernamen 749
 Subnet-Mask 749
 TCP/IP 747
 Netzwerkverbindung 746
 Neu 439
 Neu berechnen 1078, 1079
 Neuberechnung der Tabelle 363
 Neue Prozedur 786
 Neue Seite 969
 Neue_x_Werte 370
 Neues Menü 988
 Neukalkulation 1078
 Next 885
 Nicht ausführbare Anweisungen 784
 Notebook-Tastaturen 158
 Nothing 916
 NT-Server 749
 Nullwerte 1077
 Null-Zeiger 908
 Numerische Achsenbeschriftungen
 formatieren 682
 Numerische Konstante 381
 Nummernzeichen 111
 Nur direkte 211
 Nur sichtbare Zellen 207
 Nur Text (immer) 992
 Nur Text (in Menüs) 992
 Nur von Zellposition abhängig 587

O

Oberflächendiagramme 640
 Oberflächendiagramm-Varianten 640
 Object
 Datentyp 828
 Schlüsselwort 915
 Objekt 207, 785
 drucken 587
 einbetten 569
 entfernen 779
 erstellen 571
 ex- und importieren 779
 Objekt.Eigenschaft 909
 Objekt.Eigenschaft = Wert 910
 Objekt.Methode 908
 Objekt-Auflistungen 905

Objektbezüge speichern 915
 Objektbezug 908, 915
 Objekt-Bibliothek 833
 Objektdatentypen 915
 Objekte, auswählen 967
 Objekteigenschaften 780, 910
 Objekthierarchie 903
 Objektkatalog 798, 834, 911
 Objekt-Listenfeld 792
 Objektmanipulation 582
 Objektorientiert 784
 Objektränder 582
 Objektrahmen 583
 Format 583
 Objekt-Überlappung 581
 Objektvariablen 914
 ODBC 445, 540
 ODBC-Treiber 446
 ODER 386
 Öffentlich 851
 Öffentliche Variablen 853
 Öffnen 152
 Office-Assistent 97
 Office-Zwischenablage 563
 OK 79
 OLAP
 Cube 511
 Cube-Abfragedatei 516
 Cube-Assistent 513, 516
 Cube-Definition 514
 Cube-Dimensionen 513
 Cube-Typ 514
 Offline Cube 512
 Server 511
 On Error 895
 On Error Resume Next 897
 On-Line Analytical Processing 511
 Online-Hilfe 775
 Open-Anweisung 1038
 Operator 114
 Optimale Breite 244
 Optimale Höhe 244
 Option Base 859
 Option Explicit 831
 Optional 234, 362

Optionale Argumente 846
Optionen
 zur Berechnung von Datumsreihen 197
 zur Berechnung von Zahlenreihen 199
Optionsfeld 967
OQY 516
Ordner 142
Outlook 298
Outlook Express 298
Outlook Express-Integration 299
Output 1039

P

Paletten-Listenfeld 75
Papierformat 281, 282
Papiergröße 275
Papierlänge 291
Papierzufuhr 275
ParamArray 866
Parameter 408
Parameterabfrage 737
Parameter-Array 866
Parameterwert 737
Passwort 1006
PDS 798
Persönliche Makro-Arbeitsmappe 767
Personal Web Server 745
 Installation 751
Personal Web-Manager 752
Personl.xls 767
Perspektive 648, 649
Perspektivenkontrolle 649
Pfad und Dateiname 286

 Bildlauf 87
 Menü 69

 Bildlauf 87
 Menü 69

 Menü 69
 Bildlauf 87
 Dialogfeld 81

 Bildlauf 87
 Dialogfeld 81
 Menü 69

Pi 389
Pivot
 Beschriftungsfelder 498
 Daten zeigen als 496
 Datenfelder 495
 Hauptrubrik 500
 Unterrubrik 500
 Vergleichs-Optionen 496
 Zusammenfassen mit 496
Pivot-Tabelle 485
Pivot-Tabellen- und Diagramm-Assistent 487
PivotTable-Feldliste 489
PivotTable-Funktionalität 730
Pixelmuster 992
Pos./Neg. Abweichung 678
Positionierung auf einen Recorder 1058
Preserve 863
Previous 910
Primärschlüssel 419
Print 1041, 1044
Print-Methode 818
PrintPreview 955
Privat 851, 858, 859, 918
PRODUKT 362
Prognose 372
Programmgesteuerter Zugriff auf Excel-Tabellen 920
Programmierungsumgebung 776
Programmschleifen 884
Programmspezifischer Fehler 900
Projekt 777
Projekte speichern 779
Projektebene 777
Projekt-Explorer 777
Property Get 917
Property Set 917
Property-Prozeduren 916
Prozeduraufrufe 841
Prozedurbezüge 854
Prozedurebene 784
Prozeduren
 Argumente 844
 Funktion 838
 Geltungsbereiche 851
 integrierte 834
 Neu 786

- objektspezifische 908
- selbstdefinierte 838
- Sub 838
- Prozedur-Listenfeld 782, 791
- Prozedurrumpf 786
- Prozentformat 224, 225
- Prozentindikator 672
- Prozentsatz 618
- Prozentzeichen 111
- Pseudozufallszahl 363
- Public 851, 853, 858, 859, 917
- Puffer 1040, 1041
- Punkt vor Strich 115
- Punktdiagramme 639
- Punktdiagramm-Varianten 639
- Put 1051, 1060
- Pyramidendiagramme 644
- Pyramiden-Varianten 644

Q

- QBasic 798
- Quadratwurzel 363
- Querformat 276, 281
- Query 445
- QuickBasic 798
- QuickInfo anzeigen 987

R

- Ränder 280
- Rahmen 252, 967
- Rahmen zeichnen... 44
- Rahmenformatierungen 252
- Rahmenlinien zeichnen... 254
- Raise 899
- Randeinstellungen 283
- Random 1051
- Range-Methode 940
- Range-Objekt 912
- Rechenvorschrift 108, 357
- Rechts nach Links orientierte Sprachen 250
- Rechtsbündig 247
- Rechtschreibprüfung 1070
- Rechtwinklige Achsen 650
- Recordlänge 1040, 1051
- Recordnummer 1050
- Records 866

- Recordvariable 867
- ReDim 858, 861
- Redimensionieren von Arrays 861
- RefEdit 968
- Referenzübergabe 847
- Regelmäßige Zahlung 387
- Register 967
 - Achsen 668, 669
 - Allgemein 152, 1083
 - Ansicht 1077
 - Ausrichtung 247, 251, 681
 - AutoAusfüllen 1083
 - Bearbeiten 109, 1080
 - Berechnen 1078
 - Datenbeschriftung 667
 - Datenreihen 674
 - Diagramm 1084
 - Eigenschaften 664
 - Farbe 1086
 - Fehlerindikator Y 671
 - Fehlerüberprüfung 1088
 - International 1087
 - Kontextmenü 310
 - Kopfzeile/Fußzeile 284
 - Muster 255, 662, 683
 - Optionen 622, 675
 - Papierformat 281
 - Platzierung 665
 - Rahmen 252
 - Rechtschreibung 1088
 - Registerkarte 78
 - Registerkartenzungen 78
 - Schrift 237, 663
 - Seitenränder 283
 - Sicherheit 1090
 - Skalierung 685
 - Speichern 1088
 - Tabelle 288
 - Umsteigen 1081
 - Zahlen 220
- Registerkartenleiste 308
- Registerleiste 94, 94
- Regression 199, 370
- Regressionsanalysen, mehrfaktorielle
 - lineare 370
- Regressionsfunktion 623

- Regressionstyp 620
- Reihen, berechnen 196
- Reihen- und Spaltenkopfzeilen 1077
- Reihenfolge der Kategorien 480
- Reihentyp 200
- Relative Bezüge 120, 125
- Relative Positionsangabe 376
- Relative/Absolute Bezugsart ignorieren 341, 342
- Rem 824
- Resume 896
- Resume Next 896
- Resume Zeile 896
-  Im selektierten Bereich 168
- Menü 69
- RGP 374
- Richtungsknöpfe 648
- Right 836
- Ringdiagramme 637
- Ringdiagramm-Option 680
- Ringdiagramm-Varianten 637
- Ringsegment 638
- RKP 374
- Rmz 387
- Rotationsachse 648
- Rotieren, Zellen 249
- Rubrik einfügen 628
- Rubriken in umgekehrter Reihenfolge 692
- Rubrikenachse 599, 686
- Rubrikenachse (X) schneidet bei Höchstwert 689
- Rubrikenachsenbeschriftungen 680
- Rubrikenanzahl zwischen Teilstrichbeschriftungen 692
- Rubrikenbeschriftung 692
- Rubrikenteilstriche 692
- Rückgängig machen 164
-  81
- S**
- Säulenabstände 676
- Satzlänge 1037, 1038, 1040, 1051
- Satznummer 1037, 1051
- Satzzeiger 1058
- Schätzen 404
- Schaltflächensymbol ändern 991
- Schaltflächensymbol bearbeiten... 991
- Schaltflächensymbol einfügen 992
- Schaltflächensymbol kopieren 991
- Schaltflächensymbol und Text 992
- Schaltflächensymbol zurücksetzen 992
- Schleife 884
- Schleifenanfang 876
- Schleifenbedingung 889
- Schleifenende 876
- Schleifenkonstruktionen 884
- Schleifenvariable 885
- Schließen 91
- Schnittpunkt, impliziter 341
- Schnittpunkt = 623
- Schnittpunkt zwischen der Rubriken- und der Größenachse 692
- Schnittpunkt-Operator 344
- Schreibschutz 147
- Schreibschutz empfehlen 147
- Schreibschutz-Empfehlung 147
- Schreibschutz-Kennwort 146
- Schreibzugriff 910, 1041
- Schrift 236
- Schriftformate 236
- Schriftkassetten 276
- Schrittweite 198, 885
- Schutz 1007
- Schwarzweißdruck 613
- Schwarzweißzellen 291
- Seek-Anweisung 1058
- Seek-Funktion 1058
- Seite löschen 969
- Seitenanzahl 286
- Seitenanzeige 505
- Seitenlayout 281
- Seitennummer 286
- Seitenränder 280, 283, 291
- Seitenreihenfolge 291
- Seitenumbruch 292, 1077
- manueller 291
- Seitenunterteilung 291
- Seitenvariable auswählen 506
- Sekundärachse 668

- Sekundärachsen 615
- Sekundärschlüssel 419
- Selbstdefinierte Dialogfelder 955
- Selbstdefinierte Formate 233
- Selbstdefinierte Spaltenbreiten 555
- Select 882
- Selektieren 656
 - von Bereichen 158
- Selektion eines Objekts in einer Gruppe 658
- Selektionskästchen 175
- Selektionsmaske 154
- Sensitivität 402
- Sequentiell 1035
- Sequentielle Dateien 1035, 1040, 1041
 - Lese-/Schreibanweisungen 1041
- Sequentielle und Binärdateien 1059
- Serielle Zahl 365, 1080
- Servername 754
- Serviceprovider 746
- Set 914
- Shared 1040
- Show 961
- Show-Methode 961
- Sicherungsdatei erstellen 144
- SIN 388
- Single 827
- Sinus-Funktion 388
- Skalierbar 242
- Skalierung 281
 - logarithmisch 687
- Smart Tag 84, 133, 1074, 1081
 - Aktienkurse 53, 743
 - AutoAusfüllen 190, 196
 - Einfügen 84, 179, 562
 - Fehler 84, 117, 227, 997
 - Indikator 84, 1074
 - Listensymbol 1074
- Solver-Dialogfelds 404
- Sonderformate 228
- Sonst_Wert 382
- Sortierkriterien 418
- Sortier-Optionen 423
- Sortierreihenfolge 425
- Sozialversicherungsnummer 228
- Spalte 68, 107
 - Selektion 160
- Spalte, Zeile 345
- Spalten
 - ausblenden 246
 - Behandlung 177
 - nicht importieren (überspringen) 549
- Spaltenbezeichnung 108, 290
- Spaltenbreite 243, 280
- Spaltenindex 376, 377, 911, 927
- Spaltennamen 345
- Spaltenorientierte Gliederung 350
- Spalten-Standardbreite 245
- Spaltenunterschiede 208
- Spannweitenlinien 678
- Speicherbedarf eines mit Type deklarierten Datentyps 1053
- Speicherkapazität 91
- Speichern 139
- Speichern unter 140
 - Webseite 542
- Speicherplatz 529
- Sperrung 1040
- Spezialfilter 432
- Sprachen, Rechts nach Links orientierte 250
- Sprung 874
- SQL-Statements 446
- Sqr 835
- Standard 992
- Standardabweichung 369, 672
- Standarddatentyp 826
- Standard-Diagrammtyp 653
- Standard-Druckformat 240
- Standardfehler 672
- Standardformatierungen 258
- Standard-Formatvorlage 262
- Standardmodus 1039
- Standardsatzlänge 1040
- Standardschaltfläche 821
- Standardschrift 240
- Standardspeicherort 1083
- Standardvorlage 327
- Standardwert 362
- Standardwörterbuch 1072
- Startordner 1083
- Startseite 703
- Static 851, 853, 858, 859, 918

Stichwortverzeichnis

- Statische Variablen 853
 - Statistik 369
 - Statistische Funktionen 476
 - Statusleiste 68, 135, 1077
 - Steuerelement-Eigenschaften 965
 - Steuerelement-Toolbox 951
 - Str 838
 -  159
 -  +  168
 -  +  87
 -  +  81
 -  +  ↓ 94
 -  +  ↑ 94
 -  + Buchstabe 69
 -  +  87
 -  +  161
 -  +  87
 -  +  87
 -  + , Dialogfeld 81
 -  + 
 - Bildlauf 87
 - Dialogfeld 81
 -  + , Bildlauf 87
 -  +  +  525
 -  +  +  161
 -  +  +  530
 -  +  81
 - String fester Länge 828
 - String variabler Länge 828
 - Stringfunktionen 364, 834
 - Stringkonstante 832
 - Strings 824
 - Stringvariablen 830
 - Stringvariablen fester Länge 831
 - Strukturen 866
 - Strukturierte Daten 1036
 - Sub- und Ereignisprozeduren testen 802
 - Subnet-Mask 749
 - Sub-Prozedur 784
 - erstellen 790
 - Subtraktion 115
 - Suchbegriff 203
 - Suchen 201
 - Suchfunktion 374, 375
 - Suchkriterienbereich 390
 - Suchkriterium 374, 375, 377, 390
 - Suchmatrix 375
 - Suchmöglichkeiten 440
 - Suchpfad 154
 - Suchseite 703
 - Suchstring 380
 - Suchtext 436
 - Summe, Symbol 133, 134
 - SUMME-Funktion 130, 357
 - SVERWEIS 1028
 - SYLK 542
 - SYLK-Format 558
 - Symbolabstand und -breite 990
 - Symbolbild bearbeiten 992
 - Symbolische Konstanten 832
 - Symbolleiste 67, 71, 214
 - an Mappe anbinden 989
 - anbinden 989
 - Eingebaute Menüs 988
 - Neues Menü 988
 - Rahmenlinie 44, 254
 - Symbolleisten-Menü 72
 - Symphony 559
 - Syntaxanalyse 787
 - Syntaxangabe 362
 - Syntaxüberprüfung 788
 - Systemmenü 90
 - Systemsteuerung 237
 - Szenarien-Übersichtsbericht 414
 - Szenario speichern... 402
 - Szenario-Manager 408
 - Szenario-PivotTable-Bericht 414
- ## T
- 
 - Anweisung 819
 - Dialogfeld 80
 - Im selektierten Bereich 168
 - Tab_Repl.Exe 556
 - Tabelle definieren 457
 - Tabelle einfügen 311
 - Tabelle verschieben 311
 - Tabelle.xlt 240, 329
 - Tabell(Tabellen entfernen 471
 - Tabell(Tabellen hinzufügen... 471
 - Tabell(Verknüpfungen... 471
 - Tabellen-Autovorlage 329

- Tabellenbereich 158, 321
- Tabellenblatt 68
- Tabellenfenster 68
- Tabellenname 286
- Tabellenübergreifende Editierungen 307
- Tabellenübergreifende Konsolidierungen 475
- Tab-Schrittweite 788
- Tabulator 556
- Tabulatoren als Trennzeichen 541
- Tabulatorzeichen 540, 556
- TAG 368
- Tagesreihe 195
- Taskleiste 1077
- Tastenfolge 765
- Tastenschlüssel 767
- TCP/IP 747
- Teamarbeit 298
 - Änderungen ablehnen 304
 - Änderungen akzeptieren 304
 - Änderungen protokollieren 304
 - Änderungen verfolgen/rückgängig machen 303
- Exchange 298
- Mailprogramme 298
- Mappe als Anlage versenden 299
- Mappe freigeben 300
- Mappe im Textkörper versenden 299
- Mappe versenden 298
- Mappe weiterleiten 298
- Mappen zusammenführen 306
- Outlook (Express) 298
- Verteiler erstellen 300
- Teilaussage 385
- Teilauswertung 135
- Teilergebnisse 442
 - gruppiert 441
- Teilstrichbeschriftung 684
- Teilungslinie 797
- Testwerte 521
- Text
 - formatierter 543
 - mit Tabulatortrennung 543
- Textausrichtung 247
- Textdatei, Ausgabe 1047
- Textdatei-Import 545
- Texterkennungszeichen 548
- Textfeld 967
- Textformat 227, 544
- Textfunktionen 364
- Textkonvertierungs-Assistent 545, 547
- Then-Zweig 876
- Ticker-Symbole 737
- Titelleiste 67
- Top 10-AutoFilter 428
- Traktor 274
- Transponieren 187, 389
- Trend 199, 200, 370
- Trendberechnung 199
- Trendgerade 372
- Trendlinien 619
- Trennzeichen 546, 547, 1043
- Trennzeichenanzahl 548
- TrueType-Schriften 241, 242
- Typ Range 926
- Typdeklarationszeichen 827
- TYPE 1044
- Type 866, 1053
- Typkonvertierung 827
- U**
- UBound 860
- Übergabe
 - als Referenz 847
 - als Wert 849
- Übergabe einer Stringvariablen 846
- Überlappend 92, 676
- Überwachte Ausdrücke anzeigen 812
- Überwachung hinzufügen 46, 316, 811, 998
- Überwachungsart 813
- Überwachungsausdruck 813
- Überwachungsfenster 316, 811, 998
- Überwachungsort 812
- Uhrzeit 286, 365, 828
- Umbruchlinien 292, 1077
- Umlaute 547
- 
 - Dialogfeld 81
 - Drag&Delete 176

- Selektieren 157
- Suchen 203
-  +  139
-  +  798
-  +  161
-  +  161
-  +  168
-  +  168
- Umschaltfeld 967
- Unabhängige Variable 370
- UND 385, 386
- Uniform Ressource Locator 715
- Unsicherheitsbereich 670
- Unterbereich 343
- Unterbrechen, wenn der Wert True ist 813
- Unterbrechen, wenn Wert geändert wurde 815
- Unterbrechung, wenn definiertes Ereignis eintritt 814
- Untereinander 93
- Unterfenster 316
- Untergeordnete Spaltenrubriken 499
- Untermenü 69
- Unterschlüssel 420
- Unterstreichung 236
- Unterteilt 92
- Until 890
- Unverankerte Leisten 73
- URL-Adresse 715
- USD 223
- V**
- Val 1048
- Value 966
- Variablen 485, 824
 - Deklarationen 829
 - Deklarationsort 853
 - deklarieren 830
 - Geltungsbereiche 852
 - lokale 842
 - private 853
 - Referenzübergabe 847
 - Wertübergabe 849
- Variablendeklaration erforderlich 831
- Variablenname 825
- Variablenübergabe 844
- Variablenübergabe als Referenz 848
- Variant 826, 828, 914
- Varianz 369
- VARIATION 374
- Variation der unabhängigen Parameter 399
- vb... 833
- VBA 774, 799
 - Projekt 856
- VBA-Konstanten 833
- VBA-Objekt-Bibliothek 834
- VBA-Programm 777
- VBA-Programmierungsumgebung 776
- vbOKCancel 834
- vbOKOnly 834
- vbQuestion 834
- Veränderbare Zelle 397, 399
- Verbindungs-Dialogfeld 735
- Verbindungslinien 676
- Verborgene Fenster 315
- Verbunde 866
- Verbundvariable 867, 1052
- VERGLEICH 374
- Vergleichsoperator 236, 380, 381, 429, 430
- Vergleichs-Optionen 496, 498
- Vergleichsspalte 208
- Vergleichstyp 375, 376
- Vergleichswert 209
- Vergrößern 318
- Vergrößerungsmodus 279
- Verkleinern 318
- Verkleinern/Vergrößern 282
- Verknüpfen 189
- Verknüpfen mit Quelldaten 482
- Verknüpfte Bereiche 189
- Verknüpfte Tabellen 465
- Verknüpfungen
 - zum Inhalt 149
 - dynamische 480
 - statische 474
- Verknüpfungswerte 1080
- Verlängerungsmodus 161
- Verlauf 142
- Veröffentlichungs-Optionen 728
- Verschachtelung von Funktionsaufrufen 359
- Verschachtelungen 940
- Verschieben 91, 169, 969

Verschieben der Bezugzelle 123
 Verschlüsselung 146
 Verschlüsselungsverfahren 146
 Vertikal 284
 Vertikaler Umbruch 293
 VERWEIS 374, 559
 Verweis 855
 Verzeichnisse, virtuelle 755
 Verzeichnisverwaltung 755
 Verzweigung 874
 Vielwertig 531
 Virtuelle Verzeichnisse 755
 Visible 966
 Visual Basic 798
 Sprachelemente 818
 Visual Basic für Anwendungen 774, 799
 Visual Basic-Editor 776
 Von Scanner oder Kamera... 589
 Von Zellposition und -größe abhängig 587
 Von Zellposition und -größe unabhängig 587
 Vordergrund 581
 Vorgänger-Zellen 211
 Vorherigen suchen 439
 Vorortaktivierung 573
 Vorschau 236
 Vorselektiert 143
 Vorzeichen 111, 363

W

Währungsformate 223, 223
 Währungs-Symbol 215, 223
 Währungstext 223
 Wagenrücklauf 1043
 WAHR 359, 381, 387
 Wahrheitsprüfung 383
 Wahrheitswert 359, 381, 386
 WAN 724
 Was wäre wenn 408
 Web
 Adresse 718
 Basisverzeichnis 716
 Browser 728
 Default-Webseite 738
 DNS 712, 754, 758
 Domänen-Namen-System 712, 754,

758
 Frontpage Server Extensions 714
 FTP-Adresse 712
 FTP-Server 711, 712, 718
 FTP-Server-Anmeldekennwort 711
 FTP-Server-
 Anmeldezugriffsvoraussetzungen 711
 FTP-Zugriff 727
 Internetbrowser 723
 Internet-Speicherort 713
 Intranet 724
 IP-Adresse 749
 Microsoft Office Web Components 728
 Optionen 759
 Personal Web Server 751
 Serviceprovider 746
 Symboleiste 702
 TCP/IP 747
 Uniform Ressource Locator 715
 URL-Adresse 715
 Verknüpfungen zum 718
 Webbrowser 723
 Webbrowser-Kompatibilität 760
 Webordner 711
 WWW-Server 711
 Web Archiv 542, 724, 727
 Webabfrage 735
 Aktualisieren 741
 ausführen 735
 editieren 738
 Eigenschaften 741
 Einstellungen 739
 erstellen 738
 Formatierung 739
 Parameter 742
 Parameterabfrage 737
 Parameterwert 737
 Verbindungs-Dialogfeld 735
 vorgefertigte 735
 Webseitenabschnitte 739
 Webbrowser 696, 723, 728, 735
 Internet Explorer 751
 Kompatibilität 760
 Webordner 142, 711, 714
 Suche in 1068

Stichwortverzeichnis

- Webseite 543, 721
 - Active Server Pages 757
 - als Web-Archiv speichern 727
 - Arbeitsmappenfunktionalität 730
 - Besuchte 707
 - Bilder komprimieren 760
 - Diagrammfunktionalität 730
 - Firmennetzwerk 746
 - HTM 721
 - HTML 721
 - Hypertext Markup Language 721
 - Interaktivität 728
 - Netzwerkverbindung 746
 - PivotTable-Funktionalität 730
 - Speichern als 727, 759
 - Veröffentlichungs-Optionen 728
 - Web-Archiv 724
 - XML-Format 728
 - XML-Kalkulationstabelle 728
 - XSL Stylesheet 728
- Webserver 745
 - Alias-Namen 756
 - Basisverzeichnis 716, 754, 755
 - dezidiert 745
 - Eigenschaften 752
 - Installation testen 754
 - Internet Information Server 745
 - Intranetadresse 754
 - Name des 754
 - Personal Web Server 745
 - Personal Web-Manager 752
 - Software 745
 - Standard-Dokument 758
 - Verwaltung 755
 - Verwaltungs-Optionen 752
 - virtuelle Verzeichnisse 755
 - Wwwroot 754
- Wegziehbare Paletten 75, 989
- Weitersuchen 203, 439
- WENN 382
- Werkzeugsammlung 956
- Wert 618
- Wert anzeigen 811
- Wert einer Eigenschaft übergeben 917
- Werte aus Spalte 521
- Werte aus Zeile 523
- Wertefunktion auswählen 495
- Wertematrix 531
- Wertübergabe 849
- While-Schleife 889
- Wide Area Network 724
- Wiederherstellen 91
- Wiederholen 164
- Wiederholungsspalten 289
- Wiederholungszeilen 289
- Windows 905, 921
 - Netzwerkeigenschaften 747
- Windows_95/98 745
- Windows_NT 745
- Windows-Netzwerk-Name 754
- Windows-Programm 562
- Winkel des ersten Kreissegments 680
- Winkelangabe in Grad 388
- Wissenschaftliches Format 226
- With 868
- WOCHENTAG 366, 368
- Wörterbuch 1072
- Word 565
- Workbooks 905
- Worttrennung 578
- Write 1041, 1043
- WURZEL 363
- Wwwroot 754
- X**
 - x_Werte 370
 - X-Achse 599
 - xl... 833
 - XLA 1075
 - XLB 990
 - xlDown 946
 - XLL 1075
 - XLODBC.XLA 446
 - Xlstart 240, 262, 326, 329, 767, 769, 1075
 - xlToLeft 946
 - xlToRight 946
 - xlUp 946
 - XLW 151
 - XML-Format 728

XML-Kalkulationstabelle 542, 728

XSL Stylesheet 728

Name: 990

Y

y_Werte 370

Y-Achse 599

Z

Z1S1 121, 210

Z1S1-Bezugsart 120

Z1S1-Format 211

Zählvariable 885

Zahlen 824

Zahlenausrichtung 247

Zahlenformate 220, 222, 559

Zahlenkonstanten 832

Zahlungsperiode 387

Zahlungszeiträume 387

Zehnerpotenz 689

Zeichenfläche 599

Zeichenfolge 68

Zeichenkette 362, 824, 828

Zeichenketten miteinander vergleichen 382

Zeichenprogramm 577

Zeichnen-Leiste 577

Zeichnungsfläche formatieren 662

Zeigemodus 119

Zeile 68, 107

 Behandlung 177

 Selektion 160

Zeile, Spalte 345

Zeilen- und Spaltenköpfe 290

Zeilen- und Spaltennamen verwenden 342

Zeilenbezug 132

Zeilenende 1043, 1044

Zeilenende-Markierung 1043

Zeilengruppen 352

Zeilenhöhe 243

Zeilenindex 376, 911, 927

Zeilenmarke 874

Zeilennamen 345

Zeilennummer 108, 290, 874

Zeilennummern einschließen 454

Zeilenumbruch 111, 249, 285, 578, 818, 820

Zeilenunterschiede 208

Zeilenvorschub 543, 1043

ZEIT 368

Zeitachse 616

Zeiteinheit 198

Zeitformate 220, 231

Zeitfunktionen 368

Zellbezug 118, 359

Zellcursor 168

Zelle 68, 107

 aktive 168

 nach links verschieben 173, 173

 nach rechts verschieben 172

 nach unten verschieben 173

Zellen rotieren 249

Zellen-Kontextmenü 165

Zellfarben 290

Zellhintergrund 255

Zellinhalt 110

Zellinhalte einrücken 250

Zellkommentar 206

Zellname 334

Zellreferenz 108, 118

Zellschutz 1008

Zellselektion 157

Zentriert über Markierung 248

Ziehen und Ablegen 169

Zielbereich 475, 550

Zielwert 397

Zielwertsuche 396

Zielzelle 397, 399

Ziffernblock 68

Zins 387, 388

Zinssatz 387

Zoom 279

Zoom-Funktion 318

Zufallszahl 363

Zugriff 1038

Zugriff auf eine Array-Variable 857

Zugriffsarten 1039

Zugriffsform 1038

Zugriffsmodus 1039

Zukünftiger Wert 387

Zurücksetzen... 989

Zuweisungsausdruck 826

Stichwortverzeichnis

ZW	388	Zwischenergebnisse	265
Zwang zur Deklaration	831	Zwischenraum	677
Zweidimensionales Array	864	Zylinderdiagramme	642
Zwischenablage	110, 561, 562, 944	Zylinderdiagramm-Varianten	642
Office	563	Zzr	387