

LERNEN EINFACH GEMACHT



4. Auflage

Excel-VBA

für
dummies[®]



Für alle Excel-Versionen geeignet, auch für Microsoft 365

Eigene Arbeitsblattfunktionen erstellen, Aufgaben automatisieren

Mit Fehlern im Code umgehen und sie beseitigen








Dick Kusleika

Excel-VBA für Dummies

Schummelseite

STANDARDTASTENKOMBINATIONEN FÜR DEN VISUAL BASIC-EDITOR

Mit den folgenden Tastenkombinationen können Sie schnell im Visual Basic-Editor navigieren.

Drücken Sie	Um diese Aktion auszulösen
	Zwischen Excel-Fenster und Visual Basic-Editor wechseln
	Kontextmenü des aktiven Fensters anzeigen (entspricht einem Rechtsklick mit der Maus)
	Projekt-Explorer öffnen
	Eigenschaftfenster öffnen
	Objektbrowser öffnen
	VBA-Hilfe öffnen
	Geöffnetes Modul-Fenster aktivieren

TASTENKOMBINATIONEN FÜR DAS BEARBEITEN VON CODE IM CODEFENSTER

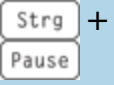

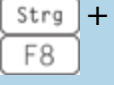


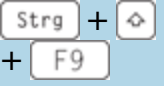
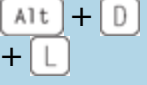
Früher oder später werden Sie zur gleichen Zeit an mehreren Excel-Makros arbeiten. Mit der Maus zwischen und innerhalb der Makro-Prozeduren zu navigieren, kann etwas umständlich sein. Die folgenden Tastenkombinationen ermöglichen es, schnell zu einer Prozedur zu springen, sich in den Modulen zu bewegen und sogar die Stelle zu finden, an der eine bestimmte Variable deklariert wurde.

Drücken Sie	Um diese Aktion auszulösen
Strg + ↓	Nächste Prozedur auswählen
Strg + ↑	Vorige Prozedur auswählen
Strg + Bild ↕	Bildschirminhalt um eine Seite nach unten bewegen
Strg + Bild ↕	Bildschirminhalt um eine Seite nach unten bewegen
↕ + F2	Zur ausgewählten Funktion oder Variablen gehen
Strg + ↕ + F2	Zur letzten Position zurückgehen
Strg + Pos1	Zum Anfang des Moduls gehen
Strg + Ende	Zum Ende des Moduls gehen
Strg + →	Einfügemarke wortweise nach rechts bewegen
Strg + ←	Einfügemarke wortweise nach links bewegen
Ende	Einfügemarke an das Ende der Zeile bewegen
Pos1	Einfügemarke an den Anfang der Zeile bewegen
↔	Aktuelle Zeile einrücken
↕ + ↔	Einrückung der aktuellen Zeile entfernen
Strg + J	Eigenschaften und Methoden des aktuellen Objekts anzeigen

TASTENKOMBINATIONEN FÜR DAS DEBUGGEN VON CODE IM VISUAL BASIC-EDITOR








Das Debuggen Ihres Codes ist ein wichtiger Aspekt bei der Programmierung von Excel-Makros. Sie können auf die Debugging-Features zwar auch über das Menü des Visual Basic-Editors zugreifen, jedoch kommen Sie mit den folgenden Tastenkombinationen schneller zum Ziel.

Drücken Sie	Um diese Aktion auszulösen
F5	Aktuelle Prozedur ausführen oder deren Ausführung fortsetzen, falls sie unterbrochen wurde

Drücken Sie	Um diese Aktion auszulösen
	Ausführung der aktuell laufenden Prozedur unterbrechen
	Debug-Modus aktivieren und Code zeilenweise ausführen
	Code bis zur aktuellen Cursorposition ausführen
	Im Debug-Modus die aktuelle Zeile überspringen
	Haltepunkt für die aktuelle Zeile ein- beziehungsweise ausschalten
	Alle Haltepunkte entfernen
	Aktuelles Visual-Basic-Projekt kompilieren

TASTENKOMBINATIONEN ZUR NAVIGATION IM PROJEKT-EXPLORER DES VISUAL BASIC-EDITORS

Wollen Sie in Ihren Visual-Basic-Projekten navigieren, ohne dazu die Maus zu verwenden? Excel stellt Ihnen genau diese Möglichkeit zur Verfügung. Probieren Sie die folgenden Tastenkombinationen aus, um zwischen Projekten und Modulen zu navigieren.

Drücken Sie	Um diese Aktion auszulösen
	Voriges Element in der Projektliste auswählen
	Nächstes Element in der Projektliste auswählen
	Zur ersten Datei in der Projektliste springen
	Zur letzten Datei in der Projektliste springen
	Ausgewählten Ordner erweitern
	Ausgewählten Ordner reduzieren
	Codefenster für die ausgewählte Datei öffnen





Dick Kusleika

Excel-VBA für dummies®

4. Auflage

Übersetzung aus dem Amerikanischen
von Rainer G. Haselier

WILEY
WILEY-VCH GmbH

Excel-VBA für Dummies

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

4. Auflage 2022

© 2022 Wiley-VCH GmbH, Boschstraße 12, 69469 Weinheim, Germany

Original English language edition Excel VBA Programming For Dummies © 2022 by Wiley Publishing, Inc. All rights reserved including the right of reproduction in whole or in part in any form. This translation published by arrangement with John Wiley and Sons, Inc.

Copyright der englischsprachigen Originalausgabe Excel VBA Programming For Dummies © 2022 by Wiley Publishing, Inc.

Alle Rechte vorbehalten inklusive des Rechtes auf Reproduktion im Ganzen oder in Teilen und in jeglicher Form. Diese Übersetzung wird mit Genehmigung von John Wiley and Sons, Inc. publiziert.

Wiley, the Wiley logo, Für Dummies, the Dummies Man logo, and related trademarks and trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries. Used by permission.

Wiley, die Bezeichnung »Für Dummies«, das Dummies-Mann-Logo und darauf bezogene Gestaltungen sind Marken oder eingetragene Marken von John Wiley & Sons, Inc., USA, Deutschland und in anderen Ländern.

Das vorliegende Werk wurde sorgfältig erarbeitet.
Dennoch übernehmen Autoren und Verlag für die
Richtigkeit von Angaben, Hinweisen und Ratschlägen
sowie eventuelle Druckfehler keine Haftung.

Print ISBN: 978-3-527-71959-4

ePub ISBN: 978-3-527-83812-7

Coverfoto: © Moon Safari - stock.adobe.com

Korrektur: Claudia Lötschert, Neuss

Über den Autor

Dick Kusleika arbeitet seit mehr als 20 Jahren mit Microsoft Office. Er war 12 Jahre in Folge Microsoft MVP (Most Valuable Professional) und hat mehrere Bücher über Excel und Access geschrieben.

Widmung

Für Nancy.

Danksagung

Dank an Rod Stephens für das Aufspüren von Fehlern und für wertvolle Anregungen.

Inhaltsverzeichnis

Cover

Titelblatt

Impressum

Über den Autor

Widmung

Einführung

Über dieses Buch

Törichte Annahmen über die Leser

Symbole, die in diesem Buch verwendet werden

Wie es weitergeht

Teil I: Excel VBA - die ersten Schritte

Kapitel 1: VBA kennenlernen

Die VBA-Grundlagen verstehen

Wissen, was mit VBA möglich ist

Das meiste aus VBA herausholen

VBA-Konzepte verstehen

Sicherstellen der Excel-Kompatibilität

Kapitel 2: Einfache Makros erstellen

Die Registerkarte »Entwicklertools« anzeigen

Ein Makro erstellen

Die Umgebung vorbereiten

Ein Makro aufzeichnen

Das Makro ausführen

Ein Makro im Visual Basic-Editor anzeigen

Das Makro abändern

Arbeitsmappen speichern, die Makros enthalten

Makro-Sicherheit verstehen

Teil II: Die Zusammenarbeit von VBA und Excel

Kapitel 3: Der Visual Basic-Editor

[Den Visual Basic-Editor kennenlernen](#)

[Arbeiten im Projekt-Explorer](#)

[Arbeiten im Codebereich](#)

[Anpassung des VBE](#)

Kapitel 4: Einführung in das Excel-Objektmodell

[Arbeiten mit dem Excel-Objektmodell](#)

[In Objekteigenschaften und Methoden eintauchen](#)

[VBA-Ressourcen zu weiterführenden Informationen](#)

Kapitel 5: Sub- und Funktionsprozeduren in VBA

[Sub-Prozeduren im Vergleich zu Funktionsprozeduren](#)

[Namen für Sub-Prozeduren und Funktionsprozeduren](#)

[Sub-Prozeduren ausführen](#)

[Funktionsprozeduren ausführen](#)

Kapitel 6: Mit dem Makro-Rekorder arbeiten

[Grundlagen der Aufzeichnung](#)

[Vorbereitung auf die Aufzeichnung](#)

[Zwischen relativem und absolutem Modus wählen](#)

[Dem Makro-Rekorder bei der Arbeit zusehen](#)

[Aufzeichnungsoptionen für Ihr Makro angeben](#)

[Code aufräumen, den der Makro-Rekorder aufgezeichnet hat](#)

Teil III: Programmierkonzepte

Kapitel 7: Wichtige Elemente der Sprache VBA

[Kommentare in Ihrem VBA-Code](#)

[Variablen, Konstanten und Datentypen](#)

[Zuweisungsbefehle verwenden](#)

[Arrays](#)

[Die Verwendung von Zeilenmarken \(Labels\)](#)

Kapitel 8: Mit Range-Objekten arbeiten

[Auf »Range«-Objekte verweisen](#)

[Auf einen Bereich mithilfe von Eigenschaften verweisen](#)

[Mit Eigenschaften des »Range«-Objekts arbeiten](#)

[Aktionen mit Methoden des »Range«-Objekts unternehmen](#)

Kapitel 9: VBA- und Arbeitsblattfunktionen

[Funktionen verstehen](#)

[Eingebaute VBA-Funktionen verwenden](#)

[Arbeitsblattfunktionen in VBA verwenden](#)

[Weitere Informationen über die Verwendung von Arbeitsblattfunktionen](#)

[Benutzerdefinierte Funktionen verwenden](#)

Kapitel 10: Steuerung des Programmablaufs und Entscheidungen

[Schwimmen mit dem Strom](#)

[Die »GoTo«-Anweisung](#)

[Entscheidungen, Entscheidungen!](#)

[Ihr Code für eine Schleife](#)

[»For Each-Next«-Schleifen für Collections ausführen](#)

Kapitel 11: Automatische Prozeduren und Ereignisse

[Vorbereitung auf das große Ereignis](#)

[Wissen, wo man den Ereigniscode einfügt](#)

[Eine Prozedur zur Ereignisverarbeitung schreiben](#)

[»Workbook«-Ereignisse auslösen](#)

[Aktivierungsereignisse verwenden](#)

[Programmierung von Arbeitsblattereignissen](#)

[Ereignisse verstehen, die keinen Objekten zugeordnet sind](#)

Kapitel 12: Techniken zur Fehlerverarbeitung

[Fehlertypen](#)

[Ein fehlerhaftes Beispielmakro](#)

[Alternative Wege der Fehlerverarbeitung](#)

[Fehlerverarbeitung: die Details](#)

[Ein beabsichtigter Fehler](#)

Kapitel 13: Techniken, mit denen Sie Fehler loswerden

[Verschiedene Fehlerarten](#)

[Programmierfehler identifizieren](#)

[Debugging-Techniken](#)

[Debugging-Tools verwenden](#)

[Tipps, wie Sie weniger Programmierfehler machen](#)

Kapitel 14: VBA-Programmierbeispiele

[Mit Bereichen arbeiten](#)

[Excel-Einstellungen abändern](#)

[Diagramme](#)

[VBA-Geschwindigkeitstipps](#)

Kapitel 15: Einfache Dialogfelder

[In VBA mit dem Benutzer interagieren](#)

[Meldungen mit der »MsgBox«-Funktion anzeigen](#)

[Daten mit der Funktion »InputBox« abfragen](#)

[Den Benutzer eine Datei oder einen Ordner auswählen lassen](#)

[Eingebaute Excel-Dialogfelder anzeigen](#)

Teil IV: Kommunikation mit Ihren Benutzern

Kapitel 16: UserForm-Grundlagen

[Erkennen, wann man ein UserForm braucht](#)

[UserForms erstellen: ein Überblick](#)

[Mit UserForms arbeiten](#)

[Ein UserForm-Beispiel](#)

Kapitel 17: UserForm-Steuer-elemente verwenden

[Steuer-elemente im Dialogfeld - die ersten Schritte](#)

[Details zu den Steuerelementen in einem Dialogfeld kennenlernen](#)

[Die Arbeit mit Steuerelementen in Dialogfeldern](#)

[Dialogfeld-Ästhetik](#)

Kapitel 18: UserForms - Techniken und Tricks

[Dialogfelder verwenden](#)

[Ein UserForm-Beispiel](#)

[Beispiel für ein Listenfeld-Steuerelement](#)

[Einen Bereich auswählen](#)

[Mehrere Gruppen von Optionsfeldern verwenden](#)

[Ein Drehfeld und ein Textfeld verwenden](#)

[Ein UserForm als Fortschrittsanzeige verwenden](#)

[Ein nicht modales Dialogfeld mit Registerkarten erstellen](#)

[Ein Diagramm in einem UserForm anzeigen](#)

[Eine Checkliste für Dialogfelder](#)

Kapitel 19: Zugriff auf Ihre Makros über die Benutzeroberfläche

[Das Menüband anpassen](#)

[Die Excel-Benutzeroberfläche mit VBA anpassen](#)

Teil V: Das große Ganze

Kapitel 20: Arbeitsblattfunktionen erstellen

[Vereinfachen Sie mit benutzerdefinierten Funktionen Ihre Arbeit](#)

[Grundlagen der VBA-Funktionen](#)

[Funktionen schreiben](#)

[Mit Funktionsargumenten arbeiten](#)

[Wrapper-Funktionen](#)

[Funktionen, die ein Array zurückgeben](#)

[Das Dialogfeld »Funktion einfügen« verwenden](#)

Kapitel 21: Add-Ins in Excel erstellen

[Was ist ein Add-In?](#)

[Gründe für die Erstellung von Add-Ins](#)

[Die Arbeit mit Add-Ins](#)

[Add-In-Grundlagen](#)

[Sehen Sie sich ein Beispiel für ein Add-In an](#)

Teil VI: Der Top-Ten-Teil

Kapitel 22: Zehn nützliche Tipps für den Visual Basic-Editor

[Kommentarblöcke verwenden](#)

[Mehrere Codezeilen kopieren](#)

[Zwischen Modulen und Prozeduren springen](#)

[Beamern Sie sich zu Ihren Funktionen](#)

[In der richtigen Prozedur bleiben](#)

[Den Code schrittweise ausführen](#)

[Zu einer bestimmten Codezeile springen](#)

[Die Codeausführung an einer bestimmten Stelle unterbrechen](#)

[Den Anfang und das Ende des Variablenwerts anzeigen](#)

[Die automatische Syntaxüberprüfung ausschalten](#)

Kapitel 23: Ressourcen mit Hilfe zu VBA

[Lassen Sie Excel die Makros für Sie schreiben](#)

[Nutzen Sie das Hilfesystem](#)

[Stibitzen Sie Code im Internet](#)

[Userforen optimal nutzen](#)

[Besuchen Sie Experten-Blogs](#)

[Suchen Sie auf YouTube nach Schulungsvideos](#)

[Nehmen Sie an Onlinekursen teil](#)

[Vom Office Dev Center lernen](#)

[Analysieren Sie andere Excel-Dateien in Ihrem Unternehmen](#)

[Fragen Sie die Excel-Experten vor Ort](#)

Kapitel 24: Zehn Dinge, die Sie beherzigen oder vermeiden sollten

[Deklarieren Sie alle Variablen](#)

[Verwechseln Sie Passwörter nicht mit Sicherheit](#)

[Bereinigen Sie Ihren Code](#)

[Schreiben Sie nicht alles in eine Prozedur](#)

[Denken Sie daran, dass es noch andere Software gibt](#)

[Gehen Sie nicht davon aus, dass alle Benutzer Makros aktivieren](#)

[Gewöhnen Sie sich an, zu experimentieren](#)

[Gehen Sie nicht davon aus, dass Ihr Code in anderen Excel-Versionen funktioniert](#)

[Denken Sie immer an Ihre Benutzer](#)

[Vergessen Sie die Datensicherung nicht!](#)

[**Abbildungsverzeichnis**](#)

[**Stichwortverzeichnis**](#)

[**End User License Agreement**](#)

Tabellenverzeichnis

Kapitel 7

[Tabelle 7.1: In VBA eingebaute Datentypen](#)

[Tabelle 7.2: Gültigkeitsbereich von Variablen](#)

[Tabelle 7.3: Operatoren in VBA](#)

[Tabelle 7.4: Logische Operatoren von VBA](#)

Kapitel 9

[Tabelle 9.1: VBA-Funktionen mit praktischen Nebenwirkungen](#)

[Tabelle 9.2: Die praktischsten eingebauten Funktionen von VBA](#)

Kapitel 10

[Tabelle 10.1: Programmierkonstrukte für Entscheidungen](#)

Kapitel 11

[Tabelle 11.1: Arbeitsmappenereignisse](#)

[Tabelle 11.2: Arbeitsblattereignisse](#)

Kapitel 12

[Tabelle 12.1: Verwendung der On-Error-Anweisung](#)

[Tabelle 12.2: Verwendung der Resume-Anweisung](#)

Kapitel 15

[Tabelle 15.1: Argumente der MsgBox-Funktion](#)

[Tabelle 15.2: Konstanten, die in der MsgBox-Funktion verwendet werden](#)

[Tabelle 15.3: Konstanten, die als Rückgabewerte für die MsgBox-Funktion verwendet...](#)

[Tabelle 15.4: Argumente der InputBox-Funktion](#)

[Tabelle 15.5: Argumente der GetOpenFilename-Methode](#)

[Tabelle 15.6: Argumente der GetSaveAsFilename-Methode](#)

Kapitel 16

[Tabelle 16.1: Steuerelemente in der Werkzeugsammlung](#)

Kapitel 17

[Tabelle 17.1: Allgemeine Eigenschaften von Steuerelementen](#)

Kapitel 18

[Tabelle 18.1: Einstellungen für die MultiSelect-Eigenschaft](#)

Kapitel 20

[Tabelle 20.1: Provisionsraten nach Verkäufen](#)

Illustrationsverzeichnis

Kapitel 2

[Abbildung 2.1: Die Registerkarte ENTWICKLERTOOLS ist normalerweise ausgeblendet, ...](#)

[Abbildung 2.2: Wenn Sie ein Makro aufzeichnen, wird das Dialogfeld MAKRO AUFZEICH...](#)

[Abbildung 2.3: Das ausgefüllte Dialogfeld MAKRO AUFZEICHNEN](#)

[Abbildung 2.4: Der VBE zeigt den VBA-Code in Modul1 von Mappe1 an.](#)

[Abbildung 2.5: Wenn Ihre Arbeitsmappe Makros enthält und Sie sie in einem Format ...](#)

[Abbildung 2.6: Der Abschnitt MAKROEINSTELLUNGEN im Dialogfeld TRUST CENTER](#)

[Abbildung 2.7: Warnung von Excel, dass die zu öffnende Datei Makros enthält](#)

[Abbildung 2.8: Die Warnung von Excel, dass die gerade geöffnete Arbeitsmappe Makr...](#)

Kapitel 3

[Abbildung 3.1: Der VBE ist ein flexibler Begleiter.](#)

[Abbildung 3.2: Der Projekt-Explorer zeigt mehrere Projekte an. Eines davon ist er...](#)

[Abbildung 3.3: Zu viele Codebereichen machen das VBE-Fenster unübersichtlich.](#)

[Abbildung 3.4: Die Prozedur `GuessName` zeigt dieses Dialogfeld an.](#)

[Abbildung 3.5: Die Registerkarte EDITOR im Dialogfeld OPTIONEN](#)

[Abbildung 3.6: Ein Beispiel für die Option ELEMENTE AUTOMATISCH AUFLISTEN](#)

[Abbildung 3.7: Die Option AUTOMATISCHE QUICKINFO zeigt Hilfe zur `MsgBox`-Funktion ...](#)

[Abbildung 3.8: Auf der Registerkarte EDITORFORMAT können Sie das Aussehen des VBE...](#)

[Abbildung 3.9: Die Registerkarte ALLGEMEIN im Dialogfeld OPTIONEN](#)

[Abbildung 3.10: Die Registerkarte VERANKERN im Dialogfeld OPTIONEN](#)

Kapitel 4

[Abbildung 4.1: Dieses Meldungsfeld zeigt die `Value`-Eigenschaft eines `Range`-Objekt...](#)

[Abbildung 4.2: Der VBE zeigt während der Eingabe eine Liste der Argumente an.](#)

[Abbildung 4.3: Beispiel für das Hilfesystem von VBA](#)

[Abbildung 4.4: Suche nach Objekten im Objektkatalog](#)

[Abbildung 4.5: Das Feature ELEMENTE AUTOMATISCH AUFLISTEN hilft Ihnen, Eigenschaf...](#)

Kapitel 5

[Abbildung 5.1: Verwendung der in VBA enthaltenen Funktion `InputBox`, um eine Zahl ...](#)

[Abbildung 5.2: Anzeige der Kubikwurzel einer Zahl über die Funktion `MsgBox`](#)

[Abbildung 5.3: Das Dialogfeld MAKRO listet alle verfügbaren Sub-Prozeduren auf.](#)

[Abbildung 5.4: Im Dialogfeld MAKROOPTIONEN können Sie die Optionen für Ihre Makro...](#)

[Abbildung 5.5: Das Menüband mit den Steuerelementen, die zur Verfügung stehen, we...](#)

[Abbildung 5.6: Wenn Sie einer Tabelle eine Schaltfläche hinzufügen, zeigt Excel a...](#)

[Abbildung 5.7: Das Ausführen einer Funktion im Direktfenster zeigt sofort das Erg...](#)

[Abbildung 5.8: Die Funktion CubeRoot wird im Dialogfeld FUNKTION EINFÜGEN in...](#)

[Abbildung 5.9: Die Verwendung der Funktion CubeRoot in Formeln](#)

Kapitel 6

[Abbildung 6.1: Eine praktische Anordnung der Fenster, um den Makro-Rekorder bei s...](#)

[Abbildung 6.2: Das Dialogfeld MAKRO AUFZEICHNEN bietet mehrere Optionen.](#)

Kapitel 7

[Abbildung 7.1: Wenn Sie !\[\]\(4fe57c3593bf1b21d272ae7ac8dfaf77_img.jpg\) + !\[\]\(67b4b7a7e28d2fb85c0437cda45ea068_img.jpg\) drücken, wird eine Liste mit Variablennamen, reserv...](#)

[Abbildung 7.2: Jedes VBA-Modul hat einen Deklarationen-Abschnitt, der vor allen S...](#)

Kapitel 8

[Abbildung 8.1: Auswahl eines nicht zusammenhängenden Bereichs](#)

[Abbildung 8.2: Das Meldungsfeld zeigt die Address-Eigenschaft eines 5 × 5 Zellen ...](#)

Kapitel 9

[Abbildung 9.1: Berechnung der Länge Ihres Namens](#)

[Abbildung 9.2: Eine Möglichkeit, eine Liste mit VBA-Funktionen anzuzeigen](#)

[Abbildung 9.3: Eine Arbeitsblattfunktion in Ihrem VBA-Code verwenden](#)

[Abbildung 9.4: Der Bereich Preisliste enthält Preise für Teile.](#)

[Abbildung 9.5: So erhalten Sie eine Liste der Arbeitsblattfunktionen, die Sie in ...](#)

Kapitel 10

[Abbildung 10.1: Eine von der Prozedur CheckCell ausgegebene Meldung](#)

[Abbildung 10.2: In einer Schleife wird eine Hintergrundfarbe für Zeilen festgeleg...](#)

[Abbildung 10.3: Diese Zellen wurden durch eine verschachtelte For-Next-Schleife g...](#)

[Abbildung 10.4: Mithilfe einer Schleife ein Schachbrettmuster erstellen](#)

Kapitel 11

[Abbildung 11.1: Der Projekt-Explorer zeigt Elemente für ein Projekt an.](#)

[Abbildung 11.2: Auswahl eines Ereignisses im Objektmodul DieseArbeitsmappe](#)

[Abbildung 11.3: Die Prozedur zur Ereignisverarbeitung wird ausgeführt, wenn die A...](#)

[Abbildung 11.4: Mit der ereignisverarbeitenden Prozedur Workbook_Open verfolgen, ...](#)

[Abbildung 11.5: Wenn ein Diagrammblatt aktiviert wird, sieht der Benutzer diese M...](#)

[Abbildung 11.6: Datenüberprüfung mit einer Ereignisprozedur](#)

Kapitel 12

[Abbildung 12.1: Die InputBox-Funktion zeigt ein Dialogfeld an, in dem der Benutze...](#)

[Abbildung 12.2: Excel zeigt diese Fehlermeldung an, wenn die Prozedur versucht, d...](#)

[Abbildung 12.3: Die Ausführung der Prozedur erzeugt einen Fehler, wenn ein Diagra...](#)

[Abbildung 12.4: Ein Laufzeitfehler in der Prozedur erzeugt diese hilfreiche Fehle...](#)

[Abbildung 12.5: Wenn ein Fehler auftritt, kann der Benutzer entscheiden, ob er es...](#)

Kapitel 13

[Abbildung 13.1: Eine Fehlermeldung wie diese bedeutet häufig, dass Ihr VBA-Code e...](#)

[Abbildung 13.2: Der Wert von drei Variablen in einem Meldungsfeld](#)

[Abbildung 13.3: Mit `Strg` + `Pause`? halten Sie die Ausführung Ihres Codes an und erhalten v...](#)

[Abbildung 13.4: Die `Debug.Print`-Anweisungen erzeugen Ausgaben im Direktbereich.](#)

[Abbildung 13.5: Die markierte Anweisung kennzeichnet einen Haltepunkt in dieser P...](#)

[Abbildung 13.6: Eine typische Szene im Unterbrechungsmodus](#)

[Abbildung 13.7: Im Dialogfeld ÜBERWACHUNG HINZUFÜGEN können Sie eine Be...](#)

[Abbildung 13.8: Das Überwachungsfenster zeigt alle Überwachungsausdrücke an.](#)

[Abbildung 13.9: Das Lokal-Fenster zeigt alle lokalen Variablen und ihren Inhalt a...](#)

Kapitel 14

[Abbildung 14.1: Dieser Bereich kann beliebig viele Zeilen umfassen.](#)

[Abbildung 14.2: Mit der VBA-Funktion `InputBox` wird ein Wert vom Benutzer abgefrag...](#)

[Abbildung 14.3: Excel mag es nicht, wenn Sie versuchen, eine Mehrfachauswahl zu k...](#)

[Abbildung 14.4: Sie können Excel anweisen, solche Warnungen nicht anzuzeigen, wäh...](#)

Kapitel 15

[Abbildung 15.1: Ein einfaches Meldungsfeld](#)

[Abbildung 15.2: Ein einfaches Meldungsfeld mit zwei Schaltflächen](#)

[Abbildung 15.3: Das `buttons`-Argument der `MsgBox`-Funktion bestimmt, was in dem Mel...](#)

[Abbildung 15.4: Dieses von der `MsgBox`-Funktion erstellte Dialogfeld zeigt einen T...](#)

[Abbildung 15.5: Die `InputBox`-Funktion zeigt dieses Dialogfeld an.](#)

[Abbildung 15.6: Ein weiteres Beispiel für die Verwendung der `InputBox`-Funktion](#)

[Abbildung 15.7: Mit der `InputBox`-Methode des `Application`-Objekts wird ein Bereich...](#)

[Abbildung 15.8: Die Methode `GetOpenFilename` zeigt ein benutzerdefinierbares Dialo...](#)

[Abbildung 15.9: Eines der in Excel eingebauten Dialogfelder mithilfe von VBA anze...](#)

[Abbildung 15.10: Verwenden Sie die Registerkarte MENÜBAND ANPASSEN, um einen...](#)

Kapitel 16

[Abbildung 16.1: Durch Anzeige eines UserForms können Sie Informationen vom Benutz...](#)

[Abbildung 16.2: Ein neues UserForm-Objekt](#)

[Abbildung 16.3: Im Eigenschaftfenster ändern Sie die Eigenschaften von UserForm...](#)

[Abbildung 16.4: Das UserForm mit zwei Schaltflächen](#)

[Abbildung 16.5: Dies ist das UserForm, nachdem drei Optionsfelder in den Rahmen e...](#)

[Abbildung 16.6: Weisen Sie eine Tastenkombination für die Ausführung des ChangeCa...](#)

[Abbildung 16.7: Das Makro ChangeCase wird der Symbolleiste für den Schnellzugriff...](#)

[Abbildung 16.8: Das UserForm wird ausgeführt.](#)

[Abbildung 16.9: Der Text wurde in Großbuchstaben umgewandelt.](#)

Kapitel 17

[Abbildung 17.1: Ein UserForm im VBE mit ein paar Steuerelementen](#)

[Abbildung 17.2: Im Eigenschaftfenster ändern Sie die Eigenschaften eines Steuer...](#)

[Abbildung 17.3: Einige Eigenschaften werden durch Auswahl aus einer Drop-down-Lis...](#)

[Abbildung 17.4: Dialogfeld mit CheckBox-Steuerelementen](#)

[Abbildung 17.5: Dialogfeld mit Kombinationsfeldern](#)

[Abbildung 17.6: Befehlsschaltflächen](#)

[Abbildung 17.7: Ein Bildfeld mit einem Foto](#)

[Abbildung 17.8: Bezeichnungsfelder können sehr unterschiedlich aussehen.](#)

[Abbildung 17.9: Listenfelder](#)

[Abbildung 17.10: Verwendung eines MultiPage-Steuerelements, um ein Dialogfeld mit...](#)

[Abbildung 17.11: Zwei Sätze Optionsfelder, jeweils in einem Rahmen gruppiert](#)

[Abbildung 17.12: Zwei RefEdit-Steuer-elemente](#)

[Abbildung 17.13: Eine Bildlaufleiste mit darunterliegendem Bezeichnungsfeld](#)

[Abbildung 17.14: Drehfelder](#)

[Abbildung 17.15: Textfelder](#)

[Abbildung 17.16: Umschaltfelder](#)

[Abbildung 17.17: Mit dem Befehl FORMAT | AUSRICHTEN ändern Sie die Ausrichtung vo...](#)

[Abbildung 17.18: Das Dialogfeld AKTIVIERREIHENFOLGE](#)

[Abbildung 17.19: Um direkten Zugriff auf Steuer-elemente zu bieten, die keine Tast...](#)

Kapitel 18

[Abbildung 18.1: Dieses Dialogfeld fordert den Benutzer auf, einen Namen einzutrag...](#)

[Abbildung 18.2: Durch Ausführung der Prozedur LogDinnerGuestWird das Dialogfeld a...](#)

[Abbildung 18.3: Verwendung des benutzerdefinierten Dialogfelds für die Dateneinga...](#)

[Abbildung 18.4: Bestimmen, welches Element in einem Listenfeld ausgewählt ist](#)

[Abbildung 18.5: Bestimmung der ausgewählten Elemente in einem Listenfeld, das ein...](#)

[Abbildung 18.6: Dieses Dialogfeld ermöglicht dem Benutzer, einen Bereich auszuwäh...](#)

[Abbildung 18.7: Dieses Dialogfeld enthält drei Optionsfeldgruppen.](#)

[Abbildung 18.8: Ein UserForm mit Drehfeld und einem begleitenden Textfeld](#)

[Abbildung 18.9: Dieses UserForm dient als Fortschrittsanzeige für ein langsames M...](#)

[Abbildung 18.10: Das UserForm mit der Fortschrittsanzeige](#)

[Abbildung 18.11: Die drei Registerkarten eines MultiPage-Steuer-elementes](#)

[Abbildung 18.12: Anzeige eines Diagramms in einem UserForm](#)

Kapitel 19

[Abbildung 19.1: Die Registerkarte MENÜBAND ANPASSEN im Dialogfeld EXCEL-OPTI...](#)

[Abbildung 19.2: Die Registerkarte ANSICHT, nachdem ich die neue Gruppe TEXT IN SP...](#)

[Abbildung 19.3: RibbonX-Code im Custom UI Editor](#)

[Abbildung 19.4: Die VBA-Callback-Prozedur, die durch Anklicken der Schaltfläche i...](#)

[Abbildung 19.5: Der Beweis, dass es mit XML möglich ist, einen neuen Menübandbefeh...](#)

[Abbildung 19.6: Ein Steuerelement auf der Registerkarte ADD-INS is...](#)

[Abbildung 19.7: Das Kontextmenü cell mit einem benutzerdefinierten Menüeintrag: G...](#)

Kapitel 20

[Abbildung 20.1: Verwendung der Funktion Commission in einem Arbeitsblatt](#)

[Abbildung 20.2: Verwendung der Funktion Commission2, die zwei Argumente entgegenn...](#)

[Abbildung 20.3: Mit der benutzerdefinierten Funktion SummOdd werden nur die unger...](#)

[Abbildung 20.4: Verwendung der ExtractElement-Funktion, um ein Element aus einer ...](#)

[Abbildung 20.5: Verwendung der Funktion MonthNames, um ein zwölfelementiges Array...](#)

[Abbildung 20.6: Verwendung einer benutzerdefinierten Funktion, um einen sortierte...](#)

[Abbildung 20.7: Standardmäßig enthält das Dialogfeld FUNKTION EINFÜGEN keine...](#)

[Abbildung 20.8: Jetzt zeigt das Dialogfeld für die benutzerdefinierte Funktion ei...](#)

[Abbildung 20.9: Standardmäßig zeigt das Dialogfeld FUNKTIONSARGUMENTE nur Beschre...](#)

Kapitel 21

[Abbildung 21.1: Das Dialogfeld ADD-INS listet alle Excel bekannten Add-Ins auf.](#)

[Abbildung 21.2: Das UserForm für das Add-In ChangeCase](#)

[Abbildung 21.3: Im Bereich mit den Dokumenteigenschaften können Si...](#)

[Abbildung 21.4: Das Dialogfeld ADD-INS, in dem das neue Add-In ausgewählt ist](#)

[Abbildung 21.5: Ein Add-In zu einem Nicht-Add-In machen](#)

Kapitel 22

[Abbildung 22.1: Ein Hochkomma am Anfang einer Zeile wandelt diese Zeile in einem ...](#)

[Abbildung 22.2: Verwenden Sie die Symbolleiste BEARBEITEN, wenn Sie einen Block a...](#)

[Abbildung 22.3: Wenn Sie beim Ziehen die !\[\]\(c3d993ca47bfe2a953c700506ce31fa0_img.jpg\)-Taste gedrückt halten, erstellen Sie e...](#)

[Abbildung 22.4: Drücken Sie !\[\]\(d66ff64371a51729ac8c1cdaa685ba6f_img.jpg\) + !\[\]\(0f31ebba7abcd47777e178db26f29705_img.jpg\), wenn der Cursor auf dem Namen einer Variablen ...](#)

[Abbildung 22.5: Beschränken Sie den Bildlauf auf die aktive Prozedur.](#)

[Abbildung 22.6: Drücken Sie !\[\]\(003082e50e3009141f59bd5df831749f_img.jpg\), um Ihr Makro schrittweise ausführen zu lassen.](#)

[Abbildung 22.7: Ziehen Sie den Pfeil, während Ihr Code schrittweise ausgeführt wi...](#)

[Abbildung 22.8: Ein Haltepunkt wird durch einen roten Punkt und dunkelrot unterle...](#)

[Abbildung 22.9: Anzeige der letzten Zeichen einer String-Variablen](#)

[Abbildung 22.10: Eine unvollendete Codezeile führt zu diesem irritierenden Warnhi...](#)

[Abbildung 22.11: Unterdrücken Sie während der Programmierung die Warnhinweise.](#)

Einführung

Herzlich willkommen, angehender Excel-Programmierer
...

Sie haben zweifellos gute Gründe, ein Buch über VBA-Programmierung zu lesen. Vielleicht haben Sie einen neuen Job bekommen (herzlichen Glückwunsch). Vielleicht versuchen Sie, einige der sich wiederholenden Datenverarbeitungsaufgaben, die Sie erledigen müssen, zu automatisieren. Vielleicht sind Sie tief im Herzen ein Computerfreak. Was auch immer der Grund ist, danke, dass Sie sich für dieses Buch entschieden haben.

In diesem Buch finden Sie alle Informationen, um schnell mit VBA loslegen zu können. Und selbst wenn Sie noch keinerlei Programmiererfahrung besitzen, wird Ihnen dieses Buch helfen. Anders als in den meisten anderen Programmierbüchern finden Sie hier genau die Fakten, die Sie benötigen, um schnell Ihre Fähigkeiten bei der VBA-Programmierung mit Excel auszubauen.

Über dieses Buch

In jeder größeren Buchhandlung (sei es in Ihrem Städtchen oder online) finden Sie eine Vielzahl an Excel-Büchern. Anhand der folgenden Liste können Sie entscheiden, ob dieses Buch wirklich für Sie geeignet ist. Dieses Buch

- ✓ ist für durchschnittliche bis fortgeschrittene Excel-Benutzer gedacht, die durch die Programmierung mit Visual Basic for Applications (VBA) ihre Arbeit beschleunigen wollen.
- ✓ bedingt keine Vorkenntnisse der Programmierung.

- ✓ deckt die am häufigsten verwendeten Befehle ab.
- ✓ ist für die aktuellen Excel-Versionen geeignet.
- ✓ könnte Ihnen bisweilen ein Lächeln ablocken – es gibt hier sogar Cartoons!

Wenn Sie mit einer älteren Excel-Version arbeiten, *könnte* das Buch infrage kommen, aber einige Dinge haben sich geändert. Wahrscheinlich kommen Sie mit einer Voraufgabe besser zurecht.

Und ja, dies ist *kein* Excel-Buch für Einsteiger. Wenn Sie nach einem allgemeinen Excel-Buch suchen, sehen Sie sich das folgende Buch an, das ebenfalls bei Wiley-VCH erschienen ist:

- ✓ *Excel 2021 für Dummies* von Greg Harvey

Dieses Buch gibt es auch zu früheren Excel-Versionen.

Beachten Sie, dass dieses Buch nicht *Das vollständige Handbuch für die Excel-VBA-Programmierung für Dummies* heißt. Dieses Buch deckt nicht alle Aspekte der Excel-Programmierung ab – aber man kann davon ausgehen, dass Sie wahrscheinlich nicht *alle* Informationen zu diesem Thema brauchen. Wenn Sie dieses Buch gelesen haben und glauben, Sie brauchen ein umfassenderes Buch zur Excel-Programmierung, empfehle ich Ihnen *Excel-VBA Power-Programmierung für Dummies*, ebenfalls bei Wiley-VCH erschienen.

Um den Inhalt besser zugänglich zu machen, haben wir dieses Buch in sechs Teile unterteilt:

- ✓ [Teil I](#): Excel-VBA – Die ersten Schritte
- ✓ [Teil II](#): Die Zusammenarbeit von VBA und Excel
- ✓ [Teil III](#): Programmierkonzepte

- ✓ [Teil IV](#): Kommunikation mit Ihren Benutzern
- ✓ [Teil V](#): Das große Ganze
- ✓ [Teil VI](#): Der Top-Ten-Teil

Konventionen in diesem Buch

Manchmal spreche ich über Tastenkombinationen – das bedeutet, Sie halten die eine Taste gedrückt, während Sie die andere drücken. `Strg` + `Z` bedeutet, dass Sie die `Strg`-Taste gedrückt halten, während Sie `Z` drücken.

Für Menübefehle benutze ich ein spezielles Zeichen, um Elemente des Menübands voneinander zu trennen. Beispielsweise verwenden Sie den folgenden Befehl, um innerhalb einer Arbeitsmappe einen benannten Bereich zu erstellen:

FORMELN | DEFINIERTE NAMEN | NAMEN
DEFINIEREN

FORMELN ist die Registerkarte oben im Menüband. DEFINIERTE NAMEN ist die Gruppe auf der Registerkarte FORMELN und NAMEN DEFINIEREN ist der eigentliche Befehl.

Der Visual Basic-Editor verwendet immer noch die alten Menüs und Menüleisten. Falls Sie also die Anweisung lesen, dass Sie EXTRAS | OPTIONEN wählen sollen, bedeutet dies, öffnen Sie das Menü EXTRAS und wählen Sie dort den Menübefehl OPTIONEN aus.

Bei der Programmierung mit Excel entwickeln Sie *Code* – das sind die Anweisungen, die Excel ausführt. Code in diesem Buch wird in nichtproportionaler Schrift dargestellt, etwa wie folgt:

```
Range("A1:A12").Select
```

Einige lange Codezeilen passen nicht in eine Zeile auf einer Buchseite. In diesem Fall verwende ich das

Standardtrennzeichen von VBA, ein Leerzeichen gefolgt von einem Unterstrich. Ein Beispiel:

```
Selection.PasteSpecial Paste:=xlValues, _  
    Operation:=xlNone, SkipBlanks:=False, _  
    Transpose:=False
```

Diesen Code können Sie genau so eingeben, aber auch innerhalb einer Zeile (dann lassen Sie das Leerzeichen und den Unterstrich weg).

Makrosicherheit

Es gibt nicht nur gute Menschen. Man hat den Eindruck, es gibt immer irgendeinen findigen Betrüger, der versucht, Sie auszunutzen oder irgendwelche Probleme zu verursachen. In der Programmierwelt ist das nicht anders. Vielleicht haben Sie schon von Computerviren gehört, die recht unangenehme Dinge mit Ihrem System veranstalten. Aber wussten Sie auch, dass sich Computerviren auch in einer Excel-Datei befinden können? Ganz recht. Es ist sogar ziemlich einfach, mit VBA einen Computervirus zu schreiben. Ein unbedarfter Benutzer kann eine Excel-Datei öffnen und den Virus auf andere Excel-Arbeitsmappen übertragen und sogar andere Systeme damit infizieren.

Im Laufe der Jahre hat sich Microsoft immer mehr Gedanken über Sicherheitsaspekte gemacht. Das ist nicht schlecht, bedeutet aber auch, dass die Excel-Benutzer wissen müssen, was passiert. Über den Befehl DATEI | OPTIONEN | TRUST CENTER erreichen Sie die Sicherheitseinstellungen von Excel. Hier gibt es unzählige Optionen. Man erzählt sich, dass es schon Benutzer gegeben haben soll, die dieses Dialogfeld geöffnet haben und nie wieder gesehen wurden.

Klicken Sie auf die Registerkarte
MAKROEINSTELLUNGEN (links im Dialogfeld TRUST

CENTER). Die folgenden Optionen werden angezeigt:

- ✓ **ALLE MAKROS OHNE BENACHRICHTIGUNG DEAKTIVIEREN:** Es werden keine Makros ausgeführt, unabhängig davon, was Sie machen.
- ✓ **ALLE MAKROS MIT BENACHRICHTIGUNG DEAKTIVIEREN:** Wenn Sie eine Arbeitsmappe mit Makros öffnen, wird entweder die Nachrichtenleiste mit der Option geöffnet, die Sie anklicken können, um Makros zu aktivieren, oder Sie erhalten (wenn das Fenster mit dem Visual Basic-Editor geöffnet ist) eine Nachricht, in der Sie gefragt werden, ob Sie Makros aktivieren wollen.
- ✓ **ALLE MAKROS AUSSER DIGITAL SIGNIERTEN MAKROS DEAKTIVIEREN:** Nur Makros mit einer digitalen Signatur dürfen ausgeführt werden (aber für Signaturen, die Sie als nicht vertrauenswürdig markiert haben, erhalten Sie weiterhin Sicherheitswarnungen).
- ✓ **ALLE MAKROS AKTIVIEREN:** Alle Makros ohne Warnungen ausführen. Diese Option wird nicht empfohlen, weil möglicherweise gefährlicher Code ausgeführt werden kann.

Überlegen Sie sich das folgende Szenario: Sie schreiben eine Woche lang an einem genialen VBA-Programm, das Ihr Unternehmen völlig revolutionieren wird. Sie testen es sorgfältig und senden es an Ihren Chef. Er ruft Sie zu sich ins Büro und teilt Ihnen mit, dass Ihr Makro überhaupt nichts bewerkstelligt. Was ist passiert? Wahrscheinlich gestatten die Sicherheitseinstellungen Ihres Chefs nicht, das Makro auszuführen. Oder er verwendet die Standardeinstellung von Microsoft und deaktiviert beim Öffnen einer Datei alle Makros.