

Auf einen Blick

Einführung	23
Teil I: Excel VBA – die ersten Schritte	29
Kapitel 1: Was ist VBA?	31
Kapitel 2: Ein Sprung ins kalte Wasser	39
Teil II: Die Zusammenarbeit von VBA und Excel	51
Kapitel 3: Der Visual Basic Editor	53
Kapitel 4: Das Objektmodell in VBA	73
Kapitel 5: Sub- und Funktionsprozeduren in VBA	87
Kapitel 6: Mit dem Excel Makro-Recorder arbeiten	101
Teil III: Programmierkonzepte	113
Kapitel 7: Wichtige Elemente der Sprache VBA	115
Kapitel 8: Mit Range-Objekten arbeiten	135
Kapitel 9: VBA- und Arbeitsblattfunktionen	149
Kapitel 10: Steuerung des Programmablaufs und Entscheidungen	163
Kapitel 11: Automatische Prozeduren und Ereignisse	181
Kapitel 12: Techniken zur Fehlerverarbeitung	201
Kapitel 13: Techniken, mit denen Sie Fehler loswerden	215
Kapitel 14: VBA-Programmierbeispiele	229
Teil IV: Kommunikation mit Ihren Benutzern	251
Kapitel 15: Einfache Dialogfelder	253
Kapitel 16: UserForm-Grundlagen	269
Kapitel 17: UserForm-Steuerelemente verwenden	287
Kapitel 18: UserForms – Techniken und Tricks	307
Kapitel 19: Zugriff auf Ihre Makros über die Benutzeroberfläche	335
Teil V: Das große Ganze	347
Kapitel 20: Eigene Funktionen für Arbeitsblätter erstellen	349
Kapitel 21: Add-Ins in Excel erstellen	369
Teil VI: Der Top-Ten-Teil	381
Kapitel 22: Zehn nützliche Tipps für den Visual Basic Editor	383
Kapitel 23: Zehn Orte, an denen Sie Hilfe zu Makros erhalten	391
Kapitel 24: Zehn Dinge, die Sie beherzigen oder vermeiden sollten	397
Stichwortverzeichnis	401



Inhaltsverzeichnis

Einführung	23
Haben Sie das richtige Buch gewählt?	23
Konventionen in diesem Buch	24
Überprüfen Sie Ihre Sicherheitseinstellungen	25
Törichte Annahmen über die Leser	26
Symbole, die in diesem Buch verwendet werden	26
Beispieldateien herunterladen	27
Wie es weitergeht	27
TEIL I	
EXCEL VBA – DIE ERSTEN SCHRITTE	29
Kapitel 1	
Was ist VBA?	31
Was also ist VBA?	31
Was ist mit VBA möglich?	32
Text einfügen	33
Eine häufig ausgeführte Aufgabe automatisieren	33
Automatisierung wiederholter Operationen	33
Einen benutzerdefinierten Befehl anlegen	33
Eine benutzerdefinierte Schaltfläche erstellen	34
Neue Funktionen für die Arbeitsmappe entwickeln	34
Benutzerdefinierte Add-Ins für Excel erstellen	34
Vorteile und Nachteile von VBA	34
Vorteile von VBA	34
Nachteile von VBA	35
VBA in Kürze	35
Excel-Kompatibilität	38
Kapitel 2	
Ein Sprung ins kalte Wasser	39
Das Erste zuerst	39
Was auf Sie zukommt	40
Die ersten Schritte	40
Das Makro aufzeichnen	41
Das Makro testen	43
Das Makro überprüfen	43
Das Makro abändern	45
Arbeitsmappen speichern, die Makros enthalten	46

12 Inhaltsverzeichnis

Makro-Sicherheit	46
Weitere Informationen über das Makro NameUndZeit.	48
TEIL II	
DIE ZUSAMMENARBEIT VON VBA UND EXCEL	51
Kapitel 3	
Der Visual Basic Editor	53
Was ist der Visual Basic Editor?	53
Den VBE aktivieren.	53
VBE-Komponenten.	54
Arbeiten im Projektfenster.	56
Ein neues VBA-Modul hinzufügen	57
Ein VBA-Modul entfernen	57
Objekte exportieren und importieren.	58
Arbeiten in einem Codefenster	59
Fenster minimieren und maximieren	59
Ein Modul erstellen	60
VBA-Code in ein Modul einfügen	61
Code direkt eingeben	61
Verwendung des Makro-Recorders.	64
VBA-Code kopieren	66
Anpassung der VBA-Umgebung	67
Die Registerkarte Editor	67
Die Registerkarte Editorformat	70
Die Registerkarte Allgemein	71
Die Registerkarte Verankern	72
Kapitel 4	
Das Objektmodell in VBA	73
Ist Excel ein Objekt?	73
Navigation innerhalb der Objekthierarchie.	74
Alles dreht sich um Collections	75
Verweise auf Objekte – Referenzen	75
Navigation innerhalb der Hierarchie.	76
Objektreferenzen vereinfachen	77
Objekteigenschaften und Methoden	78
Objekteigenschaften	79
Objektmethoden	81
Objektereignisse.	82
Weitere Informationen	83
Das Hilfesystem von VBA	83
Der Objektkatalog	84
Automatische Auflistung von Eigenschaften und Methoden	85

Kapitel 5
Sub- und Funktionsprozeduren in VBA **87**

- Sub-Prozeduren im Vergleich zu Funktionsprozeduren 87
 - Sub-Prozeduren 88
 - Funktionsprozeduren 88
 - Namen für Sub-Prozeduren und Funktionsprozeduren 89
- Sub-Prozeduren ausführen 90
 - Die Sub-Prozedur direkt ausführen. 92
 - Die Prozedur vom Dialogfeld Makro aus ausführen. 92
 - Ein Makro mithilfe einer Tastenkombination ausführen 93
 - Die Prozedur über eine Schaltfläche oder eine Form ausführen 94
 - Die Prozedur von einer anderen Prozedur aus ausführen 96
- Funktionsprozeduren ausführen 97
 - Die Funktionsprozedur aus einer Sub-Prozedur aufrufen. 97
 - Aufruf einer Funktion von einer Formel in einer Tabelle aus 98

Kapitel 6
Mit dem Excel Makro-Recorder arbeiten **101**

- Grundlagen der Aufzeichnung. 101
- Vorbereitung auf die Aufzeichnung 103
- Relativ oder absolut? 103
 - Aufzeichnung im absoluten Modus. 104
 - Aufzeichnung im relativen Modus. 105
- Was wird eigentlich aufgezeichnet? 106
- Aufzeichnungsoptionen 107
 - Makro-Name. 108
 - Tastenkombination 108
 - Makro speichern in 108
 - Beschreibung 109
- Und wie effizient ist das Ganze? 109

TEIL III
PROGRAMMIERKONZEPTE **113**

Kapitel 7
Wichtige Elemente der Sprache VBA **115**

- Kommentare in Ihrem VBA-Code 115
- Variablen, Konstanten und Datentypen 117
 - Variablen verstehen. 117
 - Was sind Datentypen in VBA? 119
 - Variablen deklarieren und ihnen einen Gültigkeitsbereich zuweisen 120
 - Mit Konstanten arbeiten 126
 - Vorgefertigte Konstanten 127
 - Mit Zeichenfolgen arbeiten. 127

14 Inhaltsverzeichnis

Mit Datumswerten arbeiten	128
Zuweisungsbefehle verwenden	129
Beispiele für den Zuweisungsbefehl	129
Das Gleichheitszeichen	130
Hilfreiche Operatoren	130
Arrays	131
Arrays deklarieren	132
Mehrdimensionale Arrays	132
Dynamische Arrays	133
Die Verwendung von Zeilenmarken (Labels)	134

Kapitel 8

Mit Range-Objekten arbeiten **135**

Ein schneller Überblick	135
Andere Methoden, auf einen Bereich zu verweisen	137
Die Cells-Eigenschaft	137
Die Offset-Eigenschaft	138
Einige praktische Eigenschaften des Range-Objekts	139
Die Value-Eigenschaft	139
Die Text-Eigenschaft	140
Die Count-Eigenschaft	141
Die Column- und Row-Eigenschaften	141
Die Address-Eigenschaft	141
Die HasFormula-Eigenschaft	142
Die Font-Eigenschaft	143
Die Interior-Eigenschaft	143
Die Formula-Eigenschaft	145
Die NumberFormat-Eigenschaft	146
Einige praktische Methoden des Range-Objekts	146
Die Select-Methode	146
Die Methoden Copy und Paste	147
Die Clear-Methode	148
Die Delete-Methode	148

Kapitel 9

VBA- und Arbeitsblattfunktionen **149**

Was ist eine Funktion?	149
Eingebaute VBA-Funktionen verwenden	150
Beispiele für VBA-Funktionen	150
VBA-Funktionen, die mehr tun, als einen Wert zurückzugeben	152
VBA-Funktionen finden	153
Arbeitsblattfunktionen in VBA verwenden	156
Beispiele für Arbeitsblattfunktionen	156
Eingabe von Arbeitsblattfunktionen	159
Weitere Informationen über die Verwendung von Arbeitsblattfunktionen	160
Benutzerdefinierte Funktionen verwenden	160

Kapitel 10
Steuerung des Programmablaufs und Entscheidungen..... 163

Schwimmen mit dem Strom 163
 Die GoTo-Anweisung 163
 Entscheidungen, Entscheidungen! 165
 Die If-Then-Struktur 166
 Die Select-Case-Struktur 170
 Ihr Code für eine Schleife 173
 For-Next-Schleifen 173
 Do-While-Schleifen 178
 Do-Until-Schleifen 178
 For-Each-Next-Schleifen für Collections ausführen 179

Kapitel 11
Automatische Prozeduren und Ereignisse 181

Vorbereitung auf das große Ereignis 181
 Sind Ereignisse praktisch? 183
 Programmierung von Prozeduren zur Ereignisverarbeitung 183
 Wo steht der VBA-Code? 184
 Eine Prozedur zur Ereignisverarbeitung schreiben 185
 Einführende Beispiele 186
 Das Open-Ereignis für eine Arbeitsmappe 186
 Das BeforeClose-Ereignis für eine Arbeitsmappe 189
 Das BeforeSave-Ereignis für eine Arbeitsmappe 189
 Beispiele für Aktivierungsereignisse 190
 Aktivierungs- und Deaktivierungsereignisse in einem Arbeitsblatt 190
 Aktivierungs- und Deaktivierungsereignisse in einer Arbeitsmappe 191
 Arbeitsmappen-Aktivierungsereignisse 193
 Weitere Arbeitsblattereignisse 193
 Das BeforeDoubleClick-Ereignis 194
 Das BeforeRightClick-Ereignis 194
 Das Change-Ereignis 195
 Ereignisse, die keinen Objekten zugeordnet sind 197
 Das OnTime-Ereignis 197
 Tastenaktionen 199

Kapitel 12
Techniken zur Fehlerverarbeitung 201

Fehlertypen 201
 Ein fehlerbehaftetes Beispiel 202
 Das Makro ist nicht ganz perfekt 203
 Das Makro ist immer noch nicht perfekt 204
 Ist das Makro immer noch nicht perfekt? 204
 Verabschieden Sie sich von der Perfektion! 206
 Fehler anders verarbeiten 206

16 Inhaltsverzeichnis

Noch einmal die Prozedur EnterSquareRoot	206
Die On-Error-Anweisung	207
Fehlerverarbeitung: Die Details	208
Fortsetzung nach einem Fehler	208
Fehlerverarbeitung im Überblick	210
Erkennen, wann Fehler ignoriert werden können	210
Spezifische Fehler identifizieren	211
Ein beabsichtigter Fehler	212

Kapitel 13

Techniken, mit denen Sie Fehler loswerden 215

Verschiedene Fehlerarten	215
Programmierfehler identifizieren	216
Debugging-Techniken	217
Ihren Code überprüfen	218
Die MsgBox-Funktion verwenden	218
Debug.Print-Anweisungen einfügen	220
Verwendung des VBA-Debuggers	221
Der Debugger	221
Haltepunkte in Ihrem Code setzen	221
Das Überwachungsfenster verwenden	224
Das Lokal-Fenster	226
Tipps, wie Sie weniger Programmierfehler machen	226

Kapitel 14

VBA-Programmierbeispiele 229

Mit Bereichen arbeiten	229
Einen Bereich kopieren	230
Einen variablen Bereich kopieren	231
Auswahl bis zum Ende einer Zeile oder Spalte	232
Eine Zeile oder Spalte auswählen	233
Einen Bereich verschieben	233
Einen Bereich effizient durchlaufen	234
Einen Bereich effizient durchlaufen (Teil II)	236
Einen Zellwert anfordern	236
Den Auswahltyp bestimmen	237
Eine Mehrfachauswahl erkennen	238
Excel-Einstellungen abändern	238
Boolesche Einstellungen ändern	239
Nicht-Boolesche Einstellungen ändern	239
Diagramme	240
AddChart im Vergleich zu AddChart4	241
Den Diagrammtyp ändern	242
Die ChartObjects-Collection durchlaufen	243
Diagrammeigenschaften ändern	243
Diagrammformatierungen anwenden	244
VBA-Geschwindigkeitstipps	245

Die Bildschirmaktualisierung abschalten	245
Die automatische Berechnung ausschalten	246
Lästige Warnungen deaktivieren	247
Objektreferenzen vereinfachen	247
Variablentypen deklarieren	248
Die With-End-With-Struktur	248

**TEIL IV
KOMMUNIKATION MIT IHREN BENUTZERN 251**

**Kapitel 15
Einfache Dialogfelder 253**

Alternativen zu UserForms	253
Die MsgBox-Funktion	254
Ein einfaches Meldungsfeld anzeigen	255
Eine Antwort aus einem Meldungsfeld ermitteln	255
Meldungsfelder anpassen	257
Die InputBox-Funktion	259
Die Syntax von InputBox	260
Ein Beispiel für eine InputBox	260
Ein weiterer Typ InputBox	262
Die Methode GetOpenFilename	263
Die Syntax für die GetOpenFilename-Methode	263
Ein Beispiel für GetOpenFilename	264
Die GetSaveAsFilename-Methode	265
Einen Ordnernamen ermitteln	266
Eingebaute Excel-Dialogfelder anzeigen	267

**Kapitel 16
UserForm-Grundlagen 269**

Erkennen, wann man ein UserForm braucht	269
UserForms erstellen: Ein Überblick	270
Mit UserForms arbeiten	271
Ein neues UserForm einfügen	271
Einem UserForm Steuerelemente hinzufügen	272
Die Eigenschaften eines UserForm-Steuerelements ändern	272
Das Codefenster für ein UserForm anzeigen	274
Ein UserForm anzeigen	275
Informationen aus einem UserForm verarbeiten	275
Ein UserForm-Beispiel	276
Das UserForm anlegen	276
Die Befehlsschaltflächen hinzufügen	277
Optionsfelder einfügen	278
Prozeduren zur Ereignisverarbeitung einfügen	280
Ein Makro anlegen, das das Dialogfeld anzeigt	281
Das Makro für den Benutzer bereitstellen	282
Das Makro testen	283

Kapitel 17	
UserForm-Steuerelemente verwenden	287
Steuerelemente im Dialogfeld – die ersten Schritte	287
Steuerelemente hinzufügen	288
Eigenschaften von Steuerelementen – Grundlagen	289
Steuerelemente in einem Dialogfeld: Die Details	290
Das CheckBox-Steuerelement – Kontrollkästchen	291
Das ComboBox-Steuerelement – Kombinationsfeld	292
Das CommandButton-Steuerelement – Befehlsschaltfläche	293
Das Frame-Steuerelement – Rahmen	294
Das Image-Steuerelement (Bildfeld)	294
Das Label-Steuerelement (Bezeichnungsfeld)	295
Das ListBox-Steuerelement (Listenfeld)	296
Das MultiPage-Steuerelement – Multiseiten	297
Das OptionButton-Steuerelement – Optionsfelder	297
Das RefEdit-Steuerelement	298
Das ScrollBar-Steuerelement – Bildlaufleiste	299
Das SpinButton-Steuerelement – Drehfeld	300
Das TabStrip-Steuerelement – Register	300
Das TextBox-Steuerelement – Textfeld	301
Das ToggleButton-Steuerelement – Umschaltfeld	302
Die Arbeit mit Steuerelementen in Dialogfeldern	302
Steuerelemente verschieben und ihre Größe ändern	302
Steuerelemente ausrichten und Abstände einhalten	303
Tastaturbenutzern helfen	303
Ein UserForm testen	305
Dialogfeld-Ästhetik	306
Kapitel 18	
UserForms – Techniken und Tricks	307
Dialogfelder verwenden	307
Ein UserForm-Beispiel	308
Das Dialogfeld erstellen	308
Code für die Anzeige des Dialogfelds	311
Das Makro bereitstellen	311
Ihr Dialogfeld ausprobieren	312
Prozeduren zur Ereignisverarbeitung einfügen	312
Die Daten auswerten	314
Jetzt funktioniert das Dialogfeld	315
Ein Listenfeld-Beispiel	315
Ein Listenfeld füllen	316
Das ausgewählte Element bestimmen	317
Mehrfachauswahlen bestimmen	318
Einen Bereich auswählen	320
Mehrere Optionsfeldgruppen verwenden	321
Ein Drehfeld und ein Textfeld verwenden	322
Ein UserForm als Fortschrittsanzeige verwenden	324

Ein Dialogfeld als Fortschrittsanzeige einrichten.	325
Die Prozeduren.	326
Wie dieses Beispiel funktioniert.	327
Ein nicht modales Dialogfeld mit Registerkarten erstellen.	328
Ein Diagramm in einem UserForm anzeigen.	330
Dialogfeld-Checkliste.	332

Kapitel 19
Zugriff auf Ihre Makros über die Benutzeroberfläche 335

Das Menüband anpassen.	335
Das Menüband manuell anpassen.	335
Dem Menüband ein Makro hinzufügen.	338
Anpassung des Menübands mit XML.	338
Kontextmenüs anpassen.	343
Dem Kontextmenü Cell einen neuen Eintrag hinzufügen.	344
Was hat sich seit Excel 2007 geändert?	346

TEIL V
DAS GROßE GANZE 347

Kapitel 20
Eigene Funktionen für Arbeitsblätter erstellen 349

Wozu benutzerdefinierte Funktionen?	349
Grundlagen der VBA-Funktionen.	350
Funktionen schreiben.	351
Mit Funktionsargumenten arbeiten.	352
Eine Funktion ohne Argumente.	352
Eine Funktion mit einem Argument.	353
Eine Funktion mit zwei Argumenten.	355
Eine Funktion mit einem Bereichsargument.	356
Eine Funktion mit einem optionalen Argument.	358
Wrapper-Funktionen.	359
Die NumberFormat-Funktion.	360
Die ExtractElement-Funktion.	360
Die SayIt-Funktion.	361
Die IsLike-Funktion.	361
Funktionen, die ein Array zurückgeben.	362
Ein Array mit Monatsnamen zurückgeben.	362
Eine sortierte Liste zurückgeben.	363
Das Dialogfeld Funktion einfügen verwenden.	364
Die Funktionsbeschreibung anzeigen.	364
Argumentbeschreibungen.	366

Kapitel 21
Add-Ins in Excel erstellen 369

Und was ist ein Add-In?	369
Wozu legt man Add-Ins an?	370

Die Arbeit mit Add-Ins.	371
Add-In-Grundlagen	372
Ein Add-In-Beispiel.	373
Einrichtung der Arbeitsmappe	373
Die Arbeitsmappe testen	375
Beschreibende Informationen hinzufügen	376
Den VBA-Code schützen	377
Das Add-In anlegen	377
Das Add-In öffnen	378
Das Add-In weitergeben	379
Das Add-In ändern	379

**TEIL VI
DER TOP-TEN-TEIL..... 381**

**Kapitel 22
Zehn nützliche Tipps für den Visual Basic Editor..... 383**

Kommentarblöcke verwenden	383
Mehrere Codezeilen kopieren	385
Zwischen Modulen und Prozeduren springen	385
Beamten Sie sich zu Ihren Funktionen	385
In der richtigen Prozedur bleiben	386
Den Code schrittweise ausführen	386
Zu einer bestimmten Codezeile springen	387
Die Codeausführung an einer bestimmten Stelle unterbrechen	388
Den Anfang und das Ende des Variablenwertes anzeigen	389
Die automatische Syntaxüberprüfung ausschalten	389

**Kapitel 23
Zehn Orte, an denen Sie Hilfe zu Makros erhalten 391**

Lassen Sie Excel die Makros für Sie schreiben	391
Verwenden Sie die VBA-Hilfedateien	392
Stibitzen Sie Code im Internet	392
Userforen optimal nutzen	393
Besuchen Sie Experten-Blogs	394
Suchen Sie auf YouTube nach Schulungsvideos	394
Nehmen Sie an Online-Kursen teil	395
Vom Office Dev Center lernen	395
Analysieren Sie andere Excel-Dateien in Ihrem Unternehmen	395
Fragen Sie die Excel-Experten vor Ort	396

**Kapitel 24
Zehn Dinge, die Sie beherzigen oder vermeiden sollten 397**

Deklariieren Sie alle Variablen	397
Verwechseln Sie Passwörter nicht mit Sicherheit	398
Bereinigen Sie Ihren Code	398
Schreiben Sie nicht alles in eine Prozedur	399

Inhaltsverzeichnis 21

Denken Sie daran, dass es noch andere Software gibt 399
Gehen Sie nicht davon aus, dass alle Benutzer Makros aktivieren 399
Gewöhnen Sie sich an, zu experimentieren 399
Gehen Sie nicht davon aus, dass Ihr Code in anderen Excel-Versionen
funktioniert. 400
Denken Sie immer an Ihre Benutzer. 400
Vergessen Sie die Sicherung nicht! 400

Stichwortverzeichnis 401

