

Richard Mansfield

MASTERING

VBA

for Microsoft® Office 2010

Master VBA Fundamentals
and Essentials

Create Custom Applications
and Macros for Office 2010



SERIOUS SKILLS.

Mastering

VBA for Microsoft® Office

2010

Richard Mansfield



WILEY

Wiley Publishing, Inc.

Acquisitions Editor: Agatha Kim
Development Editor: Denise Lincoln
Technical Editor: Russ Mullen
Production Editor: Rachel Gigliotti
Copy Editor: Judy Flynn
Editorial Manager: Pete Gaughan
Production Manager: Tim Tate
Vice President and Executive Group Publisher: Richard Swadley
Vice President and Publisher: Neil Edde
Book Designers: Maureen Forys and Judy Fung
Proofreader: Publication Services, Inc.
Indexer: Ted Laux
Project Coordinator, Cover: Lynsey Stanford
Cover Designer: Ryan Sneed
Cover Image: © Pete Gardner/DigitalVision/Getty Images

Copyright © 2010 by Wiley Publishing, Inc., Indianapolis, Indiana
Published simultaneously in Canada

ISBN: 978-0-470-63400-4

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Limit of Liability/Disclaimer of Warranty: The publisher and the author make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation warranties of fitness for a particular purpose. No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional services. If professional assistance is required, the services of a competent professional person should be sought. Neither the publisher nor the author shall be liable for damages arising herefrom. The fact that an organization or Web site is referred to in this work as a citation and/or a potential source of further information does not mean that the author or the publisher endorses the information the organization or Web site may provide or recommendations it may make. Further, readers should be aware that Internet Web sites listed in this work may have changed or disappeared between when this work was written and when it is read.

For general information on our other products and services or to obtain technical support, please contact our Customer Care Department within the U.S. at (877) 762-2974, outside the U.S. at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Library of Congress Cataloging-in-Publication Data

Mansfield, Richard, 1945-
Mastering VBA for Office 2010 / Richard Mansfield. — 1st ed.
p. cm.
ISBN 978-0-470-63400-4 (pbk.)
ISBN 978-0-470-92263-7 (ebk.)
ISBN 978-0-470-92265-1 (ebk.)
ISBN 978-0-470-92264-4 (ebk.)

1. Microsoft Visual Basic for applications. 2. Microsoft Office. I. Title.
QA76.73.M53M36 2010
005.2'768--dc22

2010021585

TRADEMARKS: Wiley, the Wiley logo, and the Sybex logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. Microsoft is a registered trademark of Microsoft Corporation. All other trademarks are the property of their respective owners. Wiley Publishing, Inc. is not associated with any product or vendor mentioned in this book.

10987654321

Dear Reader,

Thank you for choosing *Mastering VBA for Microsoft Office 2010*. This book is part of a family of premium-quality Sybex books, all of which are written by outstanding authors who combine practical experience with a gift for teaching.

Sybex was founded in 1976. More than 30 years later, we're still committed to producing consistently exceptional books. With each of our titles, we're working hard to set a new standard for the industry. From the paper we print on, to the authors we work with, our goal is to bring you the best books available.

I hope you see all that reflected in these pages. I'd be very interested to hear your comments and get your feedback on how we're doing. Feel free to let me know what you think about this or any other Sybex book by sending me an email at nedde@wiley.com. If you think you've found a technical error in this book, please visit <http://sybex.custhelp.com>. Customer feedback is critical to our efforts at Sybex.

Best regards,

A handwritten signature in black ink, appearing to read 'Neil Edde', written in a cursive style.

Neil Edde
Vice President and Publisher
Sybex, an Imprint of Wiley

I dedicate this book to my good friend Cliff Way.

Acknowledgments

I'd like to thank all the good people at Sybex who contributed to this book, in particular Agatha Kim, whose encouragement made this book possible in the first place and whose thoughtful guidance was helpful during the writing process. I am also indebted to Denise Lincoln, the development editor, whose many valuable suggestions contributed to this book's readability and cohesion. My gratitude also goes to Judy Flynn, the copy editor, who via a very close, line-by-line read, improved this book in many ways; she is truly an exceptional copy editor. The technical editor, Russ Mullen, checked the book for accuracy and ensured that all the code examples work without any errors. Finally, thanks to capable Rachel Gigliotti, production editor, the book went smoothly through its final stages — author review and production.

About the Author

Mastering VBA for Microsoft Office 2010 is Richard Mansfield's 45th book. His recent titles include *Visual Basic .NET Power Tools* (Sybex, 2003), *Office Application Development All-in-One Desk Reference for Dummies* (Wiley, 2004), and *Programming: A Beginner's Guide* (McGraw-Hill, 2009). Overall, his books have sold more than 500,000 copies worldwide, and have been translated into 12 languages.

Contents at a Glance

<i>Introduction</i>	<i>xxvii</i>
Part 1 • Recording Macros and Getting Started with VBA	1
Chapter 1 • Recording and Running Macros in the Office Applications	3
Chapter 2 • Getting Started with the Visual Basic Editor.....	31
Chapter 3 • Editing Recorded Macros	65
Chapter 4 • Creating Code from Scratch in the Visual Basic Editor	87
Part 2 • Learning How to Work with VBA	105
Chapter 5 • Understanding the Essentials of VBA Syntax	107
Chapter 6 • Working with Variables, Constants, and Enumerations	123
Chapter 7 • Using Array Variables	145
Chapter 8 • Finding the Objects, Methods, and Properties You Need	167
Part 3 • Making Decisions and Using Loops and Functions	191
Chapter 9 • Using Built-in Functions	193
Chapter 10 • Creating Your Own Functions	227
Chapter 11 • Making Decisions in Your Code	245
Chapter 12 • Using Loops to Repeat Actions	265
Part 4 • Using Message Boxes, Input Boxes, and Dialog Boxes	293
Chapter 13 • Getting User Input with Message Boxes and Input Boxes.....	295
Chapter 14 • Creating Simple Custom Dialog Boxes	315
Chapter 15 • Creating Complex Dialog Boxes	381

Part 5 • Creating Effective Code.	429
Chapter 16 • Building Modular Code and Using Classes.	431
Chapter 17 • Debugging Your Code and Handling Errors.	457
Chapter 18 • Building Well-Behaved Code	487
Chapter 19 • Securing Your Code with VBA's Security Features	501
Part 6 • Programming the Office Applications	525
Chapter 20 • Understanding the Word Object Model and Key Objects	527
Chapter 21 • Working with Widely Used Objects in Word	559
Chapter 22 • Understanding the Excel Object Model and Key Objects	591
Chapter 23 • Working with Widely Used Objects in Excel.	617
Chapter 24 • Understanding the PowerPoint Object Model and Key Objects.	631
Chapter 25 • Working with Shapes and Running Slide Shows	653
Chapter 26 • Understanding the Outlook Object Model and Key Objects.	673
Chapter 27 • Working with Events in Outlook.	693
Chapter 28 • Understanding the Access Object Model and Key Objects.	713
Chapter 29 • Manipulating the Data in an Access Database via VBA	735
Chapter 30 • Accessing One Application from Another Application.	755
Chapter 31 • Programming the Office 2010 Ribbon.	783
Appendix • The Bottom Line.	811
<i>Index.</i>	<i>855</i>

Contents

<i>Introduction</i>	<i>xxvii</i>
---------------------------	--------------

Part 1 • Recording Macros and Getting Started with VBA **1**

Chapter 1 • Recording and Running Macros in the Office Applications. **3**

What Is VBA and What Can You Do with It?	3
The Difference between Visual Basic and Visual Basic for Applications	4
Understanding Macro Basics	5
Recording a Macro	6
Planning the Macro	6
Starting the Macro Recorder	7
Naming the Macro	10
Assigning a Way to Run a Macro	15
Running a Macro	20
Recording a Sample Word Macro	21
Recording a Sample Excel Macro	24
Create a Personal Macro Workbook if You Don't Have One Yet	24
Record a Sample Excel Macro	24
Assigning a Way of Running the Macro	25
Assigning a Macro to a Quick Access Toolbar Button in Word	26
Assigning a Macro to a Shortcut Key Combination	26
Deleting a Macro	26
The Bottom Line	28

Chapter 2 • Getting Started with the Visual Basic Editor **29**

Opening the Visual Basic Editor	29
Opening the Visual Basic Editor with a Macro Selected	30
Opening the Visual Basic Editor Directly	30
Navigating to a Macro	31
Using the Visual Basic Editor's Main Windows	32
The Project Explorer	32
The Object Browser	35
The Code Window	36
The Properties Window	40
The Immediate Window	42
Setting Properties for a Project	43
Customizing the Visual Basic Editor	45
Choosing Editor and View Preferences	46
Choosing and Laying Out the Editor Windows	53

Customizing the Toolbar and Menu Bar	54
Customizing the Toolbox	57
Closing the Visual Basic Editor and Returning to the Host Application	61
The Bottom Line	61

Chapter 3 • Editing Recorded Macros 63

Testing a Macro in the Visual Basic Editor	63
Stepping through a Macro	65
Setting Breakpoints	66
Commenting Out Lines	67
Stepping Out of a Macro	68
Editing the Word Macro	68
Stepping Through the <i>Transpose_Word_Right</i> Macro	70
Running the <i>Transpose_Word_Right</i> Macro	70
Creating a <i>Transpose_Word_Left</i> Macro	70
Save Your Work	72
Editing the Excel Macro	72
Unhiding the Personal Macro Workbook	72
Opening the Macro for Editing	72
Editing the Macro	74
Saving Your Work	76
Editing a PowerPoint Macro	77
Save Your Work	81
The Bottom Line	83

Chapter 4 • Creating Code from Scratch in the Visual Basic Editor 85

Setting Up the Visual Basic Editor for Creating the Procedures	85
Creating a Procedure for Word	86
Creating a Procedure for Excel	92
Creating a Procedure for PowerPoint	96
Creating a Procedure for Access	101
The Bottom Line	102

Part 2 • Learning How to Work with VBA 105

Chapter 5 • Understanding the Essentials of VBA Syntax 107

Getting Ready	107
Procedures	108
Functions	108
Subprocedures	109
Statements	109
Keywords	113
Expressions	113
Operators	113
Variables	114

Constants	114
Arguments	115
Specifying Argument Names vs. Omitting Argument Names	116
When to Include the Parentheses around the Argument List	117
Objects	117
Collections	117
Properties	118
Methods	118
Events	118
The Bottom Line	120
Chapter 6 • Working with Variables, Constants, and Enumerations	123
Working with Variables	123
Choosing Names for Variables	124
Declaring a Variable	125
Choosing the Scope and Lifetime of a Variable	128
Specifying the Data Type for a Variable	134
Working with Constants	140
Declaring Your Own Constants	140
Choosing the Scope and Lifetime for Your Constants	141
Working with Enumerations	142
The Bottom Line	142
Chapter 7 • Using Array Variables	145
What Is an Array?	145
Declaring an Array	146
Storing Values in an Array	148
Multidimensional Arrays	149
Declaring a Dynamic Array	150
Redimensioning an Array	150
Returning Information from an Array	151
Erasing an Array	151
Finding Out Whether a Variable Is an Array	151
Finding the Bounds of an Array	152
Sorting an Array	152
Searching through an Array	156
Performing a Linear Search through an Array	156
Performing a Binary Search Through an Array	160
The Bottom Line	165
Chapter 8 • Finding the Objects, Methods, and Properties You Need	167
What Is an Object?	167
Properties	169
Methods	170
Working with Collections	172
Working with an Object in a Collection	173
Adding an Object to a Collection	173

Finding the Objects You Need	174
Using the Macro Recorder to Record the Objects You Need	174
Using the Object Browser	176
Using Help to Find the Object You Need	182
Using the Auto List Members Feature.	184
Using Object Variables to Represent Objects.	186
The Bottom Line.	189

Part 3 • Making Decisions and Using Loops and Functions 191

Chapter 9 • Using Built-in Functions 193

What Is a Function?	193
Using Functions	194
Passing Arguments to a Function	196
Using Functions to Convert Data from One Type to Another	198
Using the <i>Asc</i> Function to Return a Character Code	199
Using the <i>Val</i> Function to Extract a Number from the Start of a String	199
Using the <i>Str</i> Function to Convert a Value to a String	200
Using the <i>Format</i> Function to Format an Expression	201
Using the <i>Chr</i> Function and Constants to Enter Special Characters in a String	205
Using Functions to Manipulate Strings	207
Using the <i>Left</i> , <i>Right</i> , and <i>Mid</i> Functions to Return Part of a String.	208
Using <i>InStr</i> and <i>InStrRev</i> to Find a String within Another String	212
Using <i>LTrim</i> , <i>RTrim</i> , and <i>Trim</i> to Trim Spaces from a String	214
Using <i>Len</i> to Check the Length of a String.	215
Using <i>StrConv</i> , <i>LCase</i> , and <i>UCase</i> to Change the Case of a String.	216
Using the <i>StrComp</i> Function to Compare Apples to Apples	217
Using VBA's Mathematical Functions.	218
Using VBA's Date and Time Functions.	219
Using the <i>DatePart</i> Function to Parse Dates	221
Using the <i>DateDiff</i> Function to Return an Interval	222
Using the <i>DateAdd</i> Function to Add to a Date	223
Using File-Management Functions.	223
Using the <i>Dir</i> Function to Check Whether a File Exists.	223
Returning the Current Path	225
The Bottom Line.	225

Chapter 10 • Creating Your Own Functions 227

Components of a Function	228
Creating a Function	229
Starting a Function Manually.	229
Starting a Function by Using the Add Procedure Dialog Box.	230
Passing Arguments to the Function	231
Declaring the Data Types of Arguments	232
Specifying an Optional Argument	232
Controlling the Scope of a Function	232

Examples of Functions for Any VBA-Enabled Application.....	233
Creating a Function for Word	237
Creating a Function for Excel.....	239
Creating a Function for PowerPoint	240
Creating a Function for Access	242
The Bottom Line.....	244

Chapter 11 • Making Decisions in Your Code.....245

How Do You Compare Things in VBA?	245
Testing Multiple Conditions by Using Logical Operators.....	247
<i>If</i> Statements.....	249
<i>If... Then</i>	249
<i>If... Then... Else</i> Statements	252
<i>If... Then... Elseif... Else</i> Statements	253
Creating Loops with <i>If</i> and <i>GoTo</i>	258
Nesting <i>If</i> Statements	259
<i>Select Case</i> Statements	261
Syntax	262
Example	262
The Bottom Line.....	264

Chapter 12 • Using Loops to Repeat Actions.....265

When Should You Use a Loop?	265
Understanding the Basics of Loops.....	266
Using <i>For... Next</i> Loops for Fixed Repetitions	267
<i>For... Next</i> Loops	267
<i>For Each... Next</i> Loops	275
Using an <i>Exit For</i> Statement.....	275
Using <i>Do... Loop</i> Loops for Variable Numbers of Repetitions.....	276
<i>Do While... Loop</i> Loops.....	277
<i>Do... Loop While</i> Loops.....	280
<i>Do Until... Loop</i> Loops	282
<i>Do... Loop Until</i> Loops	284
Using an <i>Exit Do</i> Statement.....	286
<i>While... Wend</i> Loops.....	287
Nesting Loops.....	288
Avoiding Infinite Loops	290
The Bottom Line.....	291

Part 4 • Using Message Boxes, Input Boxes, and Dialog Boxes..... 293

Chapter 13 • Getting User Input with Message Boxes and Input Boxes

Opening a Procedure to Work On.....	296
Displaying Status Bar Messages in Word and Excel	297
Message Boxes	299
The Pros and Cons of Message Boxes	299

Message Box Syntax	299
Displaying a Simple Message Box	300
Displaying a Multiline Message Box	301
Choosing Buttons for a Message Box	303
Choosing an Icon for a Message Box	303
Setting a Default Button for a Message Box	304
Controlling the Modality of a Message Box	306
Specifying a Title for a Message Box	307
Title Bars Can Provide Useful Information	307
Adding a Help Button to a Message Box	308
Specifying a Help File for a Message Box	309
Using Some Arguments without Others	309
Retrieving a Value from a Message Box	310
Input Boxes	311
Input Box Syntax	311
Retrieving Input from an Input Box	313
Forms: When Message Boxes and Input Boxes Won't Suffice	313
The Bottom Line	314
Chapter 14 • Creating Simple Custom Dialog Boxes	315
When Should You Use a Custom Dialog Box?	315
Creating a Custom Dialog Box	316
Designing the Dialog Box	317
Inserting a User Form	317
Renaming a User Form	321
Adding Controls to the User Form	322
Renaming Controls	326
Moving a Control	327
Changing the Caption on a Control	328
Key Properties for the Toolbox Controls	330
Working with Groups of Controls	348
Aligning Controls	349
Placing Controls	350
Adjusting the Tab Order of the Dialog Box	351
Linking a Dialog Box to a Procedure	352
Loading and Unloading a Dialog Box	353
Displaying and Hiding a Dialog Box	354
Setting a Default Command Button	354
Retrieving the User's Choices from a Dialog Box	354
Returning a String from a Text Box	355
Returning a Value from an Option Button	355
Returning a Value from a Check Box	356
Returning a Value from a List Box	356
Returning a Value from a Combo Box	358
Examples of Connecting Dialog Boxes to Procedures	358
Word Example: The Move-Paragraph Procedure	358
General Example: Opening a File from a List Box	368

Creating the Code for the User Form	371
Using an Application's Built-in Dialog Boxes from VBA	374
Displaying a Built-in Dialog Box	375
Setting and Restoring Options in a Built-in Dialog Box	378
Which Button Did the User Choose in a Dialog Box?	378
Specifying a Time-Out for a Dialog Box	379
The Bottom Line	380

Chapter 15 • Creating Complex Dialog Boxes381

Creating and Working with Complex Dialog Boxes	382
Updating a Dialog Box to Reflect the User's Choices	382
Revealing a Hidden Part of a Dialog Box	382
Tracking a Procedure in a Dialog Box	386
Using Multipage Dialog Boxes and Tab Strip Controls	388
Creating a Modeless Dialog Box	398
Choosing a Position for the Dialog Box	399
Using Events to Control Forms	400
Events That Apply Only to the <i>UserForm</i> Object	403
Events That Apply to the <i>UserForm</i> Object and to Container Controls	408
Events That Apply to Many or Most Controls	412
Events That Apply Only to a Few Controls	425
The Bottom Line	426

Part 5 • Creating Effective Code 429

Chapter 16: Building Modular Code and Using Classes431

Creating Modular Code	431
What Is Modular Code?	431
Advantages of Using Modular Code	432
How to Approach Creating Modular Code	432
Arranging Your Code in Modules	433
Calling a Procedure	433
Making Logical Improvements to Your Code	435
Making Visual Improvements to Your Code	441
Creating and Using Classes	446
What Can You Do with Class Modules?	446
A Brief Overview	447
Planning Your Class	447
Creating the Class Module	448
Naming the Class	448
Setting the <i>Instancing</i> Property	448
Declaring Variables and Constants for the Class	448
Adding Properties to the Class	449
Adding Methods to the Class	453
Using Your Class	454
The Bottom Line	456

Chapter 17: Debugging Your Code and Handling Errors	457
Principles of Debugging	457
The Different Types of Errors	458
Language Errors	459
Compile Errors	459
Runtime Errors	461
Program Logic Errors	462
VBA's Debugging Tools	463
Break Mode	464
The Step Over and Step Out Commands	465
The Locals Window	466
The Watch Window	467
The Immediate Window	470
The Call Stack Dialog Box	473
Dealing with Infinite Loops	473
Dealing with Runtime Errors	474
When Should You Write an Error Handler?	474
Trapping an Error	474
Disabling an Error Trap	476
Resuming after an Error	477
Getting a Description of an Error	480
Raising Your Own Errors	480
Suppressing Alerts	481
Handling User Interrupts in Word, Excel, and Project	481
Disabling User Input While a Procedure Is Running	482
Disabling User Input While Part of a Procedure Is Running	482
Documenting Your Code	483
The Bottom Line	484
Chapter 18: Building Well-Behaved Code	487
What Is a Well-Behaved Procedure?	487
Retaining or Restoring the User Environment	488
Leaving the User in the Best Position to Continue Working	489
Keeping the User Informed during the Procedure	489
Manipulating the Cursor	491
Displaying Information at the Beginning of a Procedure	492
Communicating with the User via a Message Box or Dialog Box at the End of a Procedure	493
Creating a Log File	493
Making Sure a Procedure Is Running under Suitable Conditions	496
Cleaning Up after a Procedure	497
Undoing Changes the Procedure Has Made	497
Removing Scratch Files and Folders	498
The Bottom Line	499

Chapter 19: Securing Your Code with VBA's Security Features . . . 501

Understanding How VBA Implements Security	501
Signing Your Macro Projects with Digital Signatures	504
What Is a Digital Certificate?	505
Getting a Digital Certificate	505
Choosing a Suitable Level of Security	514
Understanding the Security Threats Posed by VBA	514
Protecting against Macro Viruses	515
Specifying a Suitable Security Setting	515
New Security in Office 2010	516
Specifying Whom to Trust	518
Locking Your Code	521
The Bottom Line	523

Part 6 • Programming the Office Applications 525**Chapter 20 • Understanding the Word Object Model and Key Objects 527**

Examining the Word Object Model	527
Working with the <i>Documents</i> Collection and the <i>Document</i> Object	530
Creating a Document	530
Creating a Template	531
Saving a Document	531
Opening a Document	537
Closing a Document	540
Changing a Document's Template	540
Printing a Document	540
Working with the <i>ActiveDocument</i> Object	542
Working with the <i>Selection</i> Object	543
Checking the Type of Selection	544
Checking the Story Type of the Selection	545
Getting Other Information about the Current Selection	546
Inserting Text at, after, or before the Selection	550
Inserting a Paragraph in a Selection	551
Applying a Style	551
Extending a Selection	551
Collapsing a Selection	552
Creating and Using Ranges	553
Defining a Named Range	553
Redefining a Range	554
Using the Duplicate Property to Store or Copy Formatting	554
Manipulating Options	555
Making Sure Hyperlinks Require Ctrl+Clicking	555
Turning Off Overtyping	555
Setting a Default File Path	555
Turning Off Track Changes	556
The Bottom Line	557

Chapter 21 • Working with Widely Used Objects in Word.	559
Using Find and Replace via VBA.	559
Understanding the Syntax for the Execute Method	561
Putting Find and Replace to Work.	564
Working with Headers, Footers, and Page Numbers	566
Understanding How VBA Implements Headers and Footers	566
Getting to a Header or Footer	566
Checking to See If a Header or Footer Exists	567
Linking to the Header or Footer in the Previous Section.	567
Creating a Different First-Page Header.	567
Creating Different Odd- and Even-Page Headers	567
Adding Page Numbers to Your Headers and Footers.	568
Working with Sections, Page Setup, Windows, and Views.	571
Adding a Section to a Document	571
Changing the Page Setup	572
Opening a New Window Containing an Open Document	573
Closing All Windows for a Document Except the First	573
Splitting a Window.	573
Displaying the Document Map for a Window.	574
Scrolling a Window	574
Arranging Windows	574
Positioning and Sizing a Window	575
Making Sure an Item Is Displayed in the Window	575
Changing a Document's View	576
Zooming the View to Display Multiple Pages	576
Working with Tables	576
Creating a Table.	577
Selecting a Table	577
Converting Text to a Table.	578
Making Sure the Selection Is within a Table	579
Finding Out Where a Selection Is within a Table	580
Sorting a Table.	581
Adding a Column to a Table	582
Deleting a Column from a Table	582
Setting the Width of a Column.	582
Selecting a Column.	583
Adding a Row to a Table	583
Deleting a Row from a Table.	584
Setting the Height of One or More Rows	584
Selecting a Row	584
Inserting a Cell	585
Returning the Text within a Cell	585
Entering Text in a Cell	585
Deleting Cells	585
Selecting a Range of Cells	586
Converting a Table or Rows to Text	587
The Bottom Line.	588

Chapter 22 • Understanding the Excel Object Model and Key Objects	591
Getting an Overview of the Excel Object Model.	591
Understanding Excel's Creatable Objects.	593
Managing Workbooks.	593
Creating a Workbook	593
Saving a Workbook.	595
Opening a Workbook.	598
Closing a Workbook.	600
Sharing a Workbook.	601
Protecting a Workbook	601
Working with the <i>ActiveWorkbook</i> Object.	602
Working with Worksheets	603
Inserting a Worksheet	603
Deleting a Worksheet.	604
Copying or Moving a Worksheet.	604
Printing a Worksheet	605
Protecting a Worksheet	606
Working with the <i>ActiveSheet</i> Object.	607
Working with the Active Cell or Selection.	608
Working with the Active Cell	608
Working with the User's Selection.	610
Working with Ranges	610
Working with a Range of Cells.	610
Creating a Named Range	610
Deleting a Named Range.	611
Working with a Named Range	612
Working with the Used Range	612
Working with the Special Cells	612
Entering a Formula in a Cell.	613
Setting Options.	613
Setting Options in the Application Object	614
Setting Options in a Workbook	614
The Bottom Line.	615
Chapter 23 • Working with Widely Used Objects in Excel	617
Working with Charts.	617
Creating a Chart	617
Specifying the Source Data for the Chart.	619
Specifying the Chart Type.	619
Working with Series in the Chart.	619
Adding a Legend to the Chart	622
Adding a Chart Title.	622
Working with a Chart Axis.	623
Working with Windows	623
Opening a New Window on a Workbook.	623
Closing a Window	624
Activating a Window	624

Arranging and Resizing Windows	624
Zooming a Window and Setting Display Options	625
Working with Find and Replace	626
Searching with the Find Method	626
Continuing a Search with the <i>FindNext</i> and <i>FindPrevious</i> Methods.....	627
Replacing with the <i>Replace</i> Method	628
Searching for and Replacing Formatting	628
Adding Shapes	629
The Bottom Line.....	629

Chapter 24 • Understanding the PowerPoint Object Model and Key Objects631

Getting an Overview of the PowerPoint Object Model	631
Understanding PowerPoint's Creatable Objects	632
Working with Presentations.....	633
Creating a New Presentation Based on the Default Template.....	633
Creating a New Presentation Based on a Template.....	634
Opening an Existing Presentation.....	634
Saving a Presentation.....	635
Closing a Presentation.....	637
Exporting a Presentation or Some Slides to Graphics.....	637
Printing a Presentation	638
Applying a Template to a Presentation, to a Slide, or to a Range of Slides	639
Working with the Active Presentation	639
Working with Windows and Views	640
Working with the Active Window.....	640
Opening a New Window on a Presentation.....	641
Closing a Window	641
Activating a Window	641
Arranging and Resizing Windows	642
Changing the View.....	642
Working with Panes.....	642
Working with Slides	643
Adding a Slide to a Presentation	643
Inserting Slides from an Existing Presentation.....	644
Finding a Slide by Its ID Number	645
Changing the Layout of an Existing Slide	645
Deleting an Existing Slide	645
Copying and Pasting a Slide.....	646
Duplicating a Slide	646
Moving a Slide.....	646
Accessing a Slide by Name	646
Working with a Range of Slides.....	646
Formatting a Slide.....	647
Setting a Transition for a Slide, a Range of Slides, or a Master	648
Working with Masters.....	649
Working with the Slide Master.....	649

Working with the Title Master	650
Working with the Handout Master	650
Working with the Notes Master	650
Deleting a Master	650
The Bottom Line	651

Chapter 25 • Working with Shapes and Running Slide Shows . . . 653

Working with Shapes	653
Adding Shapes to Slides	653
Deleting a Shape	658
Selecting All Shapes	658
Repositioning and Resizing a Shape	658
Copying Formatting from One Shape to Another	659
Working with Text in a Shape	659
Setting an Animation for a Shape or a Range of Shapes	663
Working with Headers and Footers	665
Returning the Header or Footer Object You Want	665
Displaying or Hiding a Header or Footer Object	665
Setting the Text in a Header or Footer	666
Setting the Format for Date and Time Headers and Footers	666
Setting Up and Running a Slide Show	667
Controlling the Show Type	667
Creating a Custom Show	668
Deleting a Custom Show	668
Starting a Slide Show	668
Changing the Size and Position of the Slide Show	669
Moving from Slide to Slide	669
Pausing the Show and Using White and Black Screens	670
Starting and Stopping Custom Shows	670
Exiting the Slide Show	671
The Bottom Line	671

Chapter 26 • Understanding the Outlook Object Model and Key Objects . . . 673

Getting an Overview of the Outlook Object Model	673
Understanding Where Outlook Stores VBA Items	674
Understanding Outlook's Creatable Objects and Main User Interface Items	674
Working with the <i>Application</i> Object	675
Working with the <i>NameSpace</i> Object	675
Working with Inspectors and Explorers	678
Creating Items	679
Quitting Outlook	681
Understanding General Methods for Working with Outlook Objects	681
Using the <i>Display</i> Method	681
Using the <i>Close</i> Method	681
Using the <i>Delete</i> Method	682

Using the <i>PrintOut</i> Method	683
Using the <i>Save</i> Method	683
Using the <i>SaveAs</i> Method	683
Working with Messages	684
Creating a New Message	684
Working with the Contents of a Message	685
Adding an Attachment to a Message	685
Sending a Message	686
Working with Calendar Items	686
Creating a New Calendar Item	686
Working with the Contents of a Calendar Item	687
Working with Tasks and Task Requests	688
Creating a Task	688
Working with the Contents of a Task Item	688
Assigning a Task to a Colleague	689
Searching for Items	689
The Bottom Line	692

Chapter 27 • Working with Events in Outlook693

Working with Application-Level Events	694
Using the <i>Startup</i> Event	695
Using the <i>Quit</i> Event	696
Using the <i>ItemSend</i> Event	696
Using the <i>NewMail</i> and <i>NewMailEx</i> Events	697
Using the <i>AdvancedSearchComplete</i> and the <i>AdvancedSearchStopped</i> Events	698
Using the <i>MAPILogonComplete</i> Event	699
Using the <i>Reminder</i> Event	699
Using the <i>OptionsPagesAdd</i> Event	699
Working with Item-Level Events	700
Declaring an Object Variable and Initializing an Event	700
Understanding the Events That Apply to All Message Items	701
Understanding the Events That Apply to Explorers, Inspectors, and Views	703
Understanding the Events That Apply to Folders	705
Understanding the Events That Apply to Items and Results	705
Understanding the Events That Apply to the Outlook Bar	707
Understanding the Events That Apply to Reminders	708
Understanding the Events That Apply to Synchronization	709
Understanding Quick Steps	710
The Bottom Line	711

**Chapter 28 • Understanding the Access Object Model
and Key Objects713**

Getting Started with VBA in Access	713
Creating a Module in the VBA Editor	715
Creating a Function	715
The Revamped Macro Builder	715
Creating an Access-Style Macro to Run a Function	716

Using an <i>AutoExec</i> Macro to Initialize an Access Session	717
Running a Subprocedure.	718
Understanding the <i>Option Compare Database</i> Statement.	718
Getting an Overview of the Access Object Model	719
Understanding Creatable Objects in Access	720
Opening and Closing Databases	721
Using the <i>CurrentDb</i> Method to Return the Current Database.	721
Opening a Different Database as the Current Database and Closing the Current Database.	721
Opening Multiple Databases at Once	724
Closing a Database	725
Creating and Removing Workspaces	725
Working with the <i>Screen</i> Object.	726
Using the <i>DoCmd</i> Object to Run Access Commands	727
Using the <i>OpenForm</i> Method to Open a Form	731
Using the <i>PrintOut</i> Method to Print an Object.	732
Using the <i>RunMacro</i> Method to Run a Macro	733
The Bottom Line.	733

Chapter 29 • Manipulating the Data in an Access

Database via VBA735

Understanding How to Proceed	735
Preparing to Manage the Data in a Database	736
Adding a Reference to the Appropriate Object Library	736
Establishing a Connection to the Database	736
Opening a Recordset	737
Opening a Recordset Using ADO	737
Choosing How to Access the Data in an ADO Recordset	739
Opening a Recordset Using DAO.	743
Accessing a Particular Record in a Recordset	747
Using the <i>MoveFirst</i> , <i>MoveNext</i> , <i>MovePrevious</i> , and <i>MoveLast</i> Methods	747
Using the <i>Move</i> Method to Move by Multiple Records.	748
Searching for a Record	749
Searching for a Record in an ADO Recordset	749
Searching for a Record in a DAO Recordset.	750
Returning the Fields in a Record.	751
Editing a Record.	752
Inserting and Deleting Records.	752
Closing a Recordset	753
The Bottom Line.	753

Chapter 30 • Accessing One Application from

Another Application.755

Understanding the Tools Used to Communicate between Applications	755
Using Automation to Transfer Information.	756
Understanding Early and Late Binding	757
Creating an Object with the <i>CreateObject</i> Function.	758
Returning an Object with the <i>GetObject</i> Function.	758
Examples of Using Automation with the Office Applications.	759

Using the <i>Shell</i> Function to Run an Application	768
Returning the Task ID of the Started Application	769
Activating an Application	769
Using Data Objects to Store and Retrieve Information	770
Creating a Data Object	770
Storing Information in a Data Object	771
Returning Information from a Data Object	771
Assigning Information to the Clipboard	772
Returning Information from the Clipboard to a Data Object	772
Finding Out Whether a Data Object Contains a Given Format	773
Communicating via DDE	773
Using <i>DDEInitiate</i> to Start a DDE Connection	773
Using <i>DDERequest</i> to Return Text from Another Application	774
Using <i>DDEPoke</i> to Send Text to Another Application	775
Using <i>DDEExecute</i> to Execute a Command in Another Application	775
Using <i>DDETerminate</i> to Close a DDE Channel	776
Using <i>DDETerminateAll</i> to Close All Open DDE Channels	776
Communicating via <i>SendKeys</i>	776
The Bottom Line	781
Chapter 31 • Programming the Office 2010 Ribbon	783
Hiding the Editing Group on the Word Ribbon	784
A Word of Warning	786
XML Terminology	786
Using Built-in Icons	788
Working with Excel and PowerPoint	788
Undoing Ribbon Modifications	789
Selecting the Scope of Your Ribbon Customization	789
Adding a New Group	789
Cautions about Customizing	790
Two Ways to Find the Correct <i>idMso</i>	791
Adding Callbacks	792
Adding Attributes	794
Using Built-in Icons and ScreenTips	794
Creating Your Own Icons	794
Using Menus and Lists	795
Adding Menus	795
Adding a DropDown List Control	798
Using a DialogBoxLauncher	799
Toggling with a Toggle Button Control	800
Modifying the Ribbon in Access	801
Testing Your New Ribbon	804
Adding a Callback in Access	805
What to Look For If Things Go Wrong	806
Employ Error Message Tools	806

Cure Common User Interface Programming Problems	807
Where to Go from Here.	809
The Bottom Line.	810

Appendix • The Bottom Line 811

Chapter 1: Recording and Running Macros in the Office Applications	811
Chapter 2: Getting Started with the Visual Basic Editor	812
Chapter 3: Editing Recorded Macros	813
Chapter 4: Creating Code from Scratch in the Visual Basic Editor	814
Chapter 5: Understanding the Essentials of VBA Syntax	817
Chapter 6: Working with Variables, Constants, and Enumerations	818
Chapter 7: Using Array Variables	820
Chapter 8: Finding the Objects, Methods, and Properties You Need	821
Chapter 9: Using Built-in Functions	823
Chapter 10: Creating Your Own Functions	824
Chapter 11: Making Decisions in Your Code.	826
Chapter 12: Using Loops to Repeat Actions	827
Chapter 13: Getting User Input with Message Boxes and Input Boxes.	828
Chapter 14: Creating Simple Custom Dialog Boxes	830
Chapter 15: Creating Complex Dialog Boxes.	833
Chapter 16: Building Modular Code and Using Classes	835
Chapter 17: Debugging Your Code and Handling Errors	836
Chapter 18: Building Well-Behaved Code	837
Chapter 19: Securing Your Code with VBA's Security Features.	839
Chapter 20: Understanding the Word Object Model and Key Objects	841
Chapter 21: Working with Widely Used Objects in Word	842
Chapter 22: Understanding the Excel Object Model and Key Objects	844
Chapter 23: Working with Widely Used Objects in Excel	844
Chapter 24: Understanding the PowerPoint Object Model and Key Objects	845
Chapter 25: Working with Shapes and Running Slide Shows	846
Chapter 26: Understanding the Outlook Object Model and Key Objects.	847
Chapter 27: Working with Events in Outlook	848
Chapter 28: Understanding the Access Object Model and Key Objects	849
Chapter 29: Manipulating the Data in an Access Database via VBA	850
Chapter 30: Accessing One Application from Another Application	851
Chapter 31: Programming the Office 2010 Ribbon	853
<i>Index</i>	855

Introduction

Visual Basic for Applications (VBA) is a powerful tool that enables you to automate operations in the Microsoft Office applications and in other applications that host VBA. By automating operations using VBA, you can save yourself and your colleagues huge amounts of time and effort. Getting more work done in less time is usually good for your disposition and self-esteem, and it can do wonderful things for your job security and career.

This book shows you how to program VBA, using the Microsoft Office 2010 applications for specific examples. You can apply the principles you learn in this book to any other VBA-enabled application as well.

Where to Get the Example Code

Throughout this book you'll find many code examples, mostly written in VBA, but in Chapter 31, written in XML. Rather than type in the code, you'll save yourself time (and debugging headaches) if you copy the code from this book's web page and paste it into the VBA Editor (or in Chapter 31, into the Office Custom UI Editor). You can find all the code from this book — accurate, fully tested, and bug free — at this book's website, www.sybex.com/go/masteringvba2010.

What Can I Do with VBA?

You can use VBA to automate almost any action that you can perform interactively (manually) with an Office 2010 application. For example, in Word, you can use VBA to create a document, add text to it, format it, and edit it. In Excel, you can integrate data from multiple workbooks into a single workbook. In PowerPoint, you can automatically create a custom presentation including the latest data drawn from a variety of sources. And in Access you can create new tables and populate them with data.

VBA performs actions faster, more accurately, more reliably, and far more cheaply than any human. As long as you can define firm conditions for making a decision, VBA can also make decisions for you. By adding decision-making structures and loops (repetitions) to your code, you can take it far beyond the range of actions that any human user can perform.

Beyond automating actions you would otherwise perform manually, VBA gives you the tools to create user interfaces for your code — message boxes, input boxes, and user forms, graphical objects that you can use to create forms and custom dialog boxes.

Using VBA, you can also create a custom application that runs within the host application. For example, you could build within PowerPoint a custom application that automatically creates a presentation for you.

In addition, you can use VBA to access one application from another application. For example, Word can't do much in the way of mathematical calculations on sets of data: that's Excel's specialty. So when working with VBA from Word, you can write code that starts an Excel session, performs some calculations, and then puts the results into a Word document. Similarly, you could send objects from Excel to PowerPoint using VBA. You get the picture.

Because VBA provides a standard set of tools that differ in capability according to the specializations of the host application, once you've learned to use VBA in one application, you'll be able to apply that knowledge quickly to using VBA in another application. For example, you might start by learning VBA in order to manipulate Excel and then move on to using your VBA skills with Outlook. You'll need to learn the components of the Outlook application because they're different from the Excel components, but you'll be able to quickly apply your VBA knowledge without a problem.

As with any programming language, getting started with VBA involves a learning curve — but you can use the Macro Recorder tool built into Word and Excel to learn VBA more quickly. This book uses the Macro Recorder as the jumping-off point for you to start creating code. The book takes you through recording macros and then teaches you to edit recorded code before delving into the essentials of VBA syntax. From there, you work your way into more complex topics.

And Word, because it's the most popular Office application and because it has the most sophisticated and efficient programming tools, is used for many of the examples in this book. But there are plenty of examples showing how to program Excel, PowerPoint, Outlook, and even Access. And code that works in one Office 2010 application will generally work with other applications in the suite — with little or sometimes no modification.

What's in This Book?

This book teaches you how to use VBA to automate your work in Office 2010 applications. For its general examples, the book focuses on Word, Excel, Outlook, and PowerPoint because those are the Microsoft Office applications that you're most likely to have and because they have less eccentric programming tools and strategies than Access. The last part of the book continues the discussion of how to program these four applications, but also increases coverage of Access.

Part 1 of the book, "Recording Macros and Getting Started with VBA," comprises the following chapters:

- ◆ Chapter 1 shows you how to record a macro using the Macro Recorder in Word and Excel. You also learn several ways to run macros and how to delete them.
- ◆ Chapter 2 introduces you to the VBA Editor, the application in which you create VBA code (either by editing recorded code or by writing code from scratch) and user forms. The second half of this chapter discusses how you can customize the VBA Editor so that you can work in it more efficiently.
- ◆ Chapter 3 shows you how to edit recorded macros, using the macros you recorded in Chapter 1. You learn how to step through and test a macro in the VBA Editor.