

Microsoft Office 2016

Excel 2016 Part Three



Instructor Guide

INSTRUCTOR GUIDE

Microsoft® Office  
Excel® 2016: Part 3

Microsoft® Office Excel® 2016: Part 3

|  |  |  |
| --- | --- | --- |
| Author | Media Designer | Content Editor |
| Daniel Nicholson | Jim Fitt | Ray Burns |

Courseware Release Version 4.0

© 2016 by Velsoft Training Materials, Inc.

Notice of Rights  
No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the prior written permission of Velsoft Training Materials, Inc., except under the terms of a courseware site license agreement.

Trademark Notice

Terms such as PowerPoint, Windows, Word, Microsoft, etc. are trademarks of Microsoft, Inc. Throughout this courseware title, trademark names are used. Rather than just put a trademark symbol in each occurrence of a trademarked name, we state we are using the names only in an editorial fashion and to the benefit of the trademark owner with no intention of infringement of the trademark.

Notice of Liability

The information in this courseware title is distributed on an ‘as is’ basis, without warranty. While every precaution has been taken in the preparation of this course, neither the authors nor Velsoft Training Materials, Inc. shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the instructions contained in this book or by the computer software and hardware products described in it.

Disclaimer

We make a sincere effort to ensure the accuracy of the material described herein; however, Velsoft Training Materials, Inc. makes no warranty, expressed or implied, with respect to the quality, correctness, reliability, accuracy, or freedom from error of this document or the products it describes. Data used in examples and sample data files are intended to be fictional. Any resemblance to real persons or companies is entirely coincidental.

All information in this manual was correct at the time of writing. We are not affiliated with nor have any control over changes made to the product described in this manual. These include, but are not limited to, changes in the application’s color scheme, icon appearance and locations, addition or removal of program features, online templates, and help content. We reserve the right to make corrections to the courseware at any time and without notification.

Terms and conditions

Sample versions: If the version of courseware that you are viewing is marked as NOT FOR TRAINING, SAMPLE, or similar, then it is made available for content and style review only and cannot be used in any part of a training course. Sample versions may be shared but cannot be re-sold to a third party. For licensed users: This document may only be used under the terms of the license agreement from Velsoft Training Materials, Inc. We reserve the right to alter the licensing conditions at any time, without prior notice.

Microsoft® Office Excel® 2016: Part 3

Contents

About This Course 9

Course Prerequisites 9

Course Overview 9

Course Objectives 10

How To Use This Book 10

Lesson 1: Automating Worksheet Functionality 11

TOPIC A: Update Workbook Properties 12

Workbook Properties 12

Activity 1-1 18

TOPIC B: Create and Edit a Macro 27

Macros 27

The Record Macro Dialog Box 27

Naming Macros 29

Visual Basic for Applications 29

Copying Macros Between Workbooks 31

Macro Security Settings 33

Activity 1-2 36

TOPIC C: Apply Conditional Formatting 43

Conditional Formatting 43

Conditional Formats 44

The Conditional Formatting Rules Manager Dialog Box 47

The New Formatting Rule Dialog Box 49

Clear Rules 51

Activity 1-3 52

TOPIC D: Add Data Validation Criteria 60

Data Validation 60

The Data Validation Dialog Box 60

Activity 1-4 63

Summary 73

Review Questions 73

Lesson 2: Auditing Worksheets 74

TOPIC A: Trace Cells 75

The Trace Cells Feature 75

Tracer Arrows 76

Activity 2-1 78

TOPIC B: Troubleshoot Invalid Data and Formula Errors 81

Invalid Data 81

The Error Checking Command 82

Error Types 84

Activity 2-2 85

TOPIC C: Watch and Evaluate Formulas 89

The Watch Window 89

Formula Evaluation 91

Activity 2-3 93

TOPIC D: Create a Data List Outline 99

Outlines 99

The Outline Group 100

Activity 2-4 102

Summary 106

Review Questions 106

Lesson 3: Analyzing and Presenting Data 107

TOPIC A: Create Sparklines 108

Sparklines 108

Types of Sparklines 109

The Sparkline Tools – Design Tab 109

Activity 3-1 111

TOPIC B: Create Scenarios 115

Scenarios 115

The What-If Analysis Tools 116

The Scenario Manager Dialog Box 116

Activity 3-2 118

TOPIC C: Perform a What-If Analysis 128

Add-In Types 128

Goal Seek Feature 132

The Solver Tool 135

Activity 3-3 137

TOPIC D: Perform a Statistical Analysis with the Analysis ToolPak 145

Analysis ToolPak 145

The Data Analysis Dialog Box 146

Activity 3-4 147

TOPIC E: Create Interactive Data with Power View 157

The Power View Add-In 157

Enabling Power View 158

Creating a Power View 161

Activity 3-5 163

Summary 168

Review Questions 168

Lesson 4: Working with Multiple Workbooks 169

TOPIC A: Consolidate Data 170

Data Consolidation 170

The Consolidate Dialog Box 171

Consolidation Functions 172

Activity 4-1 173

TOPIC B: Link Cells in Different Workbooks 179

External References 179

Activity 4-2 181

TOPIC C: Merge Workbooks 184

The Compare and Merge Workbooks Feature 184

Activity 4-3 186

Summary 191

Review Questions 191

Lesson 5: Exporting Excel Data 192

TOPIC A: Export Excel Data 193

The Export Process 193

Activity 5-1 195

TOPIC B: Import a Delimited Text File 200

The Import Process 200

The Get External Data Group 201

Delimited Text Files 201

Methods of Importing Text Files 201

Activity 5-2 202

TOPIC C: Integrate Excel Data with the Web 206

The File Publishing Process 206

Publish as Web Page Dialog Box 208

Activity 5-3 210

TOPIC D: Create a Web Query 214

Web Queries 214

The New Web Query Dialog Box 215

Activity 5-4 218

Summary 224

Review Questions 224

Lesson 6: Importing and Exporting XML Data 225

TOPIC A: Import and Export XML Data 226

XML 226

XML Components 227

XML Schemas 228

XML Maps 228

The XML Source Task Pane 229

Import and Export XML Data 233

Activity 6-1 239

Summary 243

Review Questions 243

Lesson Labs 244

Lesson 1 244

Lesson Lab 1-1 244

Lesson Lab 1-2 245

Lesson 2 246

Lesson Lab 2-1 246

Lesson Lab 2-2 247

Lesson 3 248

Lesson Lab 3-1 248

Lesson Lab 3-2 249

Lesson 4 250

Lesson Lab 4-1 250

Lesson 5 251

Lesson Lab 5-1 251

Lesson Lab 5-2 252

Lesson 6 253

Lesson Lab 6-1 253

Course Wrap-Up 254

Post-Course Assessment 254

Course Summary 257

Appendices 258

Keyboard Shortcut Quick Reference Sheet 258

Glossary 260

Index 262

About This Course

Course Prerequisites

This manual assumes the user has completed or has an understanding of the materials covered in the second part of the Microsoft Office Excel 2016 courseware, including:

* Creating advanced formulas
* Analyzing data with logical and lookup functions
* Organizing worksheet data with tables
* Visualizing data with charts
* Analyzing data with PivotTables, slicers, and PivotCharts
* Inserting graphics
* Enhancing workbooks

Course Overview

Welcome to the third part of our Microsoft Office Excel 2016 courseware. This version of Excel incorporates some new features and connectivity options in efforts to make collaboration and production as easy as possible.

This course is intended to help all users get up to speed on the different aspects of Excel, including some of its more advanced features. We will cover how to automate worksheet functionality, audit worksheets, analyze data, work with multiple workbooks, export Excel data, as well as import and export XML data.

Course Objectives

By the end of this course, users should be comfortable with automating worksheet functionality, auditing worksheets, using a variety of different analysis tools, working effectively with multiple workbooks, exporting Excel data, as well as importing and exporting XML data to and from a workbook.

How To Use This Book

This course is broken up into six lessons. Each lesson focuses on several key topics, each of which are broken down into easy-to-follow concepts. At the end of each topic, you will be given an activity to complete. At the end of each lesson, we will summarize what has been covered and provide a few review questions for you to answer. Supplemental learning for selected topics is provided in the form of Lesson Labs at the end of this book.

Before you begin, download the course’s Exercise Files to a convenient location. They will be referenced throughout this course and are a key part of your learning experience.

Lesson 1:  
Automating Worksheet Functionality

Lesson Objectives

In this lesson you will learn how to:

* Update workbook properties
* Create and edit a macro
* Apply conditional formatting
* Add data validation criteria to a workbook

TOPIC A: Update Workbook Properties

In Excel 2016, you can view, add, or even edit the properties of an existing workbook. These properties can include information such as who authored the workbook, date information, and more. This type information is referred to as **metadata**, and it can very useful when managing multiple workbooks. Over the course of this topic you will learn all about workbook properties and how to work with them.

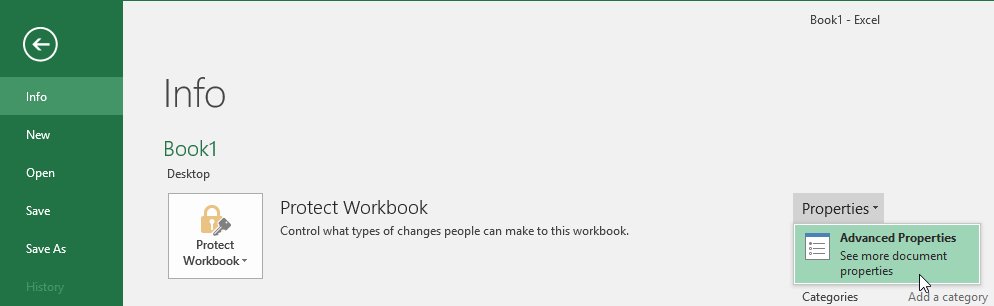
Topic Objectives

In this topic, you will learn:

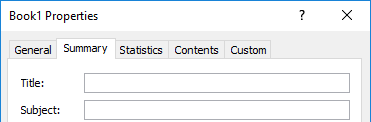
* About workbook properties

Workbook Properties

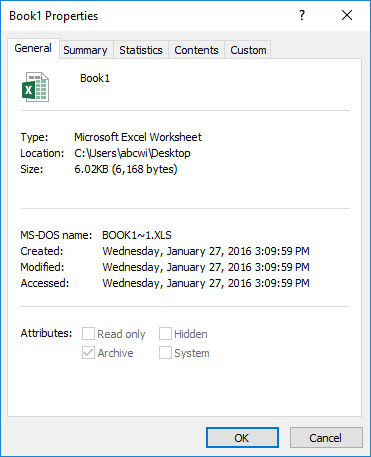
You can view your workbook’s properties from within Excel by clicking File → Info → Properties → Advanced Properties:



When the Workbook Properties dialog opens, you will see that it contains five tabs: General, Summary, Statistics, Contents, and Custom.

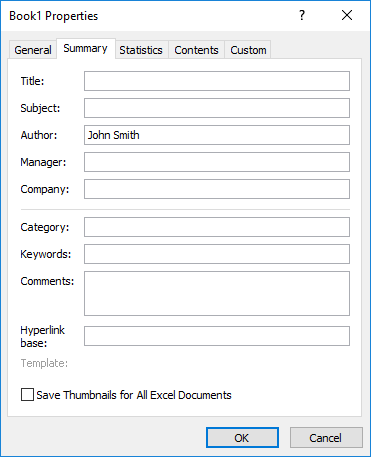


The **General tab** shows you basic information about the current workbook including the file name, file type, and file size. Additionally, this tab will show you the MS-DOS name for the current workbook, as well as when it was created, modified, and last accessed:

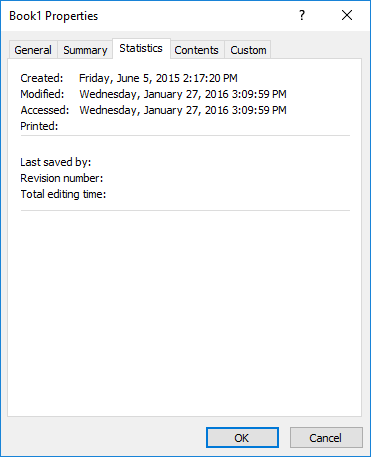


Instructor Tip: This information is provided by your computer and cannot be edited or removed.

The **Summary tab** is used to view, edit, or add authorship and keyword information about the current file. Specifically, you are able to add title, subject, author, manager, and company information to the file properties. You can also add category information, as well as keywords and comments to help identify and categorize this file:

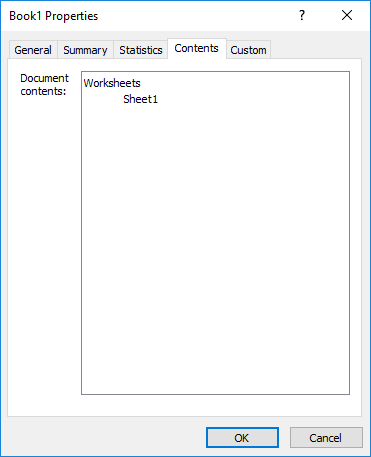


The **Statistics tab** will display when the current file was created, last modified, last accessed, and even when it was last printed. You will also be able to determine when the current file was last saved, how many times it has been revised, and how long it has been open for editing:



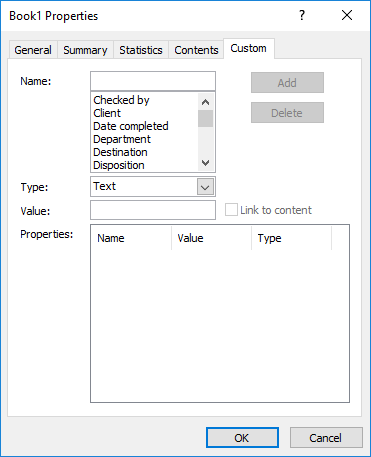
(Like the General tab, the data here is generated by your computer and cannot be edited or removed.)

The **Contents tab** will show you the number of worksheets contained within the current workbook and their names:



(Like most of the other tabs on this dialog box, the data in this tab cannot be modified.)

Finally, the **Custom tab** allows you to add custom information to the file. For example, you can add attributes like the purpose of the file or the name of the department in your organization that created it:



Keep in mind that any data that is added here will only be displayed on the Custom tab in the Properties dialog.

Activity 1-1

Updating Workbook Properties

You have created a workbook and need to send it to a client. The client requires that the properties for each workbook you send to them are fully entered and current.

|  |  |
| --- | --- |
|  | To begin, open Activity 1-1 using Microsoft Excel 2016: |
|  |  |
|  | Let’s open the Properties dialog box for the workbook. To start, click File → Info: |
|  |  |
|  | On the right-hand side of the Info category, click the Properties → Advanced Properties: |
|  |  |
|  | The Properties dialog box will now be displayed. If it is not already displayed, click the Summary tab: |
|  |  |
|  | Type “Sales Goals and Bonus Report” into the Title text box: |
|  |  |
|  | Inside the Author text box, type your own name: |
|  |  |
|  | Type “ABC Widgets” into the Company text box: |
|  |  |
|  | Inside the Category text box, type “Sales and Bonuses:” |
|  |  |
|  | Finally, in the Keywords text box type “Sales Goals, Bonus, Bonuses, Sales” with the commas included: |
|  |  |
|  | Click OK to apply the new properties: |
|  |  |
|  | Back in the Info category of Excel 2016, you will see this information populate the right side of the Info category: |
|  |  |
|  | Save your work as Activity 1-1 Complete and then close Microsoft Excel 2016. |

TOPIC B: Create and Edit a Macro

Due to the nature of Excel and how it is used, you may often find yourself repeating the same task over and over again. To help streamline your workflow, it is possible to automate those tasks by creating a macro. Macros are created using code in Visual Basic for Applications; however, you don’t need to know programming in order to create a macro. Over the course of this topic, you will learn how to create a macro using the Macro Recorder.

Topic Objectives

In this topic, you will learn:

* About macros
* About the Record Macro dialog box
* About macro naming
* About Visual Basic for Applications (VBA)
* How to copy macros between workbooks
* How to configure macro security settings

Macros

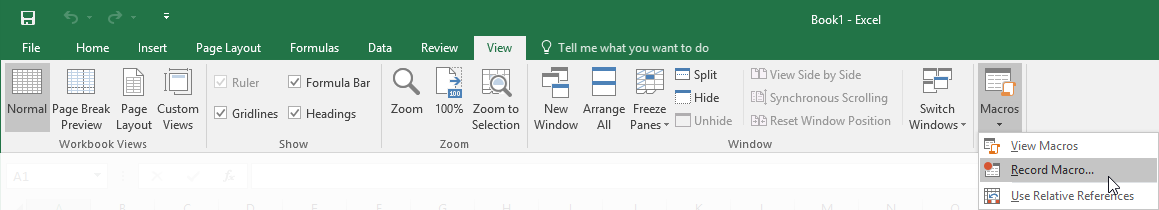
**Macros** are small programs that are created to complete a specific task or set of tasks. They are created by either recording yourself completing this task or by coding a macro that completes this task directly in VBA. In either case, once the macro is complete, you can bind it to a custom button or shortcut key to execute it from the window for the application that you are creating the macro for.

For example, a macro could be created that will protect a worksheet using predetermined settings. This would save you time from digging through the Excel interface and selecting these settings manually each time the task was required.

The Record Macro Dialog Box

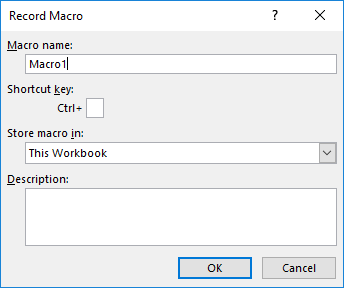
The Macro Recorder is used to create a macro by recording all of the actions that you make within Excel. For example, if you wanted to create a macro that will enable specific worksheet protection settings, you would start the recorder, complete all of the actions to apply those settings, and then stop the recorder. Once the process is complete, the Macro Recorder will translate the actions you completed into VBA code and save it as a macro that you can use again and again.

To start recording a macro, you first need to open the Record Macro dialog box by clicking View → Macros → Record Macro:



Instructor Tip: You can also find this command on the status bar.

In the Record Macro dialog box, you can enter details about the macro you are recording, including the macro name, shortcut key to execute the macro, where the macro is to be stored, and a brief description of what the macro does:



Naming Macros

The name of a macro needs to adhere to a set of naming rules:

* The name must start with an uppercase or lowercase letter.
* The name cannot contain any spaces.
* While the name can contain any letters, numbers, and the underscore character, it cannot include any other symbols or other characters.
* Finally, a macro name cannot share the name of another object (such as a worksheet) or other built-in name (such as “Function”).

**Macro Security and the .xlsm File Format**

By their nature, macros are a security risk and in the wrong hands they can cause real damage to a file or computer system. To help address this problem, any workbooks that include macros must be saved in the Macro-Enabled Workbook format (.xlsm). This format provides increased security against malicious code by prompting you to manually enable or disable macros in a file. This system allows you to determine the trustworthiness of a macro before you run it. If you choose to save a workbook that includes a macro in another format, the macro will not be saved.

**Macro Storage**

When working with macros, keep in mind that they can be stored in three different locations: the current workbook, a new workbook, or your Personal Macro workbook.

By choosing to save a macro in the current workbook or a new workbook, the macro will only be available to those workbooks when they are open.

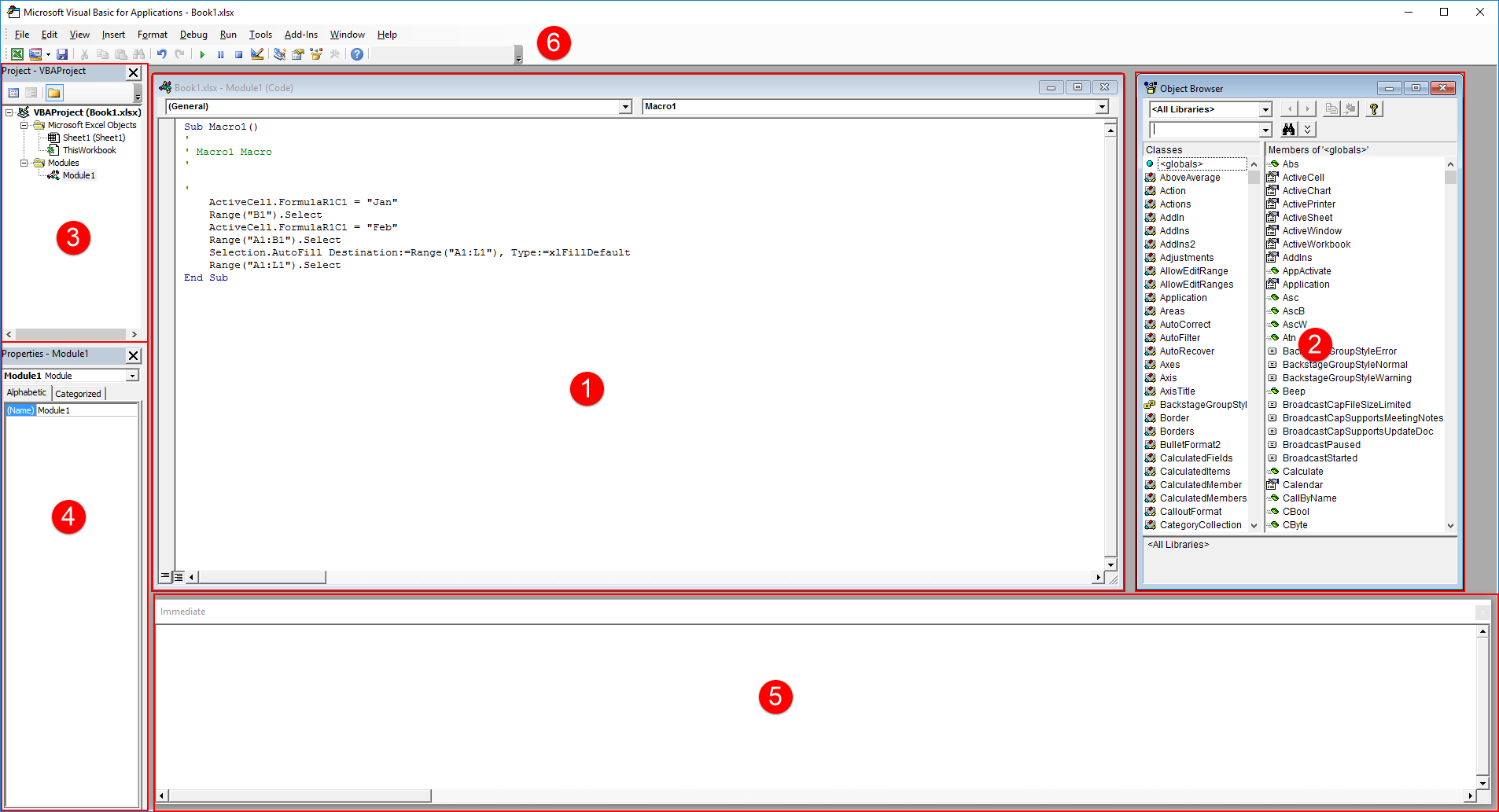
If you are creating a macro that you will need to use with multiple workbooks, the Personal Macro workbook is the best option. Keep in mind that this storage area is unique to your computer and your user account, so while you will always have access to them on your computer, they won’t be stored with the workbook.

Visual Basic for Applications

**Visual Basic for Applications** (VBA) is both a programming language that is commonly used to create macros for Microsoft Office applications and an integrated development environment (IDE). Even if you create a macro using the Macro Recorder, it is still translated into VBA code. You can later edit the VBA code that was generated by the Macro Recorder using the VBA development environment (the Visual Basic Editor).

**The Visual Basic Editor**

The Visual Basic Editor is the development environment that is used to construct and edit macros for Excel and other Microsoft Office applications. This application can be opened directly from Excel (Developer → Visual Basic), but it will open in a separate window:

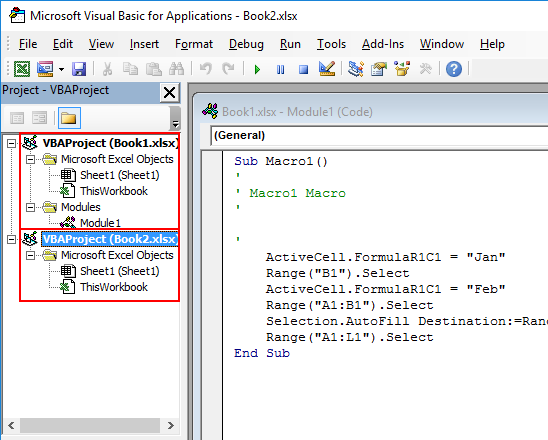


The Visual Basic Editor is comprised of five panels, plus a toolbar area.

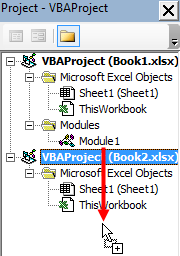
* The **Code window (1)** is where you directly type or edit the code for the open object.
* The **Object Browser (2)** is hidden by default, but will list all of the various modules and classes that can be used.
* The **Project Explorer (3)** pane will list the VBA modules in any currently open documents and templates.
* The **Properties window (4)** will list all of the properties for the object that is selected in the Project Explorer or in the Code window.
* The **Immediate window (5)** will display information when the code is being debugged.
* Finally, the **Standard toolbar (6)** includes many commands that allow you to interact with the VBA environment in a variety of different ways.

Copying Macros Between Workbooks

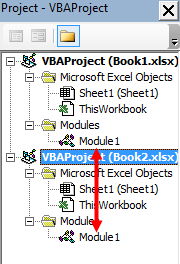
You are able to copy macros between individual workbooks using only Visual Basic for Applications. To do this open both the workbook that contains the macro that you would like to copy, as well as the destination workbook(s) that you would like to copy the macro to. Next, open VBA and examine the Project Explorer pane. Listed inside of this pane will be all of the workbooks that you currently have open (Book1 and Book2 in this example):



Still in the Project Explorer pane, click and drag the module for the macro that you would like to copy from its listing to the destination workbook:



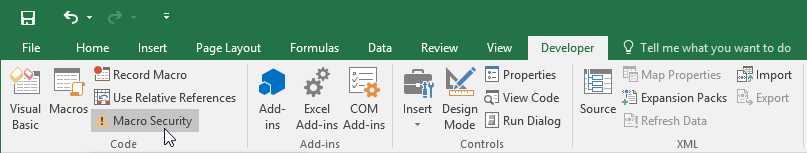
The macro will now be listed inside the original workbook, as well as the destination workbook(s):



Macro Security Settings

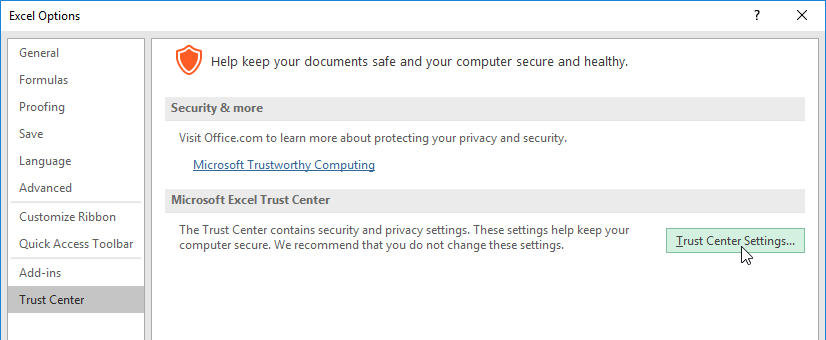
Excel includes several safeguards that help protect you against viruses that can be transmitted through macros. One of the most important safeguards that Excel uses is the digital signature, which allows users to verify that macros are from a trustworthy source. This way, if you open a workbook that contains macros, you can verify their source before enabling them. This can help prevent virus infection as macros from untrustworthy sources are at high risk of containing malicious code.

To access macro security settings, click Developer → Macro Security:

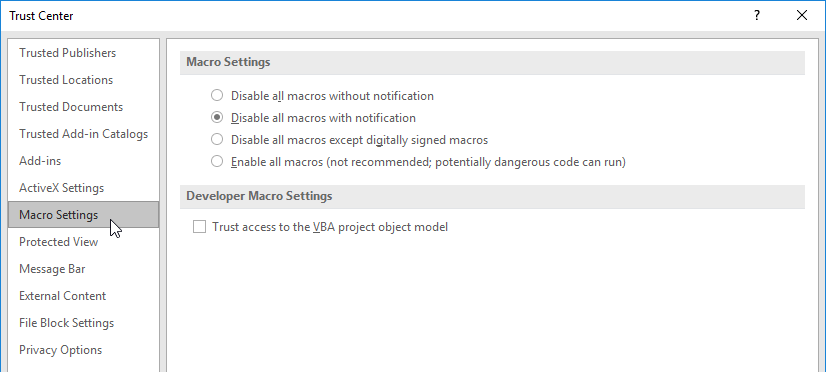


(The Developer tab is hidden by default and needs to be enabled through the Customize Ribbon category of the Excel Options dialog box.)

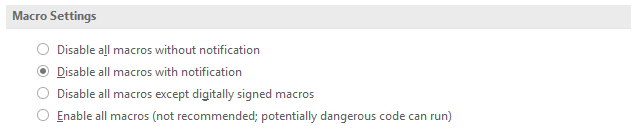
Alternatively, you can access macro security settings by clicking File → Options to open the Excel Options dialog box. Next, open the Trust Center category and click the Trust Center Settings button:



In either case, the Trust Center dialog box will be displayed on your screen. Click the Macro Settings category if it has not already been selected:



Inside this category of the Trust Center, there are four radio buttons to choose from, each of which handles macros in different ways:

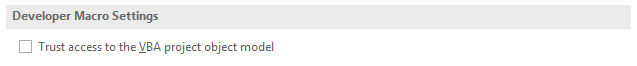


These four buttons are arranged in descending order from most secure to least secure, with the default setting being “Disable all macros with notification.” This setting is a good balance between disabling all macros and enabling all macros.

If you disable all macros, you will not be able to use macros at all, losing any functionality that is provided by them. However, enabling all macros is very dangerous; this setting will greatly increase the risk of running malicious macros.

The middle two options are generally the most commonly used. The third option, “Disable all macros except digitally signed macros,” is an especially good choice. This will run any macro that has been digitally signed by its author, but not any that are unsigned. This ensures that the macro being run is from a trusted source.

The Developer Macro Settings section contains an additional checkbox:



This option is used to control access to the VBA object model from any macro. By ensuring that this box is deselected, you can prevent malicious code from self-replicating. If there is a macro that requires access to this object model, this checkbox must be selected. By default it is deselected, and is best left as such, except for unique circumstances.

Activity 1-2

Creating a Macro

You need to create a macro that will protect your worksheet from changes made by other users.

|  |  |
| --- | --- |
|  | To begin, open Activity 1-2 using Microsoft Excel 2016: |
|  |  |
|  | Open the Record Macro dialog box by clicking View → Macros → Record Macro: |
|  |  |
|  | The Record Macro dialog box will now be on your screen. Type “Workbook\_Protection” into the “Macro name” text box: |
|  |  |
|  | Leave the “Shortcut key” text box blank. Click the “Store macro in” drop-down menu and click This Workbook: |
|  |  |
|  | Click OK: |
|  |  |
|  | The macro is now recording all of your actions in the Excel window. Enable workbook protection by clicking Review → Protect Workbook: |
|  |  |
|  | This action will open the Protect Structure and Windows dialog box. Leave the default settings unchanged and click OK: |
|  |  |
|  | With the actions that you would like to record now complete, click the Stop Recording button () on the status bar in the lower left-hand corner of the Excel window: |
|  |  |
|  | Now you should test this new macro to ensure that it operates correctly. Disable the Protect Workbook option by clicking Review → Protect Workbook: |
|  |  |
|  | Open the Macros dialog box by clicking View → Macros: |
|  |  |
|  | With the Macro dialog box now open, click to select the macro that you just recorded (Workbook\_Protection). Execute it by clicking the Run button: |
|  |  |
|  | Click the Review tab. You will see that the Protect Workbook option has been toggled back on by the macro: |
|  |  |
|  | Suppose that you want to edit this macro to take advantage of the worksheet protection feature that will enforce the size of the Excel window. Open the Macro dialog box by clicking View → Macros: |
|  |  |
|  | In the Macro dialog box, click to select the Workbook\_Protection macro. Click Edit: |
|  |  |
|  | Visual Basic for Applications will now be open with the macro module visible in the Code window: |
|  |  |
|  | Click inside the Windows:=False line and replace False with True: |
|  |  |
|  | Close the Visual Basic Editor window by clicking the Close button in the upper right-hand corner: |
|  |  |
|  | Save your changes as Activity 1-2 Complete using the Excel Macro-Enabled Workbook file type. Close Microsoft Excel 2016. |

TOPIC C: Apply Conditional Formatting

While Excel workbooks and worksheets can contain all sorts of important and enlightening data, those important bits can sometimes be lost in a sea of information. To help bring attention to patterns or outliers in your data, Excel allows you to apply conditional formatting. Conditional formatting is often used to highlight interesting and relevant data, but it can go much further. During this topic you will learn how conditional formatting works in Excel 2016, as well as how to apply it.

Topic Objectives

In this topic, you will learn:

* About conditional formatting
* About the conditional formats that are available in Excel 2016
* About the Conditional Formatting Rules Manager dialog box
* About the New Formatting Rule dialog box
* About the Clear Rules command

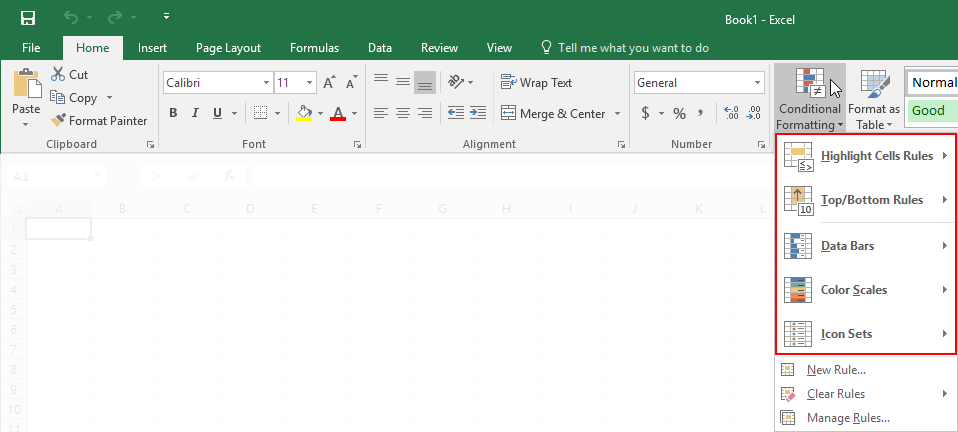
Conditional Formatting

**Conditional formatting** is a formatting type that will highlight cells whose data satisfies certain criteria. It is useful for worksheets that show financial figures (where black formatting might equal positive values, while red formatting might equal negative values), test results (where results change color based on their value), or any time you want to highlight the data that falls within a certain value.

While Excel 2016 includes some default criteria that you can use to apply conditional formatting quickly, you also have the option to create your own criteria and conditional formatting style from scratch.

Conditional Formats

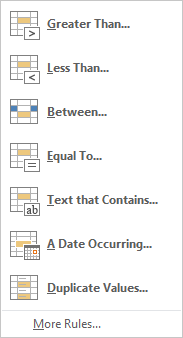
There are a number of different conditional formats that you can quickly apply by clicking Home → Conditional Formatting. Here you will see a number of format types that you can choose from:



Simply choose a rule type and then click a specific rule to apply. Here is a breakdown of all the conditional formatting rule types that are found in this menu.

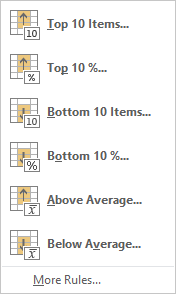
Highlight Cells Rules

This will highlight cells that are greater than, less than, between, or equal to specified values:



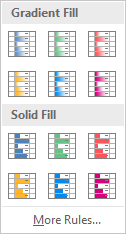
Top/Bottom Rules

This option will allow you to highlight the top or bottom numbers or percentages in the selected cells:



Data Bars

Will display colored bars that are indicative of the value in the cell. The more the cell is filled in, the higher its value compared to the other cells in the range:



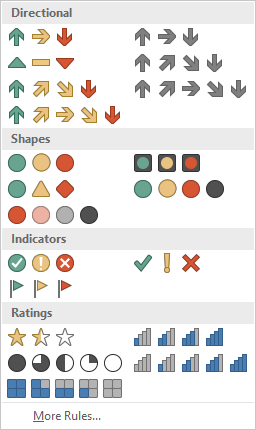
Color Scales

Will use different shades of color to represent different values, from low to high:

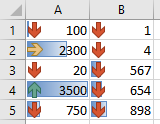


Icon Sets

Will use sets of similar icons that will visually indicate a cell’s value:

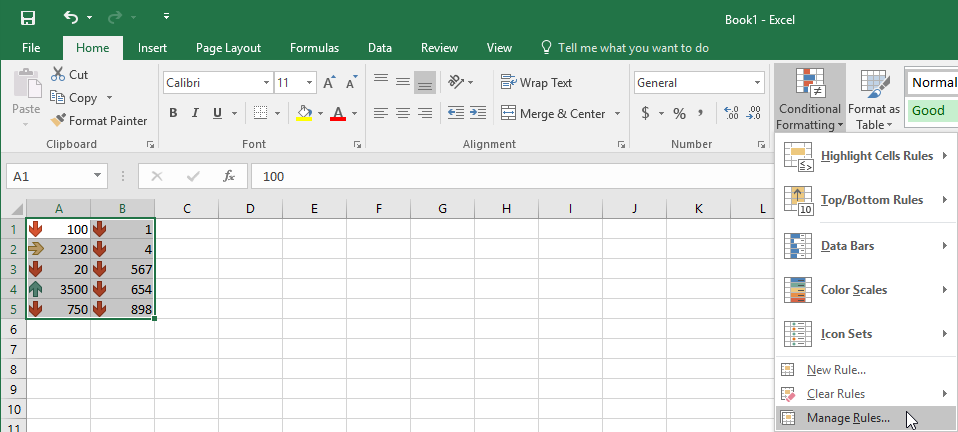


You can apply multiple conditional formatting rules to a group of cells by simply re-selecting the group and applying another condition. Here, we have added an icon set to data that already contains data bars:

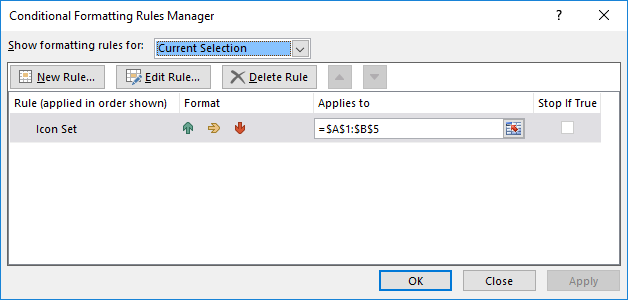


The Conditional Formatting Rules Manager Dialog Box

As you create and apply conditional formatting rules to your data, you will need to be able to manage them properly. The Conditional Formatting Rules Manager has been provided for just this purpose. It can be accessed by clicking Home → Conditional Formatting → Manage Rules:



The Conditional Formatting Rules Manager will list any conditional formatting rules that have been applied to the current selection. It also includes buttons to edit, delete, and create conditional formatting rules:

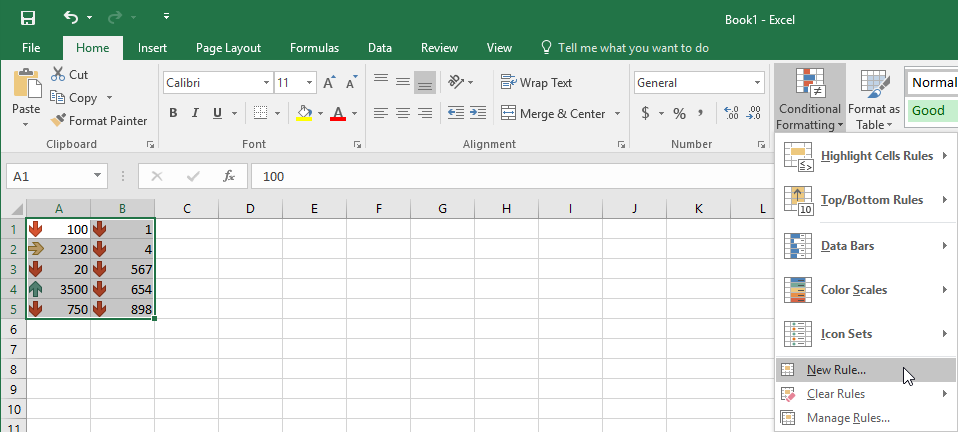


Below, you can see a breakdown of each button in the Conditional Formatting Rules Manager dialog box:

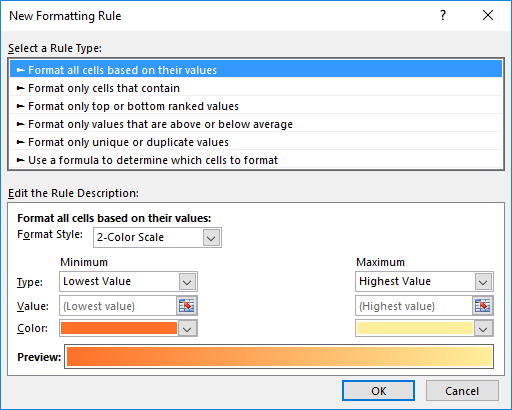
|  |  |
| --- | --- |
| “Show formatting rules for” drop-down menu | This drop-down menu is used to choose the workbook element for which you want to display formatting rules. This can be the current selection of cells, the current worksheet, or specific worksheets in the current workbook. |
| New Rule button | Clicking this button will open the New Formatting Rule dialog box, which can then be used to create a new conditional formatting rule. |
| Edit Rule button | Clicking this button will open the Edit Formatting Rule dialog box. This dialog is identical to the New Formatting Rule dialog box, with the only difference that you use it to edit existing conditional formatting rules. |
| Delete Rule button | This button will delete the selected conditional formatting rule. |
| Move Up and Move Down buttons | These buttons allow you to adjust the order of rule precedence. The top-most rule will be executed before the rules below it. |
| “Rule (applied in order shown)” column | Lists the name of all the conditional rules that have been applied to the current selection (or whichever element has been selected from the “Show formatting rules for” menu). |
| Format column | This column shows you a preview of how the conditional formatting will look for each rule. |
| “Applies to” column | This column will list the cell or range to which each rule has been applied. |
| Stop If True checkbox | This checkbox is used to control what conditional formatting rules are applied from the list. For example, if you had four conditional formatting rules applied to a range of data, you could check the Stop If True checkbox for the second rule to prevent the remaining rules from being applied. |

The New Formatting Rule Dialog Box

The New Formatting Rule dialog box is used to create new formatting rules from scratch. To access this dialog box, click Home → Conditional Formatting → New Rule:



The New Formatting Rule dialog box is divided into two sections, Select a Rule Type and Edit the Rule Description:



The Select a Rule Type section lists six rule types that you can choose from. Clicking on a rule type will change the contents of the Edit the Rule Description section to incorporate controls that are relevant to the selected rule.

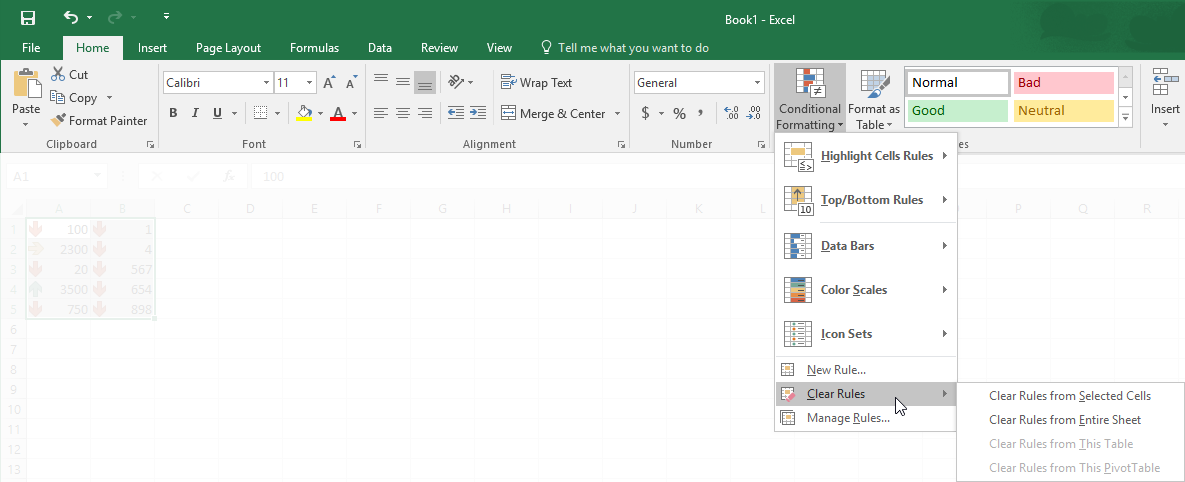
Below you can find a brief description of what each rule type does:

|  |  |
| --- | --- |
| Format all cells based on their values | This is the same rule type that is used in data bars, color scales, and icon sets. It is used to illustrate your data in each cell relative to the range of cells that were selected (low, medium, and high). |
| Format only cells that contain | This rule is used to highlight cells that meet a specified condition. For example, you could use this rule to highlight cells that are lower than a specified value. |

|  |  |
| --- | --- |
| Format only top or bottom ranked values | Using this rule you can highlight a custom number (or percentage) of top or bottom values in a selected range of data. |
| Format only values that are above or below average | This rule is used to highlight values that are above or below the average value of the cells in the selected range. Additionally, this rule also allows you to highlight data that is equal and above the average, equal and below the average, or based on standard deviations of one, two, or three (above or below). |
| Format only unique or duplicate values | This rule will highlight data that is either a duplicate of another value in the selected range or unique to it. |
| Use a formula to determine which cells to format | This rule allows you to use a formula or function to highlight cells. |

Clear Rules

To remove conditional formatting rules, click Home → Conditional Formatting → Clear Rules:



Using the options on the sub-menu, you are able to remove conditional formatting rules from an entire worksheet, selected cells, the selected table, or the selected PivotTable.

Activity 1-3

Applying Conditional Formatting

To help find outliers and trends within a weekly sales and bonus payout worksheet, you need to apply some conditional formatting.

|  |  |
| --- | --- |
|  | To begin, open Activity 1-3 from your Exercise Files folder: |
|  |  |
|  | You first want to use color scales to highlight the sales associates that made above average in sales and those that were below average. Use your cursor to select cells D5:D14: |
|  |  |
|  | Next, click Home → Conditional Formatting → Color Scales. From the Color Scales gallery, click the first option: |
|  |  |
|  | You will immediately see that those sales associates who have made more than average in sales will be highlighted in green, while those who made below are highlighted in red. Those that were close to the average are a yellow or orange shade: |
|  |  |
|  | Next, you need to create a custom conditional formatting rule that will use an icon set to identify sales associates who met or exceeded their weekly goal, as well as those who met or exceeded the minimum sales goal requirement. Ensure that the D5:D14 range is selected: |
|  |  |
|  | Next, click Home → Conditional Formatting → New Rule: |
|  |  |
|  | The New Formatting Rule dialog box will now be displayed. Click to select the “Format all cells based on their values” rule type: |
|  |  |
|  | Within the Edit the Rule Description section, click the Format Style drop-down menu and then click the Icon Sets option: |
|  |  |
|  | The controls in the Edit the Rule Description section will change to incorporate controls that are relevant to icon sets. On the row that has the green icon, click the Type drop-down menu and then click the Number option: |
|  |  |
|  | On the same row, inside the Value text box, type “15000” (the weekly goal):    (Note that you could also reference a cell that contains the value that you need.) |
|  |  |
|  | On the row that has the yellow dot icon, click the Type drop-down menu and then click the Number option: |
|  |  |
|  | On the same row, type “10000” (the minimum weekly target) into the Value text box: |
|  |  |
|  | With the new custom conditional formatting rule configured, click OK: |
|  |  |
|  | The new rule will now be applied to the selected range of data. (You may need to resize the column so that there is room for the icon set.) In this case you will see sales associates who made or exceeded their goal, those who are slipping below the base goal, and those who have not met either goal type: |
|  |  |
|  | Save your work as Activity 1-3 Complete and then close Microsoft Excel 2016. |

TOPIC D: Add Data Validation Criteria

By restricting the type of data that can be entered into your workbook, you can prevent data errors and have greater control over your data. For example, some data may need to be whole numbers while some may need to be a decimal value. Over the course of this topic, you will learn about data validation and how it is used in Excel 2016.

Topic Objectives

In this topic, you will learn:

* About data validation
* About the Data Validation dialog box

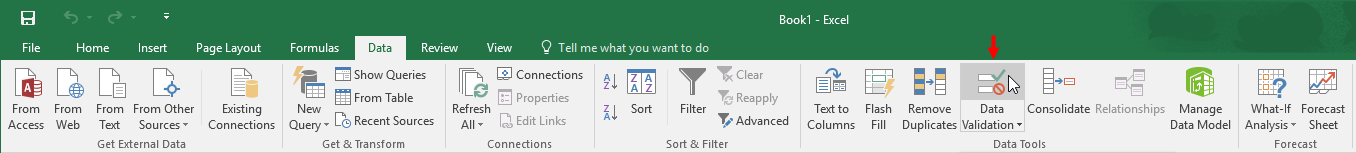
Data Validation

**Data validation** is an important part of worksheet design and functionality. It is used to make sure that the right data is being entered in the right place. For example, if you had a database table that contained a column of sales figures, you wouldn’t want to accidentally be able to enter text in the same column. To make sure problems like this don’t occur, Excel lets you set up **validation rules** to make sure the right data is being entered using the right format in the right location.

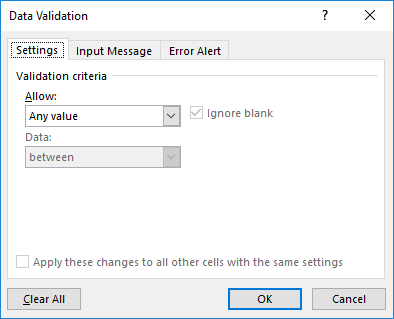
Instructor Tip: Validation rules can be applied anywhere in a worksheet. If you or someone else enters information that is not properly formatted or of the right type, you can even customize error messages that inform the user what went wrong.

The Data Validation Dialog Box

To set up data validation, you first need to open the Data Validation dialog box. This is done by clicking Data → Data Validation:



The Data Validation dialog box displays three tabs: Settings, Input Message, and Error Alert. Using the controls on these tabs you are able to set criteria settings, apply an input message, as well as set an error alert if the wrong data is entered:



Here, you will see the commands and controls in each tab in the Data Validation dialog box and what they can do:

|  |  |
| --- | --- |
| Settings | This tab lets you define the validation rule. The Allow field in particular is used to specify the data type:   * **Any value**, meaning no restriction on text or numbers * **Whole numbers** defined by a logical condition (<, >, >=, etc.) * **Decimal numbers** defined by a logical condition * **List**, as defined by a range of cells from another location * **Date** defined by a logical condition * **Time** defined by a logical condition * **Text length** with min/max length settings * **Custom** data as defined by a formula specified to your needs   Each input type comes with its own set of conditions so you can specify exactly what is allowed in this cell. |
| Input Message (optional) | Use this option to add a notification to the cell alerting users what the cell is for and what type of input is allowed. You can enter a title and a short, descriptive message. |
| Error Alert (optional) | You can use this tab to write an error message explaining the details of what went wrong. This message will be displayed if someone does enter incorrect data. Usually the error message will state the limits of input for this cell. |

Activity 1-4

Adding Data Validation Criteria

You are building a worksheet that will record the employee review ratings for each employee in your team. As ratings are to be recorded as values between 1 and 10, you need to add data validation to ensure that nothing else is entered other than whole numbers within that range.

|  |  |
| --- | --- |
|  | To begin, open Activity 1-4 from your Exercise Files folder: |
|  |  |
|  | First you need to select the range of cells to which you want to apply data validation. Use your cursor to select cells B2:B10 on the current worksheet: |
|  |  |
|  | Display the Data Validation dialog box by clicking Data → Data Validation: |
|  |  |
|  | First, you need to define the rule for data in the cell. Click the Allow drop-down menu and click the “Whole number” option: |
|  |  |
|  | As you can see, once the data type has been set, there are input fields available to set the rules based on whatever operator has been selected in the Data combo box. By default, “between” is selected:    Leave this option unchanged. |
|  |  |
|  | Inside the Minimum and Maximum text boxes, you are able to enter values directly. In this case the numbers need to be between 1 and 10, so type “1” into the Minimum text box and “10” into the Maximum text box: |
|  |  |
|  | Click the Input Message tab: |
|  |  |
|  | In this tab you can construct a message that will inform users what type of data is acceptable in the affected cells. Inside the Title text box, type “Review Rating:” |
|  |  |
|  | Within the “Input message” text area, type the following: “Please enter a number between 1 and 10.” |
|  |  |
|  | With the input message now configured, click the Error Alert tab: |
|  |  |
|  | Leave the Style drop-down menu unchanged. Type “Employee Review Rating” into the Title text box: |
|  |  |
|  | Inside the “Error message” text box, type the following: “Only numbers between 1 and 10 will be accepted.” |
|  |  |
|  | With the data validation options configured, click OK to apply them: |
|  |  |
|  | To test out the new data validation, first click any of the cells that were affected. You will see a ScreenTip appear with the input message you added: |
|  |  |
|  | Click inside cell B2 and type “7” as a value: |
|  |  |
|  | Click to select cell B3. As the value entered into cell B2 falls within the acceptable guidelines you configured, nothing will happen and the value will be accepted. Type “88%” into cell B3: |
|  |  |
|  | Now try to select another cell anywhere on the current worksheet. As the data value you just entered falls outside the acceptable guidelines for the data validation rule, the error message you previously configured will appear: |
|  |  |
|  | Clicking the Retry option will allow you to enter another value within the cell. For this example, click Cancel to clear the data from the cell: |
|  |  |
|  | Save your work as Activity 1-4 Complete and then close Microsoft Excel 2016. |

Summary

Over the course of this lesson you learned about a variety of different ways that you can automate worksheet functionality. You should now be comfortable creating and editing basic macros, applying conditional formatting, as well as adding data validation criteria.

Review Questions

1. How do you open the Workbook Properties dialog box?

To open the Workbook Properties dialog box, click File → Info. Next, click the Properties drop-down command and click Advanced Properties.

1. In brief, what are macros?

Macros are small programs that are created to complete a specific task or set of tasks.

1. Can macro names contain spaces?

No. Macro names cannot contain spaces or any special characters except for the underscore (\_). Additionally, macro names cannot start with a number or share the same name as a function or other object in VBA.

1. What will the Data Bars conditional format do once it is applied to a set of data?

This conditional format will insert small colored bars into each cell that are indicative of the value in the cell. The more the cell is filled in, the higher its value compared to the other cells in the range.



Lesson Labs

Lesson 1

Lesson Lab 1-1

|  |  |
| --- | --- |
| Objective | To understand how to update workbook properties, as well as how to create and edit a macro. |
| Briefing | You have started working with a workbook that tracks quarterly sales across each store in your company. You need to create a macro that will quickly protect workbooks and check for spelling errors. Additionally, you need to apply a new title and add your name to the workbook properties. |
| Task | Open the Advanced Properties dialog box and type “National First Quarter Sales” into the Title text box. Type your name inside the Author text box.  Using the Macro Recorder, create a basic macro labeled “Protect\_Worksheet” that will protect your workbook from changes, as well as check it for spelling errors. |
| Hints | To protect your worksheet, click Review → Protect Workbook. Use the default settings. |
| Sample Data |  |

Lesson Lab 1-2

|  |  |
| --- | --- |
| Objective | To understand how to apply conditional formatting and set up data validation. |
| Briefing | You are continuing to work on a workbook that tracks quarterly sales for each store in your company. To make it more readable, you would like to apply some conditional formatting. Additionally, you should apply some data validation. |
| Task | Apply the Green-Yellow-Red color scale conditional format to cells C4:C27.  Apply data validation to cells D4:D27 that will only allow whole numbers between 1 and 100 to be entered. Configure an input message that will warn users that they need to enter a whole number. |
| Sample Data |  |
| Follow-Up Questions | Configure an error alert for the data validation that you applied to cells D4:D27. Ensure that this error message notifies users that only whole numbers will be accepted. |

Course Wrap-Up

Post-Course Assessment

1. Which of the following items cannot be included in a macro name?
   1. Underscores
   2. Numbers
   3. Spaces
   4. Letters

Spaces cannot be included in a macro name.

1. What does a blue tracer arrow indicate when using the Trace Cells feature?
   1. The path of any formula that references cells in another worksheet
   2. The path of data for a valid formula
   3. The path of a formula that has an error within it
   4. None of the above

The blue tracer arrow indicates the path of data for a valid formula.

1. What does the #NAME? formula error type indicate?
   1. The formula does not have any values
   2. The value cannot be divided by zero
   3. The formula contains invalid numeric values
   4. There is unrecognized text in the formula

The #NAME? formula error indicates that unrecognized text appears in the cell.

1. Which of the following is a type of Sparkline?
   1. Line
   2. Block
   3. Cylindrical
   4. Pie

The three types of Sparklines are line, column, and win/loss.

1. What type of add-in is Power View?
   1. Excel
   2. Actions
   3. COM
   4. XML

The Power View is a type of COM add-in.

1. What special character needs to precede the actual cell reference when referencing cells in another workbook?
   1. @
   2. !
   3. &
   4. ?

The exclamation mark (!) needs to proceed the actual cell reference when referencing cells in another workbook.

1. What does the Sum consolidate function do?
   1. Divides values
   2. Adds all the values in a list
   3. Finds the square root of a value
   4. Multiplies values in a list

The Sum consolidate function will add all the values in a list.

1. Which of the following is a file format that you can export Excel data to?
   1. PPT
   2. DOC
   3. XML
   4. ASP

You can export Excel data to XML, HTML, TXT, CSV, PDF, and XPS formats, among many others.

1. Which of the following is a closing tag in XML?
   1. </example>
   2. <example>
   3. <example />
   4. </example/>

Of the examples shown, </example> is a closing tag in XML.

1. What are XML schema files used for?
   1. Export workbooks to an XML file
   2. Format a workbook to fit the criteria for an XML file
   3. Create a new workbook from scratch
   4. Define the rules and structure for an XML file

XML schema files are used to define the rules and structure for an associated XML file.

Course Summary

Congratulations on completing the third part of Microsoft Excel 2016 training. During this course, you learned how to:

* Automate worksheet functionality
* Audit worksheets
* Analyze and present data
* Work with multiple workbooks
* Export Excel data
* Import and export XML data

You should now feel comfortable using macros and conditional formatting to automate worksheet functionality. You now know how to audit worksheets using the Trace Cells feature and data list outlines, as well as by troubleshooting invalid data. Additionally, you should feel comfortable analyzing data by creating Sparklines, scenarios, and more. You can now work with multiple workbooks, as well as export Excel data into a variety of different formats.

Appendices

Keyboard Shortcut Quick Reference Sheet

|  |  |  |
| --- | --- | --- |
| File Management | Open a new workbook | Ctrl + N |
| Save a file | Ctrl + S |
| Open a file | Ctrl + O |
| Print worksheet | Ctrl + P |
| Close Microsoft Excel | Alt + F4 |
| Worksheet Management | Switch between worksheet tabs (left to right) | Ctrl + Page Up |
| Switch between worksheet tabs (right to left) | Ctrl + Page Down |
| Insert cells | Ctrl + Shift + + |
| Delete cells | Ctrl + - |
| Hide the selected rows | Ctrl + 9 |
| Hide the selected columns | Ctrl + 0 |
| Select all items in current worksheet | Ctrl + A |
| Toggle between displaying values/formulas | Ctrl + ` |
| Open Dialogs | Open Find and Replace dialog | Ctrl + F |
| Open Go To dialog | Ctrl + G |
| Open Format Cells dialog to last used tab | Ctrl + 1 |
| Open Data Validation dialog | Alt + D + L |
| Open Excel Options dialog box | Alt + F + T |

|  |  |  |
| --- | --- | --- |
| Text Formatting and Editing Tools | Copy text | Ctrl + X |
| Cut text | Ctrl + C |
| Paste text | Ctrl + V |
| Undo last action | Ctrl + Z |
| Redo last action | Ctrl + Y |
| Apply border to selected cells | Ctrl + Shift + & |
| Remove border from selected cells | Ctrl + Shift + \_ |
| Apply italic formatting | Ctrl + I |
| Apply bold formatting | Ctrl + B |
| Apply underlining | Ctrl + U |
| Check spelling | F7 |
| Auditing Tools | View precedent cells | Alt + M + P |
| View dependent cells | Alt + M + D |
| Remove arrows | Alt + M + A + A |
| Evaluate formula | Alt + M + V |
| Recalculate current sheet | Shift + F9 |
| Recalculate entire workbook | F9 |
| Macros | Open the Macro dialog | Alt + F8 |
| Open the Visual Basic Editor | Alt + F11 |
| Close the Visual Basic Editor | Alt + F4 |
| Run a Visual Basic program | F5 |

Glossary

absolute reference

A type of reference that will not change even if it is moved or copied to another location.

add-in

Small components that can be added to Excel to add features and functionality.

array

Any grouping of two or more adjacent cells.

arguments

Data used by functions to complete calculations.

AutoFill

A feature that is used to automatically fill sequential data into a range of cells.

Anova

Short for Analysis Of Variance. Used to examine if the averages of samples are different in a significant way.

cell

The intersection of a row and column on a worksheet.

charts

Visual representations of numeric data in a dataset.

conditional formatting

A formatting type that will highlight cells whose data satisfies certain criteria.

consolidation

The process of combining, condensing, and summarizing data from multiple sources into one destination.

correlation

Indicates if data sets trend or change with each other.

delimited text

Data that is entered with one row equaling one line of text.

dependent cells

Cells that are affected by the contents of another cell.

exporting

The process of sending data from one application to another.

external reference

A link to the contents of one or more cells within the worksheet of another workbook.

filtering

Removing data from view based upon set criteria.

form

An interface element of a workbook that is used to collect data or execute an action.

Format Painter

A tool that is used to copy formatting from one selection of text to another.

formula

A mathematical relationship expressed through symbols.

Formula Bar

A part of the Excel interface that displays the cell name, as well as values and formulas in the selected cell.

function

A preconfigured formula that is used for a specific purpose.

HTML

A programming language that describes how to display data.

importing

The process of opening data in one application that was saved in another application.

logical operators

A type of operator that is used to compare values and determine if those values meet specified criteria.

macros

Small programs that are created to complete a specific task or set of tasks.

outline

A feature in Excel that allows you to organize datasets in a worksheet into hierarchical groups.

precedent cells

Cells within a worksheet that provide data for a formula.

range names

Meaningful names that can be added to cell ranges so that they can be easily referred to and understood later.

relative reference

A cell reference that will change relative to its positioning in a worksheet.

scenario

A set of cell values that are saved and substituted into your worksheet at your convenience.

Sparklines

Small graphs contained within a single cell that are used to summarize data and display trends.

table

A dataset that is comprised of rows and columns, but is treated as one object (unlike regular data ranges).

tracer arrows

Colored arrows used to indicate the direction of the data flow to and from cells and formulas.

workspace

Saved set of Excel files.

XML

A commonly used programming language that is frequently used to describe data. Stands for Extensible Markup Language.

Index

C

Conditional Formatting

Applying 43, 51

Clearing 50

Creating custom rule 48, 53

Managing rules 47

Overview of available formats 43

What is? 42

CSV 75

D

Data Validation

Creating rule 59, 62

Example of 68

Setting error alert 60, 66

Setting input message 60, 65

What is? 59

E

Excel Options Dialog Box

Customize Ribbon category 32

Opening 32

Trust Center category 32

F

File Properties

Dialog overview 12

Modifying 13, 19

Viewing 11, 17

H

HTML 75, 81

M

Macro

Editing with Visual Basic for Applications 28, 40

Naming conventions 28

Recording 27, 35

Running 39

Security settings 32

Storage location 28

Supported file formats 28

What is? 26

Metadata 11

P

PDF 75

T

Trust Center 32, 33

V

Visual Basic Editor *See* Visual Basic for Applications

Visual Basic for Applications 26, 28

Editing macro with 40

Interface overview 28

Trusting access to the project object model 34

What is? 28

X

XML 75, 76, 77, 81

Schema files 76

XPS 75