Working with Data Series
Using Advanced Functions
Creating/Revising PivotTables
Using Scenarios and Goal Seeking
Using Data Tables
Consolidating Worksheets
Solving Problems
Working with Views
Using Auditing Tools
Using Macros
Recording Macros
Editing Macros
Creating Macro Buttons
ABOUT ONDEMAND SOFTWARE

The OnDemand Software Division of Global Knowledge is the worldwide leader of software solutions that enable rapid adoption, broad acceptance and increased accuracy in the use of enterprise applications related to enterprise resource planning, supply chain management, procurement, e-commerce and customer relationship management.

The flagship product, OnDemand Personal Navigator™, provides one synchronized documentation, training and performance support platform. Founded in 1986, the OnDemand Software Division has over 850 Global 2000 customers in 12 countries. The OnDemand Software Division of Global Knowledge, a Welsh, Carson, Anderson and Stowe portfolio company, is headquartered in King of Prussia, Pennsylvania, with offices located worldwide. Additional information can be found at http://www.ondemandgk.com.

COPYRIGHT

Copyright © Global Knowledge Software LLC. 2008. All rights reserved. Information in this document is subject to change without notice and does not represent a commitment on the part of Global Knowledge.

No part of this publication, including interior design, cover design, icons or content may be reproduced by any means, be it transmitted, transcribed, photocopied, stored in a retrieval system, or translated into any language in any form, without the prior written permission of Global Knowledge Network, Inc.

Products mentioned herein, including SAP R/3, PeopleSoft, Siebel Systems, Microsoft Windows, Microsoft Office, Microsoft FrontPage, Microsoft Outlook, Macromedia Flash, Adobe Acrobat, and JavaScript are trademarks or registered trademarks of their respective owners.

OnDemand Personal Navigator and Courseware Express are trademarks of Global Knowledge Software LLC. CustomDOC and Knowledge Pathways are registered trademark of Global Knowledge Software LLC. Global Knowledge and the Global Knowledge logo are trademarks of Global Knowledge, Inc.

DISCLAIMER

Global Knowledge has taken every effort to ensure the accuracy of this manual. If you should discover any discrepancies, please notify us immediately.

Global Knowledge Software LLC.
OnDemand Software Division
475 Allendale Road
King of Prussia, PA 19406
(610) 337-8878
www.ondemandgk.com
LESSON 1 - WORKING WITH DATA SERIES ...........................................1
Using the Series Command ...............................................................2
Creating a Linear Series ...................................................................3
Creating a Date Series .......................................................................5
Using a Stop Value ............................................................................7
Creating a Growth Trend Series ........................................................9
Exercise ..............................................................................................12
Working with Data Series .................................................................12

LESSON 2 - USING ADVANCED FUNCTIONS .....................................13
Using Lookup Functions ....................................................................14
Using the VLOOKUP Function .............................................................14
Using the HLOOKUP Function ............................................................18
Using the IF Function ........................................................................21
Using Nested IF Functions ................................................................25
Using the ISERROR Function ...............................................................28
Using an AND Condition with IF .......................................................31
Using an OR Condition with IF .........................................................33
Using the ROUND Function ..............................................................36
Limiting the Precision of Numbers ....................................................38
Exercise ..............................................................................................41
Using Advanced Functions ...............................................................41

LESSON 3 - CREATING/REVISING PIVOTTABLES ...............................43
Creating a PivotTable Report .............................................................44
Adding PivotTable Report Fields .......................................................46
Selecting a Report Filter Field Item ..................................................49
Refreshing a PivotTable Report .........................................................51
Changing the Summary Function ......................................................52
LESSON 7 - SOLVING PROBLEMS

Using Solver ................................................................. 112
Saving a Solution as a Scenario ........................................ 117
Restoring the Original Values ........................................... 118
Changing a Constraint .................................................... 119
Creating a Solver Report .................................................. 120
Viewing Solutions using Scenarios ..................................... 122
Exercise ........................................................................ 124
Solving Problems ............................................................ 124

LESSON 8 - WORKING WITH VIEWS ................................ 125

Using Views ..................................................................... 126
Creating a Normal View ................................................... 126
Creating a Custom View .................................................. 128
Displaying a View ........................................................... 129
Deleting a Custom View ................................................... 130
Exercise ........................................................................ 133
Working with Views ........................................................ 133

LESSON 9 - USING AUDITING TOOLS ........................... 135

Showing Formulas ............................................................ 136
Displaying/Removing Dependent Arrows ......................... 138
Displaying/Removing Precedent Arrows ......................... 141
Using the Trace Error Button ......................................... 145
Tracing Cells Causing Errors .......................................... 148
Using the Evaluate Formula Button ............................... 151
Using the Error Checking Button .................................... 156
Using the Watch Window ................................................ 159
Exercise ........................................................................ 164
Using Auditing Tools ....................................................... 164

LESSON 10 - USING MACROS ....................................... 167

Defining Macros ................................................................ 168
Changing Macro Security Settings ................................... 169
Adding a Folder as a Trusted Location .............................. 172
Opening a Workbook Containing Macros ....................... 176
Running a Macro ............................................................ 179
Using a Shortcut Key ...................................................... 181
Displaying the Developer Tab ......................................... 183
Opening the Visual Basic Editor Window ......................... 184
Using the Visual Basic Editor Window .............................. 187
Exercise ......................................................................... 189
Using Macros .................................................................. 189

LESSON 11 - RECORDING MACROS ........................................ 191
Recording a Macro .......................................................... 192
Assigning a Shortcut Key .................................................. 196
Using Relative References .............................................. 198
Deleting a Macro ............................................................ 201
Exercise ......................................................................... 204
Recording Macros .......................................................... 204

LESSON 12 - EDITING MACROS ............................................. 205
Writing a New Macro ....................................................... 206
Entering Macro Comments .............................................. 208
Copying Macro Commands ............................................. 211
Editing Macro Commands .............................................. 213
Typing Macro Commands .............................................. 215
Running a Macro from the Code Window ......................... 218
Exercise ......................................................................... 220
Editing Macros .............................................................. 220

LESSON 13 - CREATING MACRO BUTTONS .............................. 223
Using a Macro Button ....................................................... 224
Creating a Macro Button .................................................. 225
Copying a Macro Button .................................................. 227
Formatting a Macro Button .............................................. 229
Moving/Sizing a Macro Button ........................................ 231
Deleting a Macro Button .................................................. 233
Adding a Macro to the Quick Access Toolbar .................... 234
Changing a QAT Macro Button Image ............................... 237
Deleting a Macro Button from the QAT ............................ 240
Exercise ......................................................................... 242
Creating Macro Buttons .................................................. 242
In this lesson, you will learn how to:

- Use the Series command
- Create a linear series
- Create a date series
- Use a stop value
- Create a growth trend series
**USING THE SERIES COMMAND**

**Discussion**

If you want to enter a sequence of values in a worksheet, you can use the Series command. Although you can create a simple series using the fill handle, the Series command offers additional options.

In the Series dialog box, you can choose one of the following series types: Linear, Growth, Date, or AutoFill. You can also set a step value (the amount by which each step in the series increases or decreases) and a stop value (the value at which the series ends).

The following table displays how Excel completes a series, based on the selected cell(s):

<table>
<thead>
<tr>
<th>Series</th>
<th>Selection</th>
<th>Extended Series</th>
<th>Step Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>1</td>
<td>2, 3, 4, etc.</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1,4</td>
<td>7, 10, 13, etc.</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1000, 975</td>
<td>950, 925, 900, etc.</td>
<td>-25</td>
</tr>
<tr>
<td>Growth</td>
<td>1, 2</td>
<td>4, 8, 16, etc.</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>1, 4</td>
<td>16, 64, 256, etc.</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>1100, 1210, 1331, etc.</td>
<td>1.1</td>
</tr>
<tr>
<td>Date</td>
<td>1 Jan</td>
<td>2 Jan, 3 Jan, 4 Jan, etc.</td>
<td>1 Day</td>
</tr>
<tr>
<td></td>
<td>1/1/99</td>
<td>1/2/99, 1/3/99, 1/4/99, etc.</td>
<td>1 Day</td>
</tr>
<tr>
<td></td>
<td>1/1/99, 2/1/99</td>
<td>3/1/99, 4/1/99, 5/1/99, etc.</td>
<td>1 Month</td>
</tr>
<tr>
<td>AutoFill</td>
<td>Qtr 1</td>
<td>Qtr 2, Qtr 3, Qtr 4, etc.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sample 1</td>
<td>Sample 2, Sample 3, Sample 4, etc.</td>
<td></td>
</tr>
</tbody>
</table>

If you select a range before you open the Series dialog box, the Series in option automatically reflects the direction, and you do not need to enter a stop value; the series ends when the range is filled. If a range is not selected, the Series in option defaults to Rows, and you must enter the direction, as well as the stop value. For an AutoFill series, however, you must always select the desired range before you open the Series dialog box.
CREATING A LINEAR SERIES

Discussion

When you create a linear series, the values are increased or decreased by a specified value. By default, the step value is 1. For example, if you enter the number 1 in the first cell of a range, Excel will enter 2, 3, 4, etc. in the remaining cells of the range to complete the series. If you enter the number 10 in the first cell and enter a step value of 10, Excel will fill the subsequent cells with 20, 30, 40, etc.
Procedures

1. Select the cell in which you want a linear series to begin.
2. Type the first value in the series.
3. Press [Enter].
4. Select the range you want to fill.
5. Select the Fill button in the Editing group on the Home tab.
6. Select the Series command.
7. Under Type, select the Linear option.
8. Enter the desired step value in the Step value box.
9. Select OK.

Step-by-Step

From the Student Data directory, open SALES71.XLSX.
Create a linear series.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell in which you want a linear series to begin.</td>
<td>Click cell A4</td>
</tr>
<tr>
<td>The cell is selected.</td>
<td></td>
</tr>
<tr>
<td>2. Type the first value in the series.</td>
<td>Type 1</td>
</tr>
<tr>
<td>The text appears in the cell and on the formula bar.</td>
<td></td>
</tr>
<tr>
<td>3. Press [Enter].</td>
<td>Press [Enter]</td>
</tr>
<tr>
<td>The text is entered into the cell.</td>
<td></td>
</tr>
<tr>
<td>4. Select the range you want to fill.</td>
<td>Drag across A4:A15</td>
</tr>
<tr>
<td>The range is selected.</td>
<td></td>
</tr>
<tr>
<td>5. Select the Fill button in the Editing group on the Home tab.</td>
<td>Click Fill</td>
</tr>
<tr>
<td>The Fill menu opens.</td>
<td></td>
</tr>
<tr>
<td>6. Select the Series command.</td>
<td>Click Series</td>
</tr>
<tr>
<td>The Series dialog box opens.</td>
<td></td>
</tr>
<tr>
<td>Steps</td>
<td>Practice Data</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>----------------------------------------</td>
</tr>
<tr>
<td>7. Under <strong>Type</strong>, select the <strong>Linear</strong> option.</td>
<td>Click ☑ Linear, if necessary</td>
</tr>
<tr>
<td><em>The Linear option is selected, and the text in the Step value box is selected.</em></td>
<td></td>
</tr>
<tr>
<td>8. Enter the desired step value in the <strong>Step value</strong> box.</td>
<td>Type 1, if necessary</td>
</tr>
<tr>
<td><em>The text appears in the Step value box.</em></td>
<td></td>
</tr>
<tr>
<td>9. Select <strong>OK</strong>.</td>
<td>Click ![OK button]</td>
</tr>
<tr>
<td><em>The Series dialog box closes, and the range is filled with the linear series.</em></td>
<td></td>
</tr>
</tbody>
</table>

Click anywhere in the worksheet to deselect the range.

**CREATING A DATE SERIES**

**Discussion**

When you create a date series, the values are incremented by one of four available date units: **Day**, **Weekday**, **Month**, or **Year**. For example, if you enter **Jan-07** in the first cell of a range and select **Month** as the date unit, Excel fills the remaining cells of the range with **Feb-07**, **Mar-07**, etc.

Excel recognizes a wide variety of date formats. For instance, if you enter **Jan 1** as the starting value, Excel will recognize it as a date and create the series accordingly.

![A date series](image-url)
Procedures

1. Select the cell in which you want the first date of the series to appear.
2. Type the first date in the series.
3. Press [Enter].
4. Select the range you want to fill.
5. Select the Fill button in the Editing group on the Home tab.
6. Select the Series command.
7. Under Date unit, select the desired option.
8. Select OK.

Step-by-Step

Create a date series.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the cell in which you want the first date of the series to appear.  
*The cell is selected.* | Click cell B4 |
| 2. Type the first date in the series.  
*The text appears in the cell and on the formula bar.* | Type 1/31/07 |
| 3. Press [Enter].  
*The text is entered into the cell.* | Press [Enter] |
| 4. Select the range you want to fill.  
*The range is selected.* | Drag across B4:B15 |
| 5. Select the Fill button in the Editing group on the Home tab.  
*The Fill menu opens.* | Click Fill menu |
| 6. Select the Series command.  
*The Series dialog box opens.* | Click Series |
Steps | Practice Data
---|---
7. Under **Date unit**, select the desired option.
   *The option is selected.* | Click **Month**, if necessary
8. Select **OK**.
   *The Series dialog box closes, and the range is filled with the linear series.* | Click **OK**

Click anywhere in the worksheet to deselect the range.

### Using a Stop Value

**Discussion**

If you know the value at which you want a series to stop, you do not have to select the range you want to fill. For example, you may want a series of cells to display only the period from 1985 to 2030. The step value determines the amount by which each value in the series is incremented, and the stop value determines the last entry in the series. Excel will end the series when it reaches the stop value.

*A series with a stop value*
Procedures

1. Select the cell in which you want the series to begin.
2. Type the first value in the series.
3. Press [Enter].
4. Select the first cell in the series.
5. Select the Fill button in the Editing group on the Home tab.
6. Select the Series command.
7. Under Series in, select the Rows or Columns option, as desired.
8. Select the text in the Step value box.
9. Enter the desired step value.
10. Select the Stop value box.
11. Enter the desired stop value.
12. Select OK.

Step-by-Step

Use a stop value to create a series.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell in which you want the</td>
<td>Click cell E4</td>
</tr>
<tr>
<td>series to begin.</td>
<td></td>
</tr>
<tr>
<td>The cell is selected.</td>
<td></td>
</tr>
<tr>
<td>2. Type the first value in the series.</td>
<td>Type 1985</td>
</tr>
<tr>
<td>The text appears in the cell and on the</td>
<td></td>
</tr>
<tr>
<td>formula bar.</td>
<td></td>
</tr>
<tr>
<td>3. Press [Enter].</td>
<td>Press [Enter]</td>
</tr>
<tr>
<td>The text is entered into the cell.</td>
<td></td>
</tr>
<tr>
<td>4. Select the first cell in the series.</td>
<td>Click cell E4</td>
</tr>
<tr>
<td>The cell is selected.</td>
<td></td>
</tr>
<tr>
<td>5. Select the Fill button in the Editing</td>
<td>Click</td>
</tr>
<tr>
<td>group on the Home tab.</td>
<td></td>
</tr>
<tr>
<td>The Fill menu opens.</td>
<td></td>
</tr>
</tbody>
</table>
### Creating a Growth Trend Series

**Discussion**

You can create a growth trend for a series of numbers. Growth trends project the values in a series. The more cells you select on which to base your growth trend, the more accurate your growth trend will be.

In a growth trend series, the starting value is multiplied by the step value to arrive at the next value in the series; there must be at least two existing values in order to extrapolate the next value.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 6. Select the **Series** command.  
*The Series dialog box opens.* | Click **Series** |
| 7. Under **Series in**, select the **Rows** or **Columns** option, as desired.  
*The desired option is selected.* | Click ⚬ **Columns** |
| 8. Select the text in the **Step value** box.  
*The text in the **Step value** box is selected.* | Double-click in the **Step value** box |
| 9. Enter the desired step value.  
*The text appears in the **Step value** box.* | Type **5** |
| 10. Select the **Stop value** box.  
*The insertion point appears in the **Stop value** box.* | Press **[Tab]** |
| 11. Enter the desired stop value.  
*The text appears in the **Stop value** box.* | Type **2030** |
| 12. Select **OK**.  
*The Series dialog box closes and the series is created ending at the stop value.* | Click **OK** |
A growth trend series

You can also add a growth trend by selecting the cells on which you want to base the growth trend, dragging the fill handle with the right mouse button to select the range you want to fill, and selecting the Growth Trend command from the shortcut menu.

Procedures

1. Select the cells you want to include in the growth trend series, beginning with at least two cells containing data.
2. Select the Fill button in the Editing group on the Home tab.
3. Select the Series command.
4. Under Series in, select the Rows or Columns option, as desired.
5. Under Type, select the Growth option.
6. Select the Trend option.
7. Select .
### Step-by-Step

Create a growth trend series.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cells you want to include in the growth trend series, beginning with at least two cells containing data. &lt;br&gt; <em>The range for the growth trend series is selected.</em></td>
<td>Drag across F4:F13</td>
</tr>
<tr>
<td>2. Select the Fill button in the Editing group on the Home tab. &lt;br&gt; <em>The Fill menu opens.</em></td>
<td>Click ⌁</td>
</tr>
<tr>
<td>3. Select the Series command. &lt;br&gt; <em>The Series dialog box opens.</em></td>
<td>Click Series</td>
</tr>
<tr>
<td>4. Under Series in, select the Rows or Columns option, as desired. &lt;br&gt; <em>The desired option is selected.</em></td>
<td>Click ⌁ Columns, if necessary</td>
</tr>
<tr>
<td>5. Under Type, select the Growth option. &lt;br&gt; <em>The option is selected.</em></td>
<td>Click ⌁ Growth</td>
</tr>
<tr>
<td>6. Select the Trend option. &lt;br&gt; <em>The Trend option is selected.</em></td>
<td>Click ⌁ Trend</td>
</tr>
<tr>
<td>7. Select OK. &lt;br&gt; <em>The Series dialog box closes, and the range is filled with the growth trend series.</em></td>
<td>Click ⌁ OK</td>
</tr>
</tbody>
</table>

Click anywhere in the worksheet to deselect the range. Close SALES71.XLSX.
EXERCISE

WORKING WITH DATA SERIES

Task

Use data series to create a worksheet.

1. Open OFFER.XLSX.
2. Enter the date 06/01/07 into cell A5 and press [Enter].
3. Create a column date series in the range A5:A9, with a step value of 7 days and a stop value of 06/29/07.
4. Type the number 1 in cell A13; press [Enter]; select the range A13:A18; and create a linear series, using the default step value of 1.
5. Type Jun 07 into cell B13; press [Enter]; select the range B13:B18; and create a monthly date series, using the default step value of 1.
6. Fill the range C13:C18 with a growth trend based on the data in cells C13:C14.
7. Close the workbook without saving it.
LESSON 2 - USING ADVANCED FUNCTIONS

In this lesson, you will learn how to:

- Use lookup functions
- Use the VLOOKUP function
- Use the HLOOKUP function
- Use the IF function
- Use nested IF functions
- Use the ISERROR function
- Use an AND condition with IF
- Use an OR condition with IF
- Use the ROUND function
- Limit the precision of numbers
**USING LOOKUP FUNCTIONS**

**Discussion**

Lookup functions look up values in a lookup table and return a result based on those values. For example, if you need to look up the amount of a health insurance deduction based on an employee’s salary and type of coverage, you can use a lookup function to look up the salary and the type of coverage and return the amount of the deduction.

Before you can use a lookup function, you must create the lookup table elsewhere in the workbook and enter the desired data. This table must be sorted in ascending order.

There are three lookup functions: VLOOKUP, HLOOKUP and LOOKUP. The VLOOKUP function expects the lookup value to be in the first column and only works with vertical tables. The HLOOKUP function expects the lookup value to be in the first row and only works with horizontal tables. The LOOKUP function works with either vertical or horizontal tables.

**USING THE VLOOKUP FUNCTION**

**Discussion**

The VLOOKUP function consists of three required arguments, in the following order: lookup value, table array, and column index number. The lookup value is the value for which you want to find matching data and must appear in the first column of the lookup table; it can be a value, a text string, or a cell reference. The table array is the name or address of the lookup table. The column index number is the number of columns Excel must count over to find the matching value.

For example, you may have a parts table consisting of three columns, with the part numbers in column one and the prices in column three. To look up the price for a specified part number (lookup value), you would enter a column index value of 3; Excel would then look for the lookup value in the first column of the parts table and return the value in the third column of the same row.

In another example, you might want to use the lookup table shown below and a column index number of 2 to look up the percent of commission to be paid to a salesperson, based on various sales levels. For example, if the sales figure you want to look up is 6000, the commission would be 6%; Excel finds the lookup value (6000) in the first column and returns the value in the second column of the same row.
<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SALES</td>
<td>COMMISSION</td>
</tr>
<tr>
<td>2</td>
<td>1000</td>
<td>1%</td>
</tr>
<tr>
<td>3</td>
<td>2000</td>
<td>2%</td>
</tr>
<tr>
<td>4</td>
<td>3000</td>
<td>3%</td>
</tr>
<tr>
<td>5</td>
<td>4000</td>
<td>4%</td>
</tr>
<tr>
<td>6</td>
<td>5000</td>
<td>5%</td>
</tr>
<tr>
<td>7</td>
<td>6000</td>
<td>6%</td>
</tr>
<tr>
<td>8</td>
<td>7000</td>
<td>7%</td>
</tr>
<tr>
<td>9</td>
<td>8000</td>
<td>8%</td>
</tr>
</tbody>
</table>

The VLOOKUP function also has a optional fourth argument: range lookup. This can be either TRUE or FALSE. If the range lookup argument is FALSE, VLOOKUP will find only exact matches. If the range lookup argument is TRUE, or if a range lookup argument is not entered, VLOOKUP can find approximate matches. In this case, the lookup table must be sorted in ascending order by the first column in it; otherwise VLOOKUP may not return the correct value.

If the range lookup argument is TRUE or omitted and the lookup value does not appear in the first column of the lookup table, but falls between two values in it, Excel will use the lower of the two values. If the lookup value is smaller than any value in the first column of the lookup table, Excel returns an error message.

For example, using the lookup table shown above and a column index number of 2, if the sales figure you look up is 5700, the commission would be 5%. Since Excel determines that 5700 is located between the numbers 5000 and 6000, it returns the value in the second column of the same row as the lower number.
Creating a VLOOKUP function

In order to copy a VLOOKUP function to other cells, its table array argument must be an absolute reference. Since named ranges are always absolute references, you can assign a name to your lookup table and use that name in the VLOOKUP function.

Procedures

1. Select the cell in which you want the result of the VLOOKUP function to appear.
2. Type =vlookup and an open parenthesis ( ().
3. Select the cell containing the value you want to look up.
4. Type a comma ( ,).
5. Type the name or address of the lookup table.
6. Type a comma ( ,).
7. Enter the column index number.
8. Type the closing parenthesis ( )).
9. Press [Enter].
Step-by-Step

From the Student Data directory, open \textit{SALES72.XLSX}. Use the VLOOKUP function.

If necessary, display the \textbf{Orders} worksheet.

Select \textbf{prices} from the \textbf{Name Box} (the list box to the left of the formula bar) to view the lookup table.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell in which you want the result of the VLOOKUP function to appear. The cell is selected.</td>
<td>Click cell C5</td>
</tr>
<tr>
<td>2. Type \texttt{=vlookup} and an open parenthesis ((\ ()).) \texttt{=vlookup(} appears in the cell and on the formula bar. As you start typing a function, a Screen Tip is displayed to help you enter valid arguments.</td>
<td>Type \texttt{=vlookup(}</td>
</tr>
<tr>
<td>3. Select the cell containing the value you want to look up. A blinking marquee appears around the cell and its address appears in the VLOOKUP function.</td>
<td>Click cell B5</td>
</tr>
<tr>
<td>4. Type a comma ((,).) The comma ((, )) appears in the VLOOKUP function.</td>
<td>Type \texttt{,}</td>
</tr>
<tr>
<td>5. Type the name or address of the lookup table. The text appears in the VLOOKUP function.</td>
<td>Type \texttt{prices}</td>
</tr>
<tr>
<td>6. Type a comma ((,).) The comma ((, )) appears in the VLOOKUP function.</td>
<td>Type \texttt{,}</td>
</tr>
<tr>
<td>7. Enter the column index number. The column index number appears in the VLOOKUP function.</td>
<td>Type \texttt{2}</td>
</tr>
</tbody>
</table>
Steps | Practice Data
--- | ---
8. Type the closing parenthesis ( ). *The closing parenthesis ( ) appears in the VLOOKUP function.* | Type )
9. Press [Enter]. *The result of the VLOOKUP function appears in the cell.* | Press [Enter]

Copy the formula to the range C6:C13. Then, click anywhere in the worksheet to deselect the range.

**USING THE HLOOKUP FUNCTION**

**Discussion**

The HLOOKUP function consists of three required arguments, in the following order: lookup value, table array, and row index number. The lookup value is the value for which you want to find matching data and must appear in the top row of the lookup table; it can be a value, a text string, or a cell reference. The table array is the name or address of the lookup table. The row index number is the number of rows Excel must count down to find the matching value.

For example, you might have a lookup table consisting of two rows, with the total order amount in the top row and the corresponding shipping charge in the second row. To look up the shipping charge for a specified order amount (lookup value), you would enter a row index value of 2; Excel would then look for the lookup value in the top row of the lookup table and return the value in the second row of the same column.

You might want to use the lookup table shown below and a row index number of 2 to look up the percent of commission to be paid to a salesperson, based on various sales levels. For example, if the sales figure you want to look up is 6000, the commission would be 6%; Excel finds the lookup value (6000) in the top row and returns the value in the second row of the same column.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1000</td>
<td>2000</td>
<td>3000</td>
<td>4000</td>
<td>5000</td>
<td>6000</td>
<td>7000</td>
<td>8000</td>
</tr>
<tr>
<td>2</td>
<td>1%</td>
<td>2%</td>
<td>3%</td>
<td>4%</td>
<td>5%</td>
<td>6%</td>
<td>7%</td>
<td>8%</td>
</tr>
</tbody>
</table>

The HLOOKUP function also has a optional fourth argument: range lookup. This can be either TRUE or FALSE. If the range lookup argument is FALSE, HLOOKUP will find only exact matches. If the range lookup argument is TRUE, or if a range lookup argument is not entered, HLOOKUP can find approximate matches. In this case, the lookup table must be sorted in ascending order by its top row; otherwise HLOOKUP may not return the correct value.
If the range lookup argument is TRUE or omitted and the lookup value does not appear in the top row of the lookup table, but falls between two values in it, Excel uses the lower of the two values. If the lookup value is smaller than any value in the top row of the lookup table, Excel returns an error message.

For example, using the lookup table shown above and a row index number of 2, if the sales figure you look up is 5700, the commission would be 5%. Since Excel determines that 5700 is located between the numbers 5000 and 6000, it returns the value in the second row of the same column as the lower number.

In order to copy a HLOOKUP function to other cells, its table array argument must be an absolute reference. Since named ranges are always absolute references, you can assign a name to your lookup table and use that name in the HLOOKUP function.

You can use the Options button in the Sort dialog box to sort a range by row.
Procedures

1. Select the cell in which you want the result of the HLOOKUP function to appear.
2. Type =hlookup and an open parenthesis ( ( ).
3. Select the cell containing the value you want to look up.
4. Type a comma ( , ).
5. Type the name or address of the lookup table.
6. Type a comma ( , ).
7. Enter the row index number.
8. Type a closing parenthesis ( ) ).
9. Press [Enter].

Step-by-Step

Use the HLOOKUP function.

If necessary, display the Orders worksheet.

Select shipping from the Name Box (the list box to the left of the formula bar) to view the lookup table.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell in which you want the result of the HLOOKUP function to appear. &lt;br&gt; <em>The cell is selected.</em></td>
<td>Click cell F5</td>
</tr>
<tr>
<td>2. Type =hlookup and an open parenthesis ( ( ). &lt;br&gt; <em>hlookup</em> appears in the cell and on the formula bar. As you start typing a function, a Screen Tip is displayed to help you enter valid arguments.</td>
<td>Type =hlookup(</td>
</tr>
<tr>
<td>3. Select the cell containing the value you want to look up. &lt;br&gt; <em>A blinking marquee appears around the cell and its address appears in the HLOOKUP function.</em></td>
<td>Click cell E5</td>
</tr>
</tbody>
</table>
### Steps

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. Type a comma ( , ).</td>
<td>Type ,</td>
</tr>
<tr>
<td>The comma ( , ) appears in the HLOOKUP function.</td>
<td></td>
</tr>
<tr>
<td>5. Type the name or address of the lookup table.</td>
<td>Type shipping</td>
</tr>
<tr>
<td>The text appears in the HLOOKUP function.</td>
<td></td>
</tr>
<tr>
<td>6. Type a comma ( , ).</td>
<td>Type ,</td>
</tr>
<tr>
<td>The comma ( , ) appears in the HLOOKUP function.</td>
<td></td>
</tr>
<tr>
<td>7. Enter the row index number.</td>
<td>Type 2</td>
</tr>
<tr>
<td>The row index number appears in the HLOOKUP function.</td>
<td></td>
</tr>
<tr>
<td>8. Type a closing parenthesis ( ) .</td>
<td>Type )</td>
</tr>
<tr>
<td>The closing parenthesis ( ) appears in the HLOOKUP function.</td>
<td></td>
</tr>
<tr>
<td>The result of the HLOOKUP function appears in the cell.</td>
<td></td>
</tr>
</tbody>
</table>

Notice that since the lookup value ($639.90) is between two values in the lookup table, the returned cost ($50.00) is the same as the cost for the lower amount ($400), in cell G18.

Copy the formula to the range F6:F13. Click anywhere in the worksheet to deselect the range.

---

**USING THE IF FUNCTION**

### Discussion

Logical functions make decisions based on criteria. If the criteria evaluate to true, one action is taken; if the criteria evaluate to false, a different action is taken.

This decision-making capability of logical functions can be applied to many different situations. You can use a logical function to decide if a customer receives a discount for goods ordered. If an ordered value is greater than the specified amount, the customer receives a discount. If an ordered value is less than the specified amount, the customer does not receive a discount.

The IF function returns one value if a condition is true and another value if a condition is false. In the example above, if the value of the goods shipped is greater than the
specified amount, a true value would be returned. If the shipped value is less than the specified amount, a false value would be returned.

You can also use the IF function to display text as a result of a logical test, but you must enclose the text you want to display in quotation marks.

The syntax of an IF function is:

\[
\text{=IF(logical test, value if true, value if false)}
\]

The function arguments are described in the following table:

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
</table>
| logical test| The test condition. It can contain cell references, text in quotes, cell names, and numbers. The items are compared using the following comparison operators:  
  
  = equal to  
  <> not equal to  
  > greater than  
  >= greater than or equal to  
  < less than  
  <= less than or equal to  

<table>
<thead>
<tr>
<th>value if true</th>
<th>The desired result if the logical test is true. It can be a number, formula, cell reference, cell name, text in quotes, or another function.</th>
</tr>
</thead>
<tbody>
<tr>
<td>value if false</td>
<td>The desired result if the logical test is false. It can be a number, formula, cell reference, cell name, text in quotes, or another function.</td>
</tr>
</tbody>
</table>

Some examples of the IF function are listed in the following table:

<table>
<thead>
<tr>
<th>IF Function</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>=IF(B7&gt;10,C7*.1,0)</td>
<td>If the number in cell B7 is greater than 10, multiply the number in cell C7 by .1; otherwise, return the number 0</td>
</tr>
<tr>
<td>=IF(B7&lt;=10,C7*.1,D7*.1)</td>
<td>If the number in cell B7 is less than or equal to 10, multiply the number in cell C7 by .1; otherwise, multiply the number in cell D7 by .1</td>
</tr>
<tr>
<td>=IF(B7&lt;&gt;10,&quot;GOOD&quot;,&quot;&quot;)</td>
<td>If the number in cell B7 is not equal to 10, enter the text GOOD in the current cell; otherwise, leave the cell blank</td>
</tr>
</tbody>
</table>
### IF Function

<table>
<thead>
<tr>
<th>IF Function</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>=IF(B7=&quot;BONUS&quot;,C7+1000,C7)</td>
<td>If cell B7 contains the text BONUS, add 1000 to the number in cell C7; otherwise, enter the contents of cell C7</td>
</tr>
</tbody>
</table>

**Creating an IF function**

### Procedures

1. Select the cell in which you want the result of the IF function to appear.
2. Type `=if` and an open parenthesis (`(`).
3. Type the logical test.
4. Type a comma (`,`).
5. Type the action to be taken if the logical test is true.
6. Type a comma (`,`).
7. Type the action to be taken if the logical test is false.
8. Type the closing parentheses (`)`.

---

Villanova UNIT Training®
Step-by-Step

Use the IF function.

Display the **Bonus** worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell in which you want the result of the IF function to appear. <em>The cell is selected.</em></td>
<td>Click cell G8</td>
</tr>
<tr>
<td>2. Type <code>=if</code> and an open parenthesis <code>(</code>. DUMMY text appears in the cell and on the formula bar. As you start typing a function, a Screen Tip is displayed to help you enter valid arguments.</td>
<td>Type <code>=if(</code></td>
</tr>
<tr>
<td>3. Type the logical test. <em>The text appears in the cell and on the formula bar.</em></td>
<td>Type <code>e8&gt;f8</code></td>
</tr>
<tr>
<td>4. Type a comma <code>,</code>. <em>The comma <code>,</code> appears in the cell and on the formula bar.</em></td>
<td>Type <code>,</code></td>
</tr>
<tr>
<td>5. Type the action to be taken if the logical test is true. <em>The text appears in the cell and on the formula bar.</em></td>
<td>Type <code>e8*10%</code></td>
</tr>
<tr>
<td>6. Type a comma <code>,</code>. <em>The comma <code>,</code> appears in the cell and on the formula bar.</em></td>
<td>Type <code>,</code></td>
</tr>
<tr>
<td>7. Type the action to be taken if the logical test is false. <em>The text appears in the cell and on the formula bar.</em></td>
<td>Type <code>0</code></td>
</tr>
<tr>
<td>8. Type the closing parenthesis <code>)</code>. <em>The closing parenthesis <code>)</code> appears in the cell and on the formula bar.</em></td>
<td>Type <code>)</code></td>
</tr>
</tbody>
</table>
Notice that since the first quarter sales total for Smith, S. was below his quota, a zero (0) was entered as his bonus.

Copy the formula to the range G9:G13. Then, click anywhere in the worksheet to deselect the range.

**USING NESTED IF FUNCTIONS**

**Discussion**

You can use an IF function within another IF function to create a nested IF function. A nested IF function allows you to test for a second condition if the first condition is found false. For example, an IF function could test whether or not a number is equal to 1. If false, another IF function within the first could test whether or not the number is equal to 2.

The syntax of a nested IF function is:

```
=IF(logical test,value if true,IF(logical test,value if true,value if false))
```

You can create formulas with up to 64 levels of nested functions.

![Creating a nested IF function](image)

⚠️ You must close all parentheses in a nested IF function; i.e., the number of opening parentheses must equal the number of closing parentheses.
Procedures

1. Select the cell in which you want the result of the nested IF function to appear.
2. Type =IF and an open parenthesis ( ( ).
3. Type the first logical test.
4. Type a comma ( , ).
5. Type the action to be taken if the first logical test is true.
6. Type a comma ( , ).
7. Type IF and an open parenthesis ( ( ).
8. Type the logical test for the second IF function.
9. Type a comma ( , ).
10. Type the action to be taken if the logical test for the second IF function is true.
11. Type a comma ( , ).
12. Type the action to be taken if the second logical test is false.
13. Type two closing parentheses ( ) ).
14. Press [Enter].

Step-by-Step

Use nested IF functions.

If necessary, display the Bonus worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell in which you want the result of the nested IF function to appear. The cell is selected.</td>
<td>Click cell J8</td>
</tr>
<tr>
<td>Steps</td>
<td>Practice Data</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>2. Type =if and an open parenthesis ( ( ). =if() appears in the cell</td>
<td>Type =\textit{if()} appears in the cell and on the formula bar, and a Screen Tip appears.</td>
</tr>
<tr>
<td>3. Type the first logical test. The text appears in the cell and on</td>
<td>Type (i_8=1)</td>
</tr>
<tr>
<td>4. Type a comma ( , ). The comma ( , ) appears in the cell and on</td>
<td>Type ,</td>
</tr>
<tr>
<td>5. Type the action to be taken if the first logical test is true.</td>
<td>Type (h_8*10%)</td>
</tr>
<tr>
<td>6. Type a comma ( , ). The comma ( , ) appears in the cell and on the</td>
<td>Type ,</td>
</tr>
<tr>
<td>7. Type \textit{if} and an open parenthesis ( ( ). \textit{if}()</td>
<td>Type \textit{if()} appears in the cell and on the formula bar.</td>
</tr>
<tr>
<td>8. Type the logical test for the second IF function. The text</td>
<td>Type (i_8=2)</td>
</tr>
<tr>
<td>9. Type a comma ( , ). The comma ( , ) appears in the cell and on the</td>
<td>Type ,</td>
</tr>
<tr>
<td>10. Type the action to be taken if the logical test for the second IF</td>
<td>Type (h_8*8%)</td>
</tr>
<tr>
<td>11. Type a comma ( , ). The comma ( , ) appears in the cell and on the</td>
<td>Type ,</td>
</tr>
<tr>
<td>12. Type the action to be taken if the second logical test is false.</td>
<td>Type (h_8*7%)</td>
</tr>
</tbody>
</table>
Steps | Practice Data
--- | ---
13. Type two closing parentheses ( ) ). *The closing parentheses ( ) ) appear in the cell and on the formula bar.* | Type )
14. Press [Enter]. *The result of the nested IF function appears in the cell.* | Press [Enter]

Copy the formula to the range J9:J13. Then, click anywhere in the worksheet to deselect the range.

**USING THE ISERROR FUNCTION**

![Discussion](image)

Depending upon the circumstances, a function may return an error message instead of performing the desired calculation. For instance, a function that averages a range will return a **#DIV/0!** error message if the range contains no data. The ISERROR function is commonly used within an IF function to handle errors messages returned by a formula.

The ISERROR function tests TRUE if any of the following error messages are returned by a formula: **#N/A, #VALUE, #REF, #DIV/0!, #NUM, #NAME?, or #NULL.** It tests FALSE if anything other than an error message is returned.

The ISERR function is similar to the ISERROR function, except that it does not respond to the error value **#N/A.**

The syntax of these functions is as follows, where (value) is a cell reference or range name:

- ISERROR(value)
- ISERR(value)
Using the ISERROR function

If you are unsure of the contents of one or more cells on which the calculations are being made, you can use the ISERROR function because it provides a result regardless of the error condition.

Procedures

1. Select the cell in which you want the result of the IF function to appear.
2. Type `=if` and an open parenthesis ( ( ).
3. Type the ISERROR function as the logical test.
4. Type a comma ( , ).
5. Type the action to be taken if the ISERROR function is true.
6. Type a comma ( , ).
7. Type the action to be taken if the ISERROR function is false.
8. Type the closing parenthesis ( ) )
9. Press [Enter].
Step-by-Step

Use the ISERROR function in an IF function.

Display the Quota worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell in which you want the result of the IF function to appear. The cell is selected.</td>
<td>Click cell F15</td>
</tr>
<tr>
<td>2. Type =if and an open parenthesis ( ( .). =if() appears in the cell and on the formula bar, and a Screen Tip appears.</td>
<td>Type =if()</td>
</tr>
<tr>
<td>3. Type the ISERROR function as the logical test. The text appears in the cell and on the formula bar.</td>
<td>Type iserror(average(f8:f13))</td>
</tr>
<tr>
<td>4. Type a comma ( , ). The comma ( , ) appears in the cell and on the formula bar.</td>
<td>Type ,</td>
</tr>
<tr>
<td>5. Type the action to be taken if the ISERROR function is true. The text appears in the cell and on the formula bar.</td>
<td>Type &quot;no data&quot;</td>
</tr>
<tr>
<td>6. Type a comma ( , ). The comma ( , ) appears in the cell and on the formula bar.</td>
<td>Type ,</td>
</tr>
<tr>
<td>7. Type the action to be taken if the ISERROR function is false. The text appears in the cell and on the formula bar.</td>
<td>Type average(f8:f13)</td>
</tr>
<tr>
<td>8. Type the closing parenthesis ( ) . The closing parenthesis ( ) appears in the cell and on the formula bar.</td>
<td>Type )</td>
</tr>
<tr>
<td>9. Press [Enter]. The result of the ISERROR function appears in the cell.</td>
<td>Press [Enter]</td>
</tr>
</tbody>
</table>

The formula returns the text no data instead of the error message. Copy the formula to the range G15:I15. Then, type 50,000 in cell F8 and press [Enter]; notice that the result in cell F15 changes.
Using an AND Condition with IF

Discussion

You can use AND conditions to test multiple criteria in IF functions. For example, you may want to give a salesperson a $500 bonus if he or she produces $10,000 in sales and has at least five years experience. This example represents an AND condition. When used in an IF function, an AND condition returns a TRUE value if both arguments are true and a FALSE value if either argument is false.

The syntax of an AND condition is:

=IF(AND(logical test1,logical test2),value if true,value if false)

Creating an AND condition in an IF function

Procedures

1. Select the cell in which you want the result of the IF function to appear.
2. Type =if and an open parenthesis ( ().
3. Type the AND condition.
4. Type a comma (,).
5. Type the action to be taken if both conditions are true.
6. Type a comma ( , ).
7. Type the action to be taken if either condition is false.
8. Type a closing parenthesis ( )).
9. Press [Enter].

Step-by-Step

Use an AND condition in an IF function.

Display the Raises worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell in which you want the result of the IF function to appear. The cell is selected.</td>
<td>Click cell D8</td>
</tr>
<tr>
<td>2. Type =if and an open parenthesis ( ( ). =if( appears in the cell and on the formula bar, and a Screen Tip appears.</td>
<td>Type =if(</td>
</tr>
<tr>
<td>3. Type the AND condition. The text appears in the cell and on the formula bar.</td>
<td>Type and(b8&gt;0,c8&gt;1)</td>
</tr>
<tr>
<td>4. Type a comma ( , ). The comma ( , ) appears in the cell and on the formula bar.</td>
<td>Type ,</td>
</tr>
<tr>
<td>5. Type the action to be taken if both conditions are true. The text appears in the cell and on the formula bar.</td>
<td>Type &quot;Yes&quot;</td>
</tr>
<tr>
<td>6. Type a comma ( , ). The comma ( , ) appears in the cell and on the formula bar.</td>
<td>Type ,</td>
</tr>
<tr>
<td>7. Type the action to be taken if either condition is false. The text appears in the cell and on the formula bar.</td>
<td>Type &quot;No&quot;</td>
</tr>
</tbody>
</table>
Steps | Practice Data
---|---
8. Type a closing parenthesis ( ). The closing parenthesis ( ) appears in the cell and on the formula bar. | Type )
9. Press [Enter]. The result of the IF function appears in the cell. | Press [Enter]

The correct answer is No, since only one condition is true.

Copy the formula to the range D9:D13. Then, click anywhere in the worksheet to deselect the range.

**USING AN OR CONDITION WITH IF**

**Discussion**

You can use **OR** conditions to test multiple criteria in IF functions. For example, you may want to give a salesperson a $500 bonus if she produces $10,000 in sales or if she has at least five years experience. This example represents an **OR** condition. When used in an IF function, the **OR** condition returns a TRUE value if either argument is true and a FALSE value if both arguments are false.

The syntax of an **OR** condition is:

```excel
=IF(OR(logical test1,logical test2),value if true,value if false)
```
Creating an OR condition in an IF function

Procedures

1. Select the cell in which you want the result of the IF function to appear.
2. Type =if and an open parenthesis ( ( ).
3. Type the OR condition.
4. Type a comma ( , ).
5. Type the action to be taken if either of the conditions is true.
6. Type a comma ( , ).
7. Type the action to be taken if both conditions are false.
8. Type the closing parenthesis ( ) ).
9. Press [Enter].

Step-by-Step

Use an OR condition in an IF function.

If necessary, display the Raises worksheet.
**Steps** | **Practice Data**
---|---
1. **Select the cell in which you want the result of the IF function to appear.**
The cell is selected. | Click cell E8
2. **Type =if and an open parenthesis ( ).**
=if() appears in the cell and on the formula bar, and a Screen Tip appears. | Type =if(
3. **Type the OR condition.**
The text appears in the cell and on the formula bar. | Type or(b8>10000,c8=3)
4. **Type a comma ( ).**
The comma (,) appears in the cell and on the formula bar. | Type ,
5. **Type the action to be taken if either of the conditions is true.**
The text appears in the cell and on the formula bar. | Type "Yes"
6. **Type a comma ( ).**
The comma (,) appears in the cell and on the formula bar. | Type ,
7. **Type the action to be taken if both conditions are false.**
The text appears in the cell and on the formula bar. | Type "No"
8. **Type the closing parenthesis ( )).**
The closing parenthesis ( ) appears in the cell and on the formula bar. | Type )
9. **Press [Enter].**
The result of the IF function with the OR condition appears in the cell. | Press [Enter]

The correct answer is **No**, since neither condition is true.

Copy the formula to the range E9:E13. Then, click anywhere in the worksheet to deselect the range.
USING THE ROUND FUNCTION

Discussion

When you enter a number into an Excel worksheet, Excel can store it with up to 15 digits. Although you can format numbers so that Excel rounds off extra decimal places, Excel uses all decimal places in calculations. This feature can lead to some calculations appearing incorrect.

Rounding a number is different than formatting a number. When you round a number to a certain number of decimal places, the extra decimal places are removed and all calculations are performed using the rounded value.

The ROUND function includes the following two arguments:

ROUND(number,number of digits)

The number argument can be a value or a cell address. The number of digits argument determines the precision of the rounded number. A positive number of digits argument returns an equal number of decimal places. If the number of digits argument is 0, Excel rounds to the next whole number. A negative number of digits argument rounds exponentially to the next ten, hundred, thousand, etc.

Some examples of the ROUND function are listed in the following table:

<table>
<thead>
<tr>
<th>ROUND function</th>
<th>Cell displays</th>
</tr>
</thead>
<tbody>
<tr>
<td>=ROUND(4567.4567,1)</td>
<td>4567.5 (one decimal place)</td>
</tr>
<tr>
<td>=ROUND(4567.4567,2)</td>
<td>4567.46 (two decimal places)</td>
</tr>
<tr>
<td>=ROUND(4567.4567,0)</td>
<td>4567 (no decimal places)</td>
</tr>
<tr>
<td>=ROUND(4567.4567,-1)</td>
<td>4570 (rounds to the nearest ten)</td>
</tr>
<tr>
<td>=ROUND(B7,2)</td>
<td>The value in cell B7 rounded to two decimal places</td>
</tr>
<tr>
<td>=ROUND(B7*.1,2)</td>
<td>The result of the number in cell B7 times .1, rounded to two decimal places</td>
</tr>
</tbody>
</table>
Creating a ROUNDD function

Procedures

1. Select the cell in which you want the result of the ROUNDD function to appear.

2. Type =round and an open parenthesis ( ( ).

3. Type the value, formula, cell address, or function you want to round, followed by a comma.

4. Type the desired number of decimal places.

5. Type the closing parenthesis ( )).

6. Press [Enter].

Step-by-Step

Use the ROUNDD function.

Display the Invest worksheet.
<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell in which you want the result of the ROUND function to appear. The cell is selected.</td>
<td>Click cell F7</td>
</tr>
<tr>
<td>2. Type =round and an open parenthesis ( ). =round appears in the cell and on the formula bar.</td>
<td>Type =round(</td>
</tr>
<tr>
<td>3. Type the value, formula, cell address, or function you want to round, followed by a comma. The text appears in the cell and on the formula bar.</td>
<td>Type d7,</td>
</tr>
<tr>
<td>4. Type the desired number of decimal places. The text appears in the cell and on the formula bar.</td>
<td>Type 4</td>
</tr>
<tr>
<td>5. Type the closing parenthesis ( ). The closing parenthesis ( ) appears in the cell and on the formula bar.</td>
<td>Type )</td>
</tr>
<tr>
<td>6. Press [Enter]. The result of the ROUND function appears in the cell.</td>
<td>Press [Enter]</td>
</tr>
</tbody>
</table>

Notice that the number in cell G7 differs from that in cell E7 because it is based on the rounded number in cell F7. Then, copy the formula to the range F8:F12.

Select the range D7:D12 and click the Decrease Decimal button in the Number group on the Home tab twice to format the numbers for four decimal places. Notice that the values in column E do not change; the calculations are still based on the full number of decimal places. Use the Undo feature twice to remove the formatting.

**Limiting the Precision of Numbers**

**Discussion**

In order to calculate a worksheet using the numbers as they are displayed, you can limit the precision of formatted numbers. Limiting the precision changes the actual values in the worksheet to their formatted versions. For example, if a cell containing an actual value of 123.4567 is formatted with no decimal places, only 123 will display. In a calculation, however, Excel will still use 123.4567 (the entire number, including all decimal places). If you limit the precision of the cell to the formatted
value, Excel will use only the formatted value (123) in calculations and will actually remove all decimal places in the stored number.

![Excel Options dialog box](image)

*Limiting the precision of numbers*

Be careful when limiting the precision of numbers because you cannot undo it. You can, however, restore your original numbers if you immediately exit the worksheet without saving the changes.

![Warning message](image)

### Procedures

1. Select the Office button.
2. Select Excel Options.
3. Select the Advanced page.
4. Under the When calculating this workbook section, select the Set precision as displayed option.
5. Select OK.
6. Select OK.
Step-by-Step

Limit the precision of numbers.

If necessary, display the Invest worksheet.

Select cell C7. Notice that, although the salary displayed on the worksheet is 4975, the actual value in the cell is 4974.554. All the salaries in column C have three or more decimal places, even though the decimal places do not appear due to the cell formatting.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Office button.</td>
<td>Click</td>
</tr>
<tr>
<td>The Office menu opens.</td>
<td></td>
</tr>
<tr>
<td>2. Select the Excel Options button.</td>
<td>Click Excel Options</td>
</tr>
<tr>
<td>The Excel Options dialog box opens.</td>
<td></td>
</tr>
<tr>
<td>3. Select the Advanced page.</td>
<td>Click Advanced</td>
</tr>
<tr>
<td>The Advanced page is displayed.</td>
<td></td>
</tr>
<tr>
<td>4. Under the When calculating this workbook section, select the Set</td>
<td>Click Set precision as displayed</td>
</tr>
<tr>
<td>precision as displayed option.</td>
<td></td>
</tr>
<tr>
<td>The option is selected and a Microsoft Office Excel warning box</td>
<td></td>
</tr>
<tr>
<td>opens to inform you that the data will permanently lose accuracy.</td>
<td></td>
</tr>
<tr>
<td>5. Select OK.</td>
<td>Click OK</td>
</tr>
<tr>
<td>The Microsoft Office Excel warning box closes, and the numbers in</td>
<td></td>
</tr>
<tr>
<td>the worksheet are recalculated using limited precision.</td>
<td></td>
</tr>
<tr>
<td>6. Select OK.</td>
<td>Click OK</td>
</tr>
<tr>
<td>The Excel Options dialog box closes.</td>
<td></td>
</tr>
</tbody>
</table>

Select each of the cells in column C, one at a time. Notice that the decimal places have been removed. Close SALES72.XLSX.
EXERCISE

USING ADVANCED FUNCTIONS

Task

Use advanced functions in a workbook.

1. Open FUNCTEX.XLSX.
2. Display the Hours worksheet, if necessary.
3. Select cell F6 and use the VLOOKUP function to calculate the pay rate for full-time, part-time, or overtime work, according to job type. *(Hint: The lookup value is the Job Type in C6, the lookup table is named table, and the column index number is the Type of Time in E6.)*
4. Copy the formula to the range F7:F12.
5. Display the Sales-Previous worksheet.
6. Customers get free shipping if their average sales are more than 20,000. In cell I5, use an IF function to test the average sales. If the average sales (H5) are greater than or equal to 20,000, **Yes** should appear in the cell; otherwise, **No** should appear in the cell. *(Hint: Do not include a comma in 20000.)*
7. Copy the formula to the range I6:I13.
8. The customer discount depends on the customer status. In cell J5, use a nested IF function to determine the correct discount for each customer according to the following table. *(Hint: Test to see if the status in cell B5 is a 1, 2, or 3; otherwise, a discount of 10% is entered. Enter 5% as 0.05.)*

<table>
<thead>
<tr>
<th>Status</th>
<th>Discount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0%</td>
</tr>
<tr>
<td>2</td>
<td>5%</td>
</tr>
<tr>
<td>3</td>
<td>7.5%</td>
</tr>
<tr>
<td>4</td>
<td>10%</td>
</tr>
</tbody>
</table>
9. Copy the formula to the range J6:J13.

10. A customer status of 2 or 4 indicates a long-standing, good credit history and the customer credit limit should be raised by 5%; all other credit limits will remain the same. In cell K5, use an OR condition in an IF function to display which customers should have their credit limits raised. Display 5% or 0% in the cells accordingly.


12. Display the Sales-Current worksheet.

13. In cell C16, use an ISERROR function in an IF function to change the #DIV/0! message to No data. Copy the formula to the range D16:F16.

14. Display the Commissions worksheet.

15. Use the ROUND function to round the results of the commission formula in cell C5 to two decimal places. (Hint: Modify the existing formula in C5.) Then, copy the formula to the range C6:C13.

16. Format the numbers in the range B5:B13 so that no decimal places appear.

17. Limit the precision of numbers. Notice that the values in C5:C13 change accordingly.

18. Close the workbook without saving it.
LESSON 3 -
CREATING/REVISING PIVOTTABLES

In this lesson, you will learn how to:

- Create a PivotTable report
- Add PivotTable report fields
- Select a report filter field item
- Refresh a PivotTable report
- Change the summary function
- Add new fields to a PivotTable report
- Move PivotTable report fields
- Use Expand and Collapse buttons
- Hide/Unhide PivotTable report items
- Delete PivotTable report fields
- Create report filter pages
- Format a PivotTable report
- Create a PivotChart report
- Publish PivotTable reports to the Web
Creating a PivotTable Report

Discussion

The PivotTable is a powerful tool for summarizing, analyzing, exploring and presenting data, extracted from larger amounts of data elsewhere in your workbook. It enables you to query the data in user-friendly ways, subtotal and aggregate numeric data, summarize data by categories and subcategories, and create custom calculations and formulas. You can expand and collapse levels of data to focus your results, and drill down to details from the summary data for areas of interest to you. You can also move rows to columns or columns to rows (or “pivot”) to see different summaries of the source data.

In Excel 2007, PivotTables are much easier to use than in earlier versions of Excel. By using the new PivotTable user interface, there are fewer steps in the process to create a clear layout based on your data and changing the layout of a PivotTable is also much easier.

After you have created a PivotTable, you can take advantage of many other new or improved features to summarize, analyze and format your PivotTable data. You can quickly apply a predefined or custom style to a PivotTable and sorting and filtering your data is now simple. You can filter data by using date filters, label filters, value filters or manual filters. For greater flexibility, you can also undo most actions that you carry out to create or rearrange a PivotTable.
Procedures

1. Select any cell in the database.
2. Select the Insert tab.
3. Select the top part of the PivotTable button in the Tables group.
4. Select where you want the PivotTable report to appear.
5. Select OK.

Step-by-Step

From the Student Data directory, open PIVOT1.XLSX. Creating a PivotTable report.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select a cell containing data in the worksheet. The cell is selected.</td>
<td>Click A4</td>
</tr>
<tr>
<td>2. Select the Insert tab. The Insert tab is displayed.</td>
<td>Click Insert</td>
</tr>
<tr>
<td>3. Select the top part of the PivotTable button in the Tables group. The Create PivotTable dialog box opens and the data range is selected on the worksheet.</td>
<td>Click</td>
</tr>
<tr>
<td>4. Select where you want the PivotTable report to appear. The option is selected.</td>
<td>Click ☑ New worksheet, if necessary</td>
</tr>
<tr>
<td>5. Select OK. The Create PivotTable dialog box closes. A new worksheet is displayed and the PivotTable Field List pane opens. The PivotTable Tools contextual Options tab is displayed on the Ribbon.</td>
<td>Click OK</td>
</tr>
</tbody>
</table>
**ADDING PIVOT TABLE REPORT FIELDS**

**Discussion**

After you create a PivotTable, you use the *PivotTable Field List* to add, rearrange and remove fields. The *PivotTable Field List* pane displays two sections: a field section at the top for adding and removing fields and a layout section at the bottom for rearranging and repositioning selected fields.

It is important to understand how the *PivotTable Field List* pane works and the ways that you can arrange different types of fields so that you can achieve the results that you want.

There are two primary ways to move field names into the layout sections in the *PivotTable Field List*. If you select the check box next to each field name in the field section, the field is placed in a default area of the layout section. However, you can rearrange the fields if you wish.

You can also choose the section to which you wish to add a field by right-clicking on the field name. A shortcut menu is displayed listing the specific areas of the layout section; the field is added by simply clicking on the desired layout section.

*A completed PivotTable report*
For greater ease of use, you can move the PivotTable Field List pane to either side of the window. You can also undock the PivotTable Field List and resize it both vertically and horizontally to suit your needs. You can also right-click a field name and choose its location from the shortcut menu.

In previous versions of Excel, you could also drag field names from the Field List directly onto the appropriate areas of the PivotTable report worksheet. This option is available in Excel 2007 by clicking on the Options button in the PivotTable group on the PivotTable Tools Options tab. The PivotTable Options dialog box opens. Click on the Display tab in the dialog box, and select the Classic PivotTable layout option.

Procedures

1. Select the desired field name from the field section in the PivotTable Field List pane and drag to the Row Labels box in the layout section.
2. Release the mouse button.
3. If necessary, select the next desired field name from the field section in the PivotTable Field List pane and drag to the Row Labels box in the layout section.
4. Release the mouse button.
5. Select the desired field name from the field section in the PivotTable Field List pane and drag to the Column Labels box in the layout section.
6. Release the mouse button.
7. Select the desired field name from the field section in the PivotTable Field List pane and drag to the Values box in the layout section.
8. Release the mouse button.

Step-by-Step

Adding fields to a PivotTable report.
<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the desired field name from the field section in the **PivotTable Field List** pane and drag to the **Row Labels** box in the layout section.  
The selected field name is highlighted and the mouse pointer changes to a move handle. The field name label appears as you drag to the desired location in the layout section. | Drag the **Purchaser** field to the **Row Labels** box |
| 2. Release the mouse button.  
The selected field name is displayed in the **Row Labels** box in the layout section, and a list of the field’s data values is displayed in the PivotTable report worksheet. | Release the mouse button |
| 3. If necessary, select the next desired field name from the field section in the **PivotTable Field List** pane and drag to the **Row Labels** box in the layout section.  
The selected field name is highlighted and the mouse pointer changes to a move handle. The field name label appears as you drag to the desired location in the layout section. | Drag the **Product** field to the **Row Labels** box |
| 4. Release the mouse button.  
The selected field name is displayed in the **Row Labels** box in the layout section, and a list of the field’s data values is displayed in the PivotTable report worksheet. | Release the mouse button |
| 5. Select the desired field name from the field section in the **PivotTable Field List** pane and drag to the **Column Labels** box in the layout section.  
The selected field name is highlighted and the mouse pointer changes to a move handle. The field name label appears as you drag to the desired location in the layout section. | Drag the **Month** field to the **Column Labels** box |
Steps | Practice Data
--- | ---
6. Release the mouse button. The selected field name is displayed in the Column Labels box in the layout section, and column labels are displayed, along with a Grand Total column and row, in the PivotTable report worksheet. | Release the mouse button

7. Select the desired field name from the field section in the PivotTable Field List pane and drag to the Values box in the layout section. The selected field name is highlighted and the mouse pointer changes to a move handle. The field name label appears as you drag to the desired location in the layout section. | Drag the Sales field to the Values box

8. Release the mouse button. The selected field name is displayed in the Values box in the layout section. The field’s data values, along with Grand Total values, are displayed in the PivotTable report worksheet. | Release the mouse button

SELECTING A REPORT FILTER FIELD ITEM

Discussion

Report filters allow you to display a subset of data in a PivotTable report by displaying only the data for a specific item; you can drag any field to the Report Filter box. The report filter list then displays all the values of the selected field.

When you select an item from the report filter list, only the data relating to that single item appears in the PivotTable report. For example, if items in the Region page field include Northeast, Northwest, Southeast, etc., and you select the Northeast item, only the data for the Northeast region appears in the PivotTable report.
Procedures

1. Select the desired field name from the field section in the **PivotTable Field List** pane and drag to the **Report Filter** box in the layout section.
2. Release the mouse button.
3. Click the report filter field list.
4. Select the desired item.
5. Select **OK**.

Step-by-Step

Select a report filter field item.

If necessary, display the **Sheet1** worksheet.
### Steps

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the desired field name from the field section in the <strong>PivotTable Field List</strong> pane and drag to the <strong>Report Filter</strong> box in the layout section. <em>The selected field name is highlighted and the mouse pointer changes to a move handle. The field name label appears as you drag to the desired location in the layout section.</em></td>
<td>Drag the <strong>SalesRep</strong> field to the <strong>Report Filter</strong> box</td>
</tr>
<tr>
<td>2. Release the mouse button. <em>The selected field name is displayed in the <strong>Report Filter</strong> box in the layout section, and the field name is displayed in the <strong>PivotTable</strong> report worksheet.</em></td>
<td>Release the mouse button</td>
</tr>
<tr>
<td>3. Click the report filter field list. <em>A list of available field values is displayed.</em></td>
<td>Click ![Dropdown Icon] in cell B1</td>
</tr>
<tr>
<td>4. Select the desired item. <em>The item is selected.</em></td>
<td>Click <strong>Janice Faraco</strong></td>
</tr>
<tr>
<td>5. Select <strong>OK</strong>. <em>Only the data for the selected item appears in the PivotTable report.</em></td>
<td>Click ![OK Button]</td>
</tr>
</tbody>
</table>

Notice that only the data for the selected sales representative **Janice Faraco** is displayed.

---

### Refreshing a PivotTable Report

#### Discussion

If the source data changes, a PivotTable report does not automatically update. If you change the existing data, delete records or fields, or add records or fields to the original database range, you need to refresh the PivotTable report.

If you add records below the last row or add fields to the right of the last column in a database, you will have to open the PivotTable and PivotChart Wizard to expand the data source range to include the new rows and/or columns.
Procedures

1. Right-click any cell in a PivotTable report to display a shortcut menu.
2. Select the Refresh command.

Step-by-Step

Refresh a PivotTable report.

If necessary, display the Sheet1 worksheet and select Janice Faraco from the SalesRep report filter field.

Display the Sales worksheet. Change the number in cell D10 from 1089 to 2000. Then, display the Sheet1 worksheet again. Notice that the number 1089 still appears in the Gloves field for March.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Right-click any cell in a PivotTable report to display a shortcut menu. The PivotTable shortcut menu opens.</td>
<td>Right-click cell A3</td>
</tr>
<tr>
<td>2. Select the Refresh command. The PivotTable report updates.</td>
<td>Click Refresh</td>
</tr>
</tbody>
</table>

Display the data for all the sales representatives by selecting (All) for the SalesRep report filter field. Notice the change to the data displayed.

Changing the Summary Function

Discussion

By default, an Excel PivotTable report sums fields that contain numeric data and counts fields that contain text. The PivotTable report can provide summary information using other mathematical calculations as well. A PivotTable report can average data, provide the minimum and maximum values, or yield the standard deviation or variance. A PivotTable report can also contain multiple summaries for the same field.
The Value Field Settings dialog box

When a PivotTable report contains multiple data fields, you can change the summary function for a field by selecting any cell in the row of the data field you want to change.

Procedures

1. Right-click any cell in a PivotTable report to display a shortcut menu.
2. Select the Value Field Settings command.
3. Select the desired summary function from the Summarize value field by list box.
4. Select OK.

Step-by-Step

Change the summary function for a data field.

If necessary, display the Sheet1 worksheet.
### Steps

<table>
<thead>
<tr>
<th></th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Right-click any cell in a PivotTable report to display a shortcut menu. <em>The PivotTable shortcut menu opens.</em></td>
</tr>
<tr>
<td>2.</td>
<td>Select the <strong>Value Field Settings</strong> command. <em>The Value Field Settings dialog box opens.</em></td>
</tr>
<tr>
<td>3.</td>
<td>Select the desired summary function from the <strong>Summarize value field by</strong> list box. <em>The summary function is selected.</em></td>
</tr>
<tr>
<td>4.</td>
<td>Select <strong>OK</strong>. <em>The Value Field Settings dialog box closes, and the PivotTable report is summarized accordingly.</em></td>
</tr>
</tbody>
</table>

**Practice the Concept:** Reset the summary function for the **Sales** field to **Sum**.

### Adding New Fields to a PivotTable Report

**Discussion**

PivotTable reports can display multiple fields in the row, column, report filter, or data areas. Multiple fields can add more detail to your PivotTable report. If a PivotTable report contains a **Month** column field and a **Purchaser** row field, you can add a **Product** row field to be able to display monthly sales by customer as well.

New fields can be added to a PivotTable report by dragging them from the field list at the top of the **PivotTable Field List** pane to the desired layout area at the bottom of the pane. The new field can be positioned before or after any existing fields. The position of the field determines how the data will be summarized.

---

In previous versions of Excel, you could also drag field names from the **Field List** directly onto the appropriate areas of the **PivotTable** report worksheet. This option is available in Excel 2007 by clicking on the **Options** button in the **PivotTable** group on the **PivotTable Tools Options** tab. The PivotTable Options dialog box opens. Click on the **Display** tab in the dialog box, and select the **Classic PivotTable layout** option.
If you choose to use the Classic PivotTable layout feature, as you drag new fields into a PivotTable report, a graphic will indicate the position of the field in relation to existing fields. When positioning a field in the row area, the graphic shows a blue line on its left hand side and there is also a thick, gray vertical bar which appears to the left or right of an existing row field. When positioning a field in the column area, the graphic shows a blue line on its top side with a thick, gray vertical bar which appears above or below an existing row field.

You can drag a field to a different position in the PivotTable report at any time; this feature is helpful if you accidentally position a field in the wrong area.

Procedures

1. Select any cell in the PivotTable report to activate it and open the PivotTable Field List pane.

2. Drag the field you want to add to the PivotTable report from the top section of the PivotTable Field List pane to the desired section in the layout area at the bottom of the pane.

Step-by-Step

Add a field to a PivotTable report.

If necessary, display the Sheet1 worksheet and click in the PivotTable report to open the PivotTable Field List pane.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Drag the field you want to add to the PivotTable report from the top section of the PivotTable Field List pane to the desired section in the layout area at the bottom of the pane. <em>The field is added to the appropriate section of the PivotTable Field List pane, and to the PivotTable report.</em></td>
<td>Drag the Region field to a position below the Month field in the Column Labels section</td>
</tr>
</tbody>
</table>
You can now easily see which customers made purchases in each region in each month, as well as which products were purchased.

**Practice the Concept:** Add the *Year* field to the report filter area by dragging the *Year* field below the *SalesRep* field.

Use the report filter field lists to display the 2005 sales for *Alice Abramam*. Then, display her 2006 sales. Remove the filters by selecting *(All)* for both report filter fields. Finally, remove the *Region* field from the report by dragging it from the *Column Labels* section into the area at the top of the *PivotTable Field List* pane.

---

**MOVING PIVOT TABLE REPORT FIELDS**

**Discussion**

After you create a PivotTable, you use the *PivotTable Field List* pane to add, rearrange and remove fields. The *PivotTable Field List* displays two sections: a field section at the top for adding and removing fields and a layout section at the bottom for rearranging and repositioning selected fields.

There are three primary ways to move field names into the layout sections in the *PivotTable Field List*. By selecting the check box next to each field name in the field section, the field is placed in a default area of the layout section. However, you can rearrange the fields if you wish.

You can also choose the section to which you wish to add a field by right-clicking on the field name. A shortcut menu is displayed listing the specific areas of the layout section; the field is added by simply clicking on the desired layout section.

You can change the layout of the PivotTable report by moving fields from one area to another. By moving the fields, you can experiment with the best way to display your data.

As you move fields in the PivotTable report, thick, gray bars indicate the placement of the field. Vertical bars represent row area placement and horizontal bars represent column and page area placement.

---

For greater ease of use, you can move the *PivotTable Field List* pane to either side of the window. You can also undock the *PivotTable Field List* pane and resize it both vertically and horizontally to suit your needs. You can also right-click a field name and choose its location from the shortcut menu.
In previous versions of Excel, you could also drag field names from the Field List directly onto the appropriate areas of the PivotTable report worksheet. This option is available in Excel 2007 by clicking on the Options button in the PivotTable group on the PivotTable Tools Options tab. The PivotTable Options dialog box opens. Click on the Display tab in the dialog box, and select the Classic PivotTable layout option.

You can also move a field from one area to another by dragging the desired field from the PivotTable Field List pane to the desired area; the field is removed from its previous area and appears only in the destination area.

Procedures

1. Select the desired field name from the relevant area in the lower half of the PivotTable Field List pane and drag to the required area.

2. Release the mouse button.

Step-by-Step

Move a PivotTable report field.

If necessary, display the Sheet1 worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the desired field name from the relevant area in the lower half of the PivotTable Field List pane and drag to the required area. The selected field name is highlighted and the mouse pointer changes to a move handle. The field name label appears as you drag to the desired area.</td>
<td>Drag the Purchaser field to a position below the Product field in the Row Labels box</td>
</tr>
</tbody>
</table>
Steps | Practice Data
---|---
2. Release the mouse button. *The desired field is displayed in the required target area in the layout section, and the PivotTable report is updated appropriately on the worksheet.* | Release the mouse button

Notice that the **Purchaser** field now appears to the right of the **Product** field and that the columns are now grouped by the **Purchaser** field as well as by the **Product** field. This layout makes it easier to view what products were purchased by each customer.

**Practice the Concept:** Drag the **Purchaser** field above the **Product** field to arrange the data first by purchaser and then by product.

---

### USING EXPAND AND COLLAPSE BUTTONS

#### Discussion

You can use the **Expand** and **Collapse** buttons on **PivotTable** reports to change the range of data you see on the report worksheet. These buttons are simple and easy to use and particularly helpful if your **PivotTable** report contains a large range of information. The **Collapse** button is a minus sign in a small square to the left of the row or column heading, and the **Expand** button is a plus sign.
Procedures

1. Select the **Collapse** button to the left of the desired field value to collapse the report details.

2. Select the **Expand** button to the left of the desired field value to show the report details.

Step-by-Step

Using the **Expand** and **Collapse** buttons.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>Collapse</strong> button to the left of the desired field value to collapse the report details. <em>The list of information is collapsed as appropriate.</em></td>
<td>Click the <strong>Collapse</strong> button to the left of <strong>Athlete’s Dream</strong></td>
</tr>
<tr>
<td>2. Select the <strong>Expand</strong> button to the left of the desired field value to show the report details. <em>The list of information is expanded as appropriate.</em></td>
<td>Click the <strong>Expand</strong> button to the left of <strong>Athlete’s Dream</strong></td>
</tr>
</tbody>
</table>

HIDING/UNHIDING PIVOT TABLE REPORT ITEMS

Discussion

You do not have to display all the field items in a PivotTable report. You can hide selected items, such as some of the products in a **Product** field. You can hide everything except the data you need to view.

You can also unhide items at any time. To unhide items, you follow the same procedure as you do when hiding items; you simply select the items you want to view in the field item list or select **(Select All)** to view all the field items.
Hiding field items

Procedures

1. Select the **Row Labels** or **Column Labels** list containing the items you want to hide or unhide.

2. Select the **Select field** list, if necessary.

3. Select the field containing the items you want to hide or unhide.

4. Deselect the item you want to hide, or select the item you want to unhide.

5. Continue deselecting or selecting items as desired.

6. Select **OK**.

Step-by-Step

Hide and unhide items in a PivotTable report.

If necessary, display the **Sheet1** worksheet.
### Steps

<table>
<thead>
<tr>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1.</strong> Select the Row Labels or Column Labels list containing the items you want to hide or unhide. The Select Field dialog box opens.</td>
</tr>
<tr>
<td><strong>2.</strong> Select the Select field list, if necessary. The list of available fields is displayed.</td>
</tr>
<tr>
<td><strong>3.</strong> Select the field containing the items you want to hide or unhide. The list of available items is displayed in the lower pane.</td>
</tr>
<tr>
<td><strong>4.</strong> Deselect the item you want to hide, or select the item you want to unhide. The item is deselected or selected.</td>
</tr>
<tr>
<td><strong>5.</strong> Continue deselecting or selecting items as desired. The items are deselected or selected.</td>
</tr>
<tr>
<td><strong>6.</strong> Select OK. The Select Field dialog box closes, and the fields are hidden or unhidden in the PivotTable report.</td>
</tr>
</tbody>
</table>

Notice that Baseballs and Gloves no longer appear in the PivotTable report.

**Practice the Concept:** Unhide the data in the Product item list by selecting (Select All).

---

### Deleting PivotTable Report Fields

**Discussion**

If you no longer want to view a field in your PivotTable report, you can remove it from the report. The report will update automatically as soon as you remove the field.

A field can also be removed by right-clicking on any relevant data in the PivotTable report, and selecting the Remove “[Field name]” command.
Procedures

1. In the **PivotTable Field List** pane, deselect the tick box of the field you want to remove.

Step-by-Step

Delete a PivotTable report field.

If necessary, display the **Sheet1** worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. In the <strong>PivotTable Field List</strong> pane, deselect the tick box of the field you want to remove. <strong>The field is removed from the PivotTable report.</strong></td>
<td>Click ☑️ <strong>SalesRep</strong> to deselect it</td>
</tr>
</tbody>
</table>

**Creating Report Filter Pages**

Discussion

You can use the Show Report Filter Pages feature to create a separate PivotTable report for each item in a report filter field. This feature creates a copy of the PivotTable report in a new worksheet for each report filter field item. For example, if you have a **Region** report filter field, and you select the **Show Report Filter Pages** feature, a separate worksheet is created for each region in the **Region** report filter field list.

Report filter pages are inserted before the original PivotTable report worksheet.
Viewing Report Filter Pages for all regions

You cannot undo the **Show Report Filter Pages** feature. To remove a report filter field report, you must delete the worksheet on which it appears.

You can double-click any data field item to display its details on a separate worksheet.

**Procedures**

1. Select the **PivotTable Tools** contextual **Options** tab.
2. Select the right-hand part of the **Options** button in the **PivotTable** group.
3. Select the **Show Report Filter Pages** command.
4. Select the report filter field for which you want to create report filter field reports.
5. Select **OK**.
Step-by-Step

Create Report Filter Pages.

If necessary, display the Sheet1 worksheet.

Drag the Region field from the PivotTable Field List pane to a position below the Year field in the Report Filter box.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the PivotTable Tools contextual Options tab. The Options tab is displayed.</td>
<td>Click Options</td>
</tr>
<tr>
<td>2. Select the right-hand part of the Options button in the PivotTable group. The Options menu opens.</td>
<td>Click Options</td>
</tr>
<tr>
<td>3. Select the Show Report Filter Pages command. The Show Report Filter Pages dialog box opens</td>
<td>Click Show Report Filter Pages</td>
</tr>
<tr>
<td>4. Select the report filter field for which you want to create report filter pages. The report filter field is selected.</td>
<td>Click Region</td>
</tr>
<tr>
<td>5. Select OK. The Show Report Filter Pages dialog box closes, and a report filter field report is created for each item in the report filter field.</td>
<td>Click <img src="image" alt="OK" /></td>
</tr>
</tbody>
</table>

View each of the report filter field reports.

Display the Sheet1 worksheet. Double-click the detail item in cell C7; a new worksheet is inserted to display the data item details in a table. Then, display the Sheet1 worksheet again and double-click the SportsCity Grand Total in cell M20; again, a new worksheet is inserted to display the data item details in a table. Close PIVOT1.XLSX.
FORMATTING A PIVOT TABLE REPORT

Discussion

You can use the Layout and the PivotTable Styles features to give a PivotTable report a more professional look. PivotTable Styles provides many formats, which include cell shading, font style and color, and borders.

There are three PivotTable report Layout formats. The default style of Compact indents inner fields, allowing you to see more information at glance. The two alternatives are Outline and Tabular.

You can also manually change the format of text and numbers using normal Excel formatting commands.

To switch from one layout to another, select the PivotTable Tools contextual Design tab, select the Report Layout button in the Layout group and then select the required layout.

If you create Report Filter Pages for a formatted PivotTable report, each new page will be formatted the same as the original PivotTable report.

Selecting Clear from the PivotTable Styles menu removes all formatting from a PivotTable report.
Procedures

1. Select the **PivotTable Tools** contextual **Design** tab.
2. Select the **More** button in the **PivotTable Styles** group.
3. Select the desired style.

Step-by-Step

From the Student Data directory, open PIVOT2.XLSX.
Format a PivotTable report using **PivotTable Styles**.

If necessary, display the **Pivot1** worksheet. Click anywhere in the PivotTable report to activate it.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>PivotTable Tools</strong> contextual <strong>Design</strong> tab.</td>
<td>Click <strong>Design</strong></td>
</tr>
<tr>
<td>The <strong>Design</strong> tab is displayed.</td>
<td></td>
</tr>
<tr>
<td>2. Select the <strong>More</strong> button in the <strong>PivotTable Styles</strong> group.</td>
<td>Click <strong>PivotTable Styles</strong></td>
</tr>
<tr>
<td>The <strong>PivotTable Styles</strong> gallery opens.</td>
<td></td>
</tr>
<tr>
<td>3. Select the desired style.</td>
<td>Scroll as necessary and click <strong>Pivot Style Light 19</strong></td>
</tr>
<tr>
<td>The <strong>PivotTable Styles</strong> gallery closes and the style is applied to the PivotTable report.</td>
<td></td>
</tr>
</tbody>
</table>

**CREATING A PIVOTCHART REPORT**

Discussion

A PivotChart report allows you to manipulate large amounts of data in a graphical environment.

PivotChart reports can be created from an Excel table, or from an existing PivotTable report. Both methods yield the same result: a chart with a layout you can change in order to view different levels of detail and data. When you create a PivotChart report...
from a table, Excel automatically creates an accompanying PivotTable report on a separate worksheet.

Creating a PivotChart report is similar to creating a PivotTable report. The chart layout is created by dragging fields from the **PivotTable Field List** pane to the desired area of the PivotChart report. The chart layout, however, is slightly different than the report layout.

![A PivotChart report](image)

### Procedures

1. Select any cell in the table.
2. Select the **Insert** tab.
3. Select the lower part of the **PivotTable** button in the **Tables** group.
4. Select the **PivotChart** command.
5. Select the required location of the PivotTable and PivotChart.
6. Select **OK**.
7. Drag the desired fields from the top half of the **PivotTable Field List** pane to the required area of the lower half of the pane.
8. Repeat step 7 as required for the fields to be shown on the PivotChart.

### Step-by-Step

Create a PivotChart report.

Display the Sales worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select any cell in the table. <em>The active cell moves accordingly.</em></td>
<td>Click cell A4</td>
</tr>
<tr>
<td>2. Select the <strong>Insert</strong> tab. <em>The Insert tab is displayed.</em></td>
<td>Click Insert</td>
</tr>
<tr>
<td>3. Select the lower half of the <strong>PivotTable</strong> button in the <strong>Tables</strong> group. <em>The PivotTable menu opens.</em></td>
<td>Click PivotTable</td>
</tr>
<tr>
<td>4. Select the <strong>PivotChart</strong> command. <em>The Create PivotTable With PivotChart dialog box opens.</em></td>
<td>Click PivotChart</td>
</tr>
<tr>
<td>5. Select the required location of the PivotTable and PivotChart. <em>The option is selected.</em></td>
<td>Click New Worksheet, if necessary</td>
</tr>
<tr>
<td>6. Select <strong>OK</strong>. <em>The Create PivotTable With PivotChart dialog box closes, and the PivotTable Field List pane and PivotChart Filter Pane are displayed.</em></td>
<td>Click OK</td>
</tr>
<tr>
<td>7. Drag the desired fields from the top half of the <strong>PivotTable Field List</strong> pane to the required area of the lower half of the pane. <em>The field appears in the required area.</em></td>
<td>Drag Month to the Axis Fields box</td>
</tr>
<tr>
<td>8. Repeat step 7 as required for the fields to be shown on the PivotChart. <em>The fields are shown in their respective areas in the lower section of the PivotTable Field List pane, and the PivotChart is created.</em></td>
<td>Follow the instructions shown below the table to complete the PivotChart.</td>
</tr>
</tbody>
</table>
Drag the **Sales** field to the **Values** box and the **Region** field to the **Legend Fields** box.

**PUBLISHING PIVOTTABLE REPORTS TO THE WEB**

**Discussion**

Excel 2007 allows you to publish PivotTable reports as web pages. This allows them to be viewed on a corporate Intranet or the Internet. Each time that you subsequently save the original workbook you can republish it by selecting the **Autorepublish every time this workbook is saved** option in the Publish as Web Page dialog box.

![The Publish as Web Page dialog box](image)

**Procedures**

1. Select the **Office** button .
2. Point to **Save As**.
3. Select the **Other Formats** option.
4. Type the desired file name in the **File Name** box.
5. Select the **Save as type** list.
6. Select the **Web Page** file type.
7. Select the drive where you want to save the HTML file.
8. Open the folder where you want to save the HTML file.
9. Select \(\text{Publish}\).
10. Select the **Open published web page in browser** option.

### Step-by-Step

Publish a PivotTable report to the Web.

Display the *Sheet1* worksheet, if necessary, and ensure that the PivotChart is positioned so that it does not overlay the PivotTable.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>Office</strong> button. <em>The Office menu opens.</em></td>
<td>Click</td>
</tr>
<tr>
<td>2. Point to <strong>Save As</strong>. <em>The Save As options are displayed.</em></td>
<td>Point to <strong>Save As</strong></td>
</tr>
<tr>
<td>3. Select the <strong>Other Formats</strong> option. <em>The Save As dialog box opens</em></td>
<td>Click <strong>Other Formats</strong></td>
</tr>
<tr>
<td>4. Type the desired file name in the <strong>File Name</strong> box. <em>The text appears in the File name box.</em></td>
<td>Type <code>pvtrpt2</code></td>
</tr>
<tr>
<td>5. Select the <strong>Save as type</strong> list. <em>A list of available file type options is displayed.</em></td>
<td>Click <strong>Save as type</strong></td>
</tr>
<tr>
<td>6. Select the <strong>Web Page</strong> file type. <em>The file type is selected.</em></td>
<td>Click <strong>Web Page</strong></td>
</tr>
<tr>
<td>7. Select the drive where you want to save the HTML file. <em>A list of available folders is displayed.</em></td>
<td>Click the student data drive</td>
</tr>
<tr>
<td>8. Open the folder where you want to save the HTML file. <em>The folder opens.</em></td>
<td>Double-click to open the student data folder</td>
</tr>
</tbody>
</table>
### Steps

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>9. Select the <strong>Publish</strong> button.</td>
<td>Click <img src="Illustration" alt="Publish..." /></td>
</tr>
<tr>
<td><em>The Save As dialog box closes, and the Publish as Web Page dialog box opens.</em></td>
<td></td>
</tr>
<tr>
<td>10. Select the <strong>Open published web page in browser</strong> option.</td>
<td>Click <img src="Illustration" alt="Open published web page in browser" /></td>
</tr>
<tr>
<td><em>The option is selected.</em></td>
<td></td>
</tr>
<tr>
<td>11. Select the <strong>Publish</strong> button.</td>
<td>Click <img src="Illustration" alt="Publish" /></td>
</tr>
<tr>
<td><em>The Publish as Web Page dialog box closes, and the PivotTable sheet opens as a HTML file in the default browser.</em></td>
<td></td>
</tr>
</tbody>
</table>

It may take a minute or two for the file to open in the browser. Maximize the browser window, if necessary. Close the browser window.
Close **PIVOT2.XLSX**.
EXERCISE

CREATING/REVISING PIVOT TABLES

Task

Create and revise PivotTable reports.

1. Open EXPIVOT1.XLSX and display the World worksheet, if necessary.

2. Create a PivotTable report from the data range A4:G67. Place the PivotTable report in a new worksheet.

3. Create the following layout:

<table>
<thead>
<tr>
<th>Field</th>
<th>Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date Sold</td>
<td>Report Filter</td>
</tr>
<tr>
<td>Product</td>
<td>Row Labels</td>
</tr>
<tr>
<td>Inv Num</td>
<td>Row Labels</td>
</tr>
<tr>
<td>Sales Rep</td>
<td>Column Labels</td>
</tr>
<tr>
<td>Total Income</td>
<td>Values</td>
</tr>
</tbody>
</table>

4. Use the Date Sold list to display only the sales for 02/06/2006.

5. Display the World worksheet and change the number in cell F9 to 25.

6. Display the Sheet1 worksheet and refresh the PivotTable report.

7. Change the Date Sold list to display all dates.

8. Move the Inv Num field to the Report Filter area.

9. Delete the Date Sold field from the Report Filter area.

10. Display the number of orders for each sales representative by changing the summary function for the data area to Count. (Hint: Display the Value Field Settings for the data area.) Then, return the summary function to Sum.

11. Add the Qty Sold field below the Product field in the Row Labels area.

13. Create a separate PivotTable report for each Inv Num item in the page area. (Hint: Use the Show Report Filter Pages option from the Options button in the PivotTable group on the Options tab.)

14. Select the 3325 sheet and apply the Pivot Style Dark 4 style.

15. Publish the 3325 worksheet to the student data folder as a web page named wsgpivot and have the published web page open in your browser.

16. Maximize your browser window, if necessary.

17. Close your browser window.

18. Create a PivotChart report from the World worksheet; base the PivotChart on the same data as the PivotTable on Sheet1 and place the PivotChart report in a new worksheet.

19. Create the following layout:

<table>
<thead>
<tr>
<th>Field</th>
<th>Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product</td>
<td>Axis Fields</td>
</tr>
<tr>
<td></td>
<td>(Category)</td>
</tr>
<tr>
<td>Total Income</td>
<td>Values</td>
</tr>
<tr>
<td>Sales Rep</td>
<td>Legend Fields</td>
</tr>
<tr>
<td></td>
<td>(Series)</td>
</tr>
<tr>
<td>Date Sold</td>
<td>Report Filter</td>
</tr>
</tbody>
</table>

20. Display only those products sold by John Carpenter. Then, display all sales representatives again.

21. Close the workbook without saving it.
LESSON 4 - USING SCENARIOS AND GOAL SEEKING

In this lesson, you will learn how to:

- Use the Scenario Manager
- Create a scenario
- Display a scenario
- Edit a scenario
- Create a scenario summary report
- Use Goal Seek
Using the Scenario Manager

Discussion

The ability to perform a what-if analysis is one of the most powerful functions Excel provides. A what-if analysis allows you to substitute differing values and data in a worksheet so that you can compare various possible outcomes.

Using the Scenario Manager, you can create a variety of scenarios; in each scenario, the worksheet data is modified to produce a specific set of results. Different scenarios yield different results. For example, you can compare best case scenarios to worst case scenarios.

The first step in creating a scenario is to select the changing cells; changing cells store the data you want to replace in creating a what-if analysis. Changing cells usually play a key part in the formula. You can define up to 32 changing cells per scenario. You can identify changing cells by their address or by a range name. Each changing cell in a scenario is then assigned a projected what-if value.

Once you have created a scenario, you can view it in the worksheet. Each scenario should be saved with a unique, descriptive name.

Once you have created a group of scenarios in a worksheet, you can then create a report to compare them.

Creating a Scenario

Discussion

Each scenario displays the result of inputting a different set of values into the changing cells of a worksheet. You should give each scenario you create a unique, descriptive name.

You must also indicate the changing cells, which can be adjacent or non-adjacent. If you select the changing cells before you open the Scenario Manager, their addresses or names will appear in the Changing cells box by default. If preferred, you can select the changing cells from the Add Scenario dialog box as well.
You can create a scenario for the original set of worksheet values, so that you can easily switch back to that display, as well as use the original values in scenario reports.

**Procedures**

1. Select the desired changing cells.
2. Select the **Data** tab.
3. Select the **What-If Analysis** button in the **Data Tools** group.
4. Select the **Scenario Manager** command.
5. Select **Add...**
6. Type the desired scenario name.
7. Select **OK**.
8. Type the desired value for the first changing cell.
9. Press **[Tab]** to select the box for the next changing cell.
10. Type the desired value for the next changing cell.
11. Once values have been entered for all changing cells, select OK.

12. Select Close.

![Step-by-Step](image)

From the Student Data directory, open **PROJ1.XLSX**. Create a scenario.

View the formulas in cells B8 and B9. These projections are based upon the percentages in cells E3 and E4.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the desired changing cells.  
*The range is selected.* | Drag across E3:E4 |
| 2. Select the **Data** tab.  
*The Data tab is displayed.* | Click **Data** |
| 3. Select the **What-If Analysis** button in the **Data Tools** group.  
*The What-If Analysis menu opens.* | Click **What-If Analysis** |
| 4. Select the **Scenario Manager** command.  
*The Scenario Manager dialog box opens.* | Click **Scenario Manager** |
| 5. Select **Add**.  
*The Scenario Manager dialog box closes, and the Add Scenario dialog box opens with the insertion point in the Scenario name box.* | Click **Add...** |
| 6. Type the desired scenario name.  
*The text appears in the Scenario name box.* | Type **Best Case** |
| 7. Select **OK**.  
*The Add Scenario dialog box closes, and the Scenario Values dialog box opens with the text in the first changing cell box selected.* | Click **OK** |
**Steps** | **Practice Data**
---|---
8. Type the desired value for the first changing cell. *The value appears in the first box.* | Type 25%
9. Press [Tab] to select the box for the next changing cell. *The second changing cell box is selected.* | Press [Tab]
10. Type the desired value for the next changing cell. *The value appears in the second box.* | Type 8%
11. Once values have been entered for all changing cells, select **OK**. *The Scenario Values dialog box closes, and the new scenario appears in the Scenarios list box in the Scenario Manager dialog box.* | Click 🟢
12. Select **Close**. *The Scenario Manager dialog box closes.* | Click Close

**Practice the Concept:** Add the following three scenarios, using the same changing cells:

<table>
<thead>
<tr>
<th>Scenario Name</th>
<th>Changing cell 1 (E3)</th>
<th>Changing cell 2 (E4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Worst Case</td>
<td>10%</td>
<td>10%</td>
</tr>
<tr>
<td>Best Projection</td>
<td>18%</td>
<td>12%</td>
</tr>
<tr>
<td>Original</td>
<td>15%</td>
<td>10%</td>
</tr>
</tbody>
</table>

Then, close the Scenario Manager dialog box.

---

**DISPLAYING A SCENARIO**

**Discussion**

After you have created one or more scenarios, you can view each of them. You can view more than one scenario without closing the Scenario Manager dialog box, as well as move the Scenario Manager dialog box on the screen as necessary to view the data.
The Scenario Manager dialog box

Procedures

1. Select the Data tab, if necessary.

2. Select the What-If Analysis button in the Data Tools group.

3. Select the Scenario Manager command.

4. Select the scenario you want to view in the Scenarios list box.

5. Select Show.

Step-by-Step

Display a scenario.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Data tab, if necessary.</td>
<td>Click Data</td>
</tr>
</tbody>
</table>

The Data tab is displayed.
<table>
<thead>
<tr>
<th><strong>Steps</strong></th>
<th><strong>Practice Data</strong></th>
</tr>
</thead>
</table>
| 2. Select the What-If Analysis button in the Data Tools group.  
*The What-If Analysis menu opens.* | Click ![What-If Analysis](#) |
| 3. Select the Scenario Manager command.  
*The Scenario Manager dialog box opens.* | Click Scenario Manager |
| 4. Select the scenario you want to view in the Scenarios list box.  
*The scenario is selected.* | Click Best Case |
| 5. Select Show.  
*The formulas in the worksheet recalculate.* | Click ![Show](#) |

**Practice the Concept:** Show the Worst Case and Best Projection scenarios. Then, show the Original scenario and close the Scenario Manager dialog box.

---

**EDITING A SCENARIO**

### Discussion

You can edit any scenario you have created. You can change the values in the changing cells, or you can indicate different changing cells altogether. You can also change the name of the scenario.

### Procedures

1. Select the Data tab, if necessary.

2. Select the What-If Analysis button ![What-If Analysis](#) in the Data Tools group.

3. Select the Scenario Manager command.

4. Select the name of the scenario you want to edit in the Scenarios list box.

5. Select ![Edit...](#).

6. Select ![OK](#).
Lesson 4 - Using Scenarios and Goal Seeking

7. Select the text in the box of the changing cells you want to edit.
8. Type the desired value.
9. Select OK.
10. Select Show to view the edited scenario.
11. Select Close.

Step-by-Step

Edit a scenario.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Data tab, if necessary. The Data tab is displayed.</td>
<td>Click Data</td>
</tr>
<tr>
<td>2. Select the What-If Analysis button in the Data Tools group. The What-If Analysis menu opens.</td>
<td>Click What-If Analysis</td>
</tr>
<tr>
<td>3. Select the Scenario Manager command. The Scenario Manager dialog box opens.</td>
<td>Click Scenario Manager</td>
</tr>
<tr>
<td>4. Select the name of the scenario you want to edit in the Scenarios list box. The scenario is selected.</td>
<td>Click Worst Case</td>
</tr>
<tr>
<td>5. Select Edit. The Scenario Manager dialog box closes, and the Edit Scenario dialog box opens.</td>
<td>Click Edit...</td>
</tr>
<tr>
<td>6. Select OK. The Edit Scenario dialog box closes, and the Scenario Values dialog box opens.</td>
<td>Click OK</td>
</tr>
<tr>
<td>7. Select the text in the box of the changing cell you want to edit. The text in the box is selected.</td>
<td>Double-click in the second box</td>
</tr>
<tr>
<td>8. Type the desired value. The value appears in the box. Type 15%</td>
<td></td>
</tr>
</tbody>
</table>
**CREATE A SCENARIO SUMMARY REPORT**

**Discussion**

You can create a scenario summary report. A scenario summary report lists all the scenarios you have created in a side-by-side format so that you can compare them.

When you create a scenario summary report, the Scenario Manager automatically inserts a **Scenario Summary** sheet in the workbook and places the report on it. This sheet allows you to easily view and print the scenario summary report.

A scenario summary report appears in an outline format.
Procedures

1. Select the Data tab, if necessary.

2. Select the What-If Analysis button in the Data Tools group.

3. Select the Scenario Manager command.

4. Select 

5. Under Report type, select the desired option.

6. Click the Collapse Dialog button in the Result cells box.

7. Select the cells you want to display in the scenario summary report.

8. Click the Expand Dialog button.

9. Select 

A scenario summary report
## Step-by-Step

Create a scenario summary report.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the **Data** tab, if necessary.  
*The Data tab is displayed.* | Click **Data** |
| 2. Select the **What-If Analysis** button in the **Data Tools** group.  
*The What-If Analysis menu opens.* | Click ![What-If Analysis](image) |
| 3. Select the **Scenario Manager** command.  
*The Scenario Manager dialog box opens.* | Click **Scenario Manager** |
| 4. Select **Summary**.  
*The Scenario Manager dialog box closes, and the Scenario Summary dialog box opens.* | Click ![Summary](image) |
| 5. Under **Report type**, select the desired option.  
*The option is selected.* | Click ![Scenario summary](image), if necessary |
| 6. Click the **Collapse Dialog** button in the **Result cells** box.  
*The Scenario Summary dialog box collapses.* | Click **Result cells** ![Result cells](image) |
| 7. Select the cells you want to display in the scenario summary report.  
*The cells are selected and the range appears in the collapsed Scenario Summary dialog box.* | Drag across B8:B10 |
| 8. Click the **Expand Dialog** button.  
*The Scenario Summary dialog box expands.* | Click ![Expand Dialog](image) |
| 9. Select **OK**.  
*The Scenario Summary dialog box closes, and the scenario summary report appears on a new Scenario Summary sheet in the workbook.* | Click ![OK](image) |

After you have finished viewing the scenario summary report, display the **Sheet1** worksheet.
Lesson 4 - Using Scenarios and Goal Seeking

Excel 2007 - Lvl 4

Lesson 4 - Using Scenarios and Goal Seeking

Using Goal Seek

Discussion

Often a situation arises in which you know the value you want a formula to return, but not the values needed to arrive at that result. For example, you may want to borrow money to purchase a $3000 computer system; you can afford to borrow the money at $150 per month for two years. You can use the Goal Seek feature to determine the interest rate needed to meet your requirements.

When you use Goal Seek, Excel performs the time-consuming task of trial-and-error experimentation by changing the variable value until it arrives at the desired result.

In order to use Goal Seek, you must specify the following parameters:

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set cell</td>
<td>The cell containing the formula</td>
</tr>
<tr>
<td>To value</td>
<td>The value you want the formula to return</td>
</tr>
<tr>
<td>By changing cell</td>
<td>The cell containing the value to be changed</td>
</tr>
</tbody>
</table>

Goal Seek has found a solution
**Procedures**

1. Select the **Data** tab, if necessary.
2. Select the **What-If Analysis** button in the **Data Tools** group.
3. Select the **Goal Seek** command.
4. Select the cell containing the formula.
5. Select the **To value** box.
6. Type the value you want the formula to return.
7. Select the **By changing cell** box.
8. Select the cell containing the changing cell.
9. Select **OK**.
10. Select **OK**.

**Step-by-Step**

Use Goal Seek.

If necessary, display the **Sheet1** worksheet.

Change cell E3 to **12%** and cell E4 to **8%**. Make sure that you press [Enter] to save your entries.

<table>
<thead>
<tr>
<th><strong>Steps</strong></th>
<th><strong>Practice Data</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>Data</strong> tab, if necessary.</td>
<td><strong>Click Data</strong></td>
</tr>
<tr>
<td><em>The Data tab is displayed.</em></td>
<td></td>
</tr>
<tr>
<td>2. Select the <strong>What-If Analysis</strong> button in the <strong>Data Tools</strong> group.</td>
<td><strong>Click</strong> [What-If Analysis]</td>
</tr>
<tr>
<td><em>The What-If Analysis menu opens.</em></td>
<td></td>
</tr>
<tr>
<td>3 Select the <strong>Goal Seek</strong> command.</td>
<td><strong>Click Goal Seek</strong></td>
</tr>
<tr>
<td><em>The Goal Seek dialog box opens with the text in the Set cell box selected.</em></td>
<td></td>
</tr>
<tr>
<td>4. Select the cell containing the formula.</td>
<td><strong>Click cell B10</strong></td>
</tr>
<tr>
<td><em>The cell address appears in the Set cell box.</em></td>
<td></td>
</tr>
<tr>
<td>Steps</td>
<td>Practice Data</td>
</tr>
<tr>
<td>---------------------------------------------------------------------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>5. Select the <strong>To value</strong> box.</td>
<td>Press [Tab]</td>
</tr>
<tr>
<td><em>The insertion point appears in the To value box.</em></td>
<td></td>
</tr>
<tr>
<td>6. Type the value you want the formula to return.</td>
<td>Type <strong>2000000</strong></td>
</tr>
<tr>
<td><em>The number appears in the To value box.</em></td>
<td></td>
</tr>
<tr>
<td>7. Select the <strong>By changing cell</strong> box.</td>
<td>Press [Tab]</td>
</tr>
<tr>
<td><em>The insertion point appears in the By changing cell box.</em></td>
<td></td>
</tr>
<tr>
<td>8. Select the cell containing the changing cell.</td>
<td>Click cell E3</td>
</tr>
<tr>
<td><em>The cell address appears in the By changing cell box.</em></td>
<td></td>
</tr>
<tr>
<td>9. Select <strong>OK</strong>.</td>
<td>Click <strong>OK</strong></td>
</tr>
<tr>
<td><em>The Goal Seek dialog box closes, the Goal Seek Status dialog box opens, and the cells in the worksheet display the solution.</em></td>
<td></td>
</tr>
<tr>
<td>10. Select <strong>OK</strong>.</td>
<td>Click <strong>OK</strong></td>
</tr>
<tr>
<td><em>The Goal Seek Status dialog box closes.</em></td>
<td></td>
</tr>
</tbody>
</table>

Notice that sales must increase by 31% (cell E3) in order to achieve the desired net income.

The values in the affected cells have been changed by the Goal Seek process. If you did not want to save the results, you could now return to the original values by using the **Undo** button.

Close **PROJ1.XLSX**.
EXERCISE

USING SCENARIOS AND GOAL SEEKING

Task

Use the Scenario Manager and Goal Seek to project sales.

1. Open REPSALES.XLSX.

2. View the formula in cell C7. The projected Qtr 2 sales are calculated using the Qtr 1 sales and a growth factor of 5% (cell D2).

3. Create a scenario named Low Sales. Make D2 the changing cell and leave its value at 5%.

4. Create another scenario named High Sales. Enter 20% as the value of D2 for this scenario.

5. Create a third scenario named Best Guess; make the value of D2 12%.

6. View each of the scenarios.

7. Edit the Best Guess scenario: make the changing cell value 10%. Then, view the Best Guess scenario.

8. Create a scenario summary report using the result cell C14. Then, display the Sheet1 worksheet.

9. Use Goal Seek to determine the growth percentage necessary in order for sales to reach $495,000 in the second quarter. (Hint: Use cell D2 as the changing cell.)

10. Close the Goal Seek Status dialog box.

11. Close the workbook without saving it.
LESSON 5 -
USING DATA TABLES

In this lesson, you will learn how to:

- Work with data tables
- Place formulas in data tables
- Create a one-variable table
- Create a two-variable table
WORKING WITH DATA TABLES

Discussion

You can use a data table to vary one or two values in a formula and display all possible results in a table format. For example, you can use a data table to display all possible projected sales based on total sales and various growth percentages.

To create a data table, you must first indicate the input cells (the cells containing the variable values in the formula). In the previous sales growth example, the input cells are the cells containing the total sales and those containing the percentage growth figures. The input values are then substituted in the formula and the result of each calculation is placed into an output cell in the data table.

PLACING FORMULAS IN DATA TABLES

Discussion

When you create a data table, the placement of the formula used in the calculation varies, depending on the type of data table you are creating.

If a one-variable data table is arranged in columns, the formula used in the calculation is placed in the cell at the top of the output column. In the following diagram, column B is the output column and the formula must be placed in cell B1.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>FORMULA</td>
</tr>
<tr>
<td>2</td>
<td>25.67</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>12.50</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>35.42</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>97.16</td>
<td></td>
</tr>
</tbody>
</table>

A one-variable data table arranged in columns

If a one-variable data table is arranged in rows, the formula is placed in the cell to the left of the output row. In the following diagram, row 2 is the output row and the formula must be placed in cell A2.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>FORMULA</td>
<td>25.67</td>
<td>12.50</td>
<td>35.42</td>
<td>97.16</td>
</tr>
</tbody>
</table>

A one-variable data table arranged in rows
In a two-variable data table, both rows and column are used for input data, and the formula used in the calculations must be placed above the column input cells and to the left of the row input cells. In the following diagram, column A contains one set of variable input values and row 1 contains the other set of input values. Therefore, the formula must be placed in cell A1.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FORMULA</td>
<td>25.67</td>
<td>12.50</td>
<td>35.42</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A two-variable data table

CREATING A ONE-VARIABLE TABLE

Discussion

When you create a one-variable data table, you can enter the input values either across a row or down a column.

If the one-variable data table consists of two columns, the first column is the input column and contains all possible values for the variable. The second column is the output column. In addition, the cell directly above the output column must contain the formula used in calculating the data table.

If the one-variable data table consists of two rows, the top row is the input row, the second row is the output row, and the cell to the left of the output row must contain the formula.

The input cell is critical when creating a data table. It is the cell containing the variable value and must be included in the formula. As the variable values are entered into the data table, the result of each calculation appears in the corresponding cell in the output column.
A one-variable data table

Procedures

1. Select the cell immediately above the output column or to the left of the output row.
2. Type the formula you want to evaluate.
3. Press [Enter].
4. Select the entire range containing the formula, the input column/row, and the output column/row.
5. Select the Data tab.
6. Select the What-If Analysis button in the Data Tools group.
7. Select the Data Table command.
8. Select the Row input cell box or the Column input cell box, as necessary.
9. Select the input cell.
10. Select OK.
Step-by-Step

From the Student Data directory, open WHATIF1.XLSX. Create a one-variable data table.

If necessary, display the One Variable worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell immediately above the output column or to the left of the output row. The cell is selected.</td>
<td>Click cell B5</td>
</tr>
<tr>
<td>2. Type the formula you want to evaluate. The text appears in the cell and on the formula bar.</td>
<td>Type $a^2 \times b^2$</td>
</tr>
<tr>
<td>3. Press [Enter]. The result of the formula appears in the cell.</td>
<td>Press [Enter]</td>
</tr>
<tr>
<td>4. Select the entire range containing the formula, the input column/row, and the output column/row. The range is selected.</td>
<td>Drag across A5:B16</td>
</tr>
<tr>
<td>5. Select the Data tab. The Data tab is displayed.</td>
<td>Click Data</td>
</tr>
<tr>
<td>6. Select the What-If Analysis button in the Data Tools group. The What-If Analysis menu opens.</td>
<td>Click</td>
</tr>
<tr>
<td>7. Select the Data Table command. The Data Table dialog box opens.</td>
<td>Click Data Table</td>
</tr>
<tr>
<td>8. Select the Row input cell box or the Column input cell box, as necessary. The insertion point appears in the selected box.</td>
<td>Click in the Column input cell box</td>
</tr>
<tr>
<td>9. Select the input cell. The cell is selected and appears in the appropriate input cell box.</td>
<td>Click cell B2</td>
</tr>
<tr>
<td>10. Select OK. The Data Table dialog box closes, and the data table appears in the selected range.</td>
<td>Click OK</td>
</tr>
</tbody>
</table>
Practice the Concept: Select cell A2, type 159.99, and press [Enter]. Notice that the data table recalculates.

CREATING A TWO-VARIABLE TABLE

Discussion

You can create a two-variable data table that evaluates two different values in a formula. A two-variable data table uses both an input column and an input row. The input row is the first row and the input column is the first column in the data table. You enter the formula you want to calculate in the cell above the input column and to the left of the input row.

A two-variable data table references two input cells: one for column input data and one for row input data; both input cells must be included in the formula. The formula calculates each of the possible combinations of the row and column input cells and places the results in the corresponding, intersecting cell in the data table.

Creating a two-variable data table
Procedures

1. Select the cell above the column input values and to the left of the row input values.
2. Type the formula you want to evaluate.
3. Press [Enter].
4. Select the entire data table.
5. Select the Data tab, if necessary.
6. Select the What-If Analysis button in the Data Tools group.
7. Select the Data Table command.
8. Select the row input cell.
9. Select the Column input cell box.
10. Select the column input cell.
11. Select OK.

Step-by-Step

Create a two-variable data table.

Display the Two Variables worksheet. Notice that the unit prices of several items are listed across row 5 and that a list of increasing quantities appears down column A.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell above the column input values and to the left of the row input values. The cell is selected.</td>
<td>Click cell A5</td>
</tr>
<tr>
<td>2. Type the formula you want to evaluate. The text appears in the cell and on the formula bar.</td>
<td>Type =$a2*b2$</td>
</tr>
<tr>
<td>3. Press [Enter]. The result of the formula appears in the cell.</td>
<td>Press [Enter]</td>
</tr>
</tbody>
</table>
### Steps

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
</table>
| 4. | Select the entire data table.  
The range is selected.  |
| 5. | Select the **Data** tab, if necessary.  
The **Data** tab is displayed.  |
| 6. | Select the **What-If Analysis** button in the  
**Data Tools** group.  
The **What-If Analysis** menu opens.  |
| 7. | Select the **Data Table** command.  
The **Data Table** dialog box opens with the insertion point in the  
**Row input cell** box.  |
| 8. | Select the row input cell.  
The cell is selected and appears in the  
**Row input cell** box.  |
| 9. | Select the **Column input cell** box.  
The insertion point appears in the  
**Column input cell** box.  |
| 10. | Select the column input cell.  
The cell is selected and appears in the  
**Column input cell** box.  |
| 11. | Select **OK**.  
The **Table** dialog box closes, and the two-variable data table is created.  |

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Drag across A5:H16</td>
</tr>
<tr>
<td></td>
<td>Click <strong>Data</strong></td>
</tr>
<tr>
<td></td>
<td>Click <strong>What-If Analysis</strong></td>
</tr>
<tr>
<td></td>
<td>Click <strong>Data Table</strong></td>
</tr>
<tr>
<td></td>
<td>Click cell A2</td>
</tr>
<tr>
<td></td>
<td>Press <strong>[Tab]</strong></td>
</tr>
<tr>
<td></td>
<td>Click cell B2</td>
</tr>
<tr>
<td></td>
<td>Click <strong>OK</strong></td>
</tr>
</tbody>
</table>

Click anywhere in the worksheet to deselect the table.

**Practice the Concept:** In cell D5, type **85.12** and press **[Enter]**. Adjust the column width, if necessary, to view the values. Notice that Excel recalculates all the values in column D, based on the new input value.  
Close **WHATIF1.XLSX**.
EXERCISE

USING DATA TABLES

Task

Create one-variable and two-variable data tables.

1. Open COMMRATE.XLSX.

2. In cell B5, enter a formula to multiply the Unit Price value in cell A2 by the Quantity value in cell B2.

3. Create a one-variable data table in the range B4:G5. (Hint: The row input cell is the quantity in cell B2.)

4. In cell A11, enter a formula that multiplies the sales figure in cell A8 by the commission rate in cell B8.

5. Create a two-variable data table in the range A11:G16. (Hint: The row input cell is cell B8 and the column input cell is cell A8.)

6. Close the workbook without saving it.
LESSON 6 - CONSOLIDATING WORKSHEETS

In this lesson, you will learn how to:

- Use consolidated worksheets
- Consolidate by category
- Consolidate by position
**USING CONSOLIDATED WORKSHEETS**

**Discussion**

You can consolidate data from several worksheets into a single worksheet or from many different workbooks into a single workbook. For example, if you record sales data on several different worksheets, you can quickly consolidate the sum of all the data into a single worksheet.

To consolidate data, you must first select the range where you want to display the consolidated information (the consolidation table). The consolidation table can be on the same worksheet as the source data, in a separate worksheet, or even in a separate workbook. You must then select each range from which you want to consolidate information (the data sources). You can consolidate data from a maximum of 255 source ranges. You must also select the desired summary function; Excel provides eleven different summary functions, including **Sum**, **Count**, and **Average**.

The source data can be copied or linked to the consolidation table. If the source data is linked to the consolidation table, the consolidation table will be automatically updated each time the source data changes.

When Excel consolidates data, a row is added to the consolidation table for each source range. Although these rows are collapsed by default, you can expand them to view the source data.

**CONSOLIDATING BY CATEGORY**

**Discussion**

You can consolidate information based on categories. This method is useful when the column or row headings are the same for all the data you want to consolidate, but are positioned differently in each source area. For example, in a workbook tracking sales for several regions, the sales products may be organized differently in each worksheet. You can consolidate by category to sum all the sales products, regardless of their order on the individual worksheets.

When you consolidate by category, Excel matches the column or row labels to consolidate the data. Consequently, you must select the column and/or row headings in the consolidation table, as well as in all source areas.

A source area can contain categories of information not included in the consolidation table. Excel will only extract source information from categories that appear in the consolidation table.
Consolidating data by category

When you consolidate by category, the labels across each category must be identical; if any one label is misspelled or capitalized differently, that data will not be consolidated.

Procedures

1. Select the range for the consolidation table.
2. Select the Data tab.
3. Select the Consolidate button in the Data Tools group.
4. Select the Function list.
5. Select the desired function.
6. Select the Top row option under Use labels in.
7. Select the Left column option under Use labels in.
8. Select the Create links to source data option, if desired.
9. Click the Collapse Dialog button in the Reference box.
10. Select the first worksheet or workbook containing a source area you want to consolidate.
11. Select the desired source area, including the row and column labels.

12. Click the **Expand Dialog** button.

13. Select [Add](#).

14. Click the **Collapse Dialog** button in the Reference box.

15. Select the next worksheet or workbook containing a source area you want to consolidate.

16. Select the desired source area, including the row and column labels.

17. Click the **Expand Dialog** button.

18. Select [Add](#).

19. Select additional source areas as desired.

20. Select [OK](#).

### Step-by-Step

From the Student Data directory, open **TOTALREG.XLSX**. Consolidate by category.

View the row and column headings for the **All Regions**, **North**, **South**, and **Central** worksheets. Though placed in different positions, the category labels are identical. Notice that the **Central** worksheet contains a **Training** category not included in the other worksheets.

Display the **All Regions** worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the range for the consolidation table. <em>The range is selected.</em></td>
<td>Drag across C9:G14</td>
</tr>
<tr>
<td>2. Select the <strong>Data</strong> tab. <em>The Data tab is displayed.</em></td>
<td>Click <strong>Data</strong></td>
</tr>
<tr>
<td>3. Select the <strong>Consolidate</strong> button in the <strong>Data Tools</strong> group. <em>The Consolidate dialog box opens.</em></td>
<td>Click <strong>Consolidate</strong></td>
</tr>
<tr>
<td>4. Select the <strong>Function</strong> list. <em>A list of available functions is displayed.</em></td>
<td>Click <strong>Function</strong></td>
</tr>
<tr>
<td>Steps</td>
<td>Practice Data</td>
</tr>
<tr>
<td>---------------------------------------------------------------------</td>
<td>----------------------------------------</td>
</tr>
<tr>
<td>5. Select the desired function.</td>
<td>Click <strong>Sum</strong></td>
</tr>
<tr>
<td><em>The function is selected.</em></td>
<td></td>
</tr>
<tr>
<td>6. Select the <strong>Top row</strong> option under <strong>Use labels in.</strong></td>
<td>Click [ ] <strong>Top row</strong></td>
</tr>
<tr>
<td><em>The Top row option is selected.</em></td>
<td></td>
</tr>
<tr>
<td>7. Select the <strong>Left column</strong> option under <strong>Use labels in.</strong></td>
<td>Click [ ] <strong>Left column</strong></td>
</tr>
<tr>
<td><em>The Left column option is selected.</em></td>
<td></td>
</tr>
<tr>
<td>8. Select the <strong>Create links to source data</strong> option, if desired.</td>
<td>Click [ ] <strong>Create links to source data</strong></td>
</tr>
<tr>
<td><em>The Create links to source data option is selected.</em></td>
<td></td>
</tr>
<tr>
<td>9. Click the <strong>Collapse Dialog</strong> button in the <strong>Reference box.</strong></td>
<td>Click <img src="image" alt="Collapse Dialog" /></td>
</tr>
<tr>
<td><em>The Consolidate dialog box collapses.</em></td>
<td></td>
</tr>
<tr>
<td>10. Select the first worksheet or workbook containing a source area</td>
<td>Click the <strong>North</strong> tab</td>
</tr>
<tr>
<td>you want to consolidate.</td>
<td></td>
</tr>
<tr>
<td><em>The source area in the worksheet or workbook appears.</em></td>
<td></td>
</tr>
<tr>
<td>11. Select the desired source area, including the row and column</td>
<td>Drag across C9:G14</td>
</tr>
<tr>
<td>labels.</td>
<td></td>
</tr>
<tr>
<td><em>The range appears in the Consolidate dialog box.</em></td>
<td></td>
</tr>
<tr>
<td>12. Click the <strong>Expand Dialog</strong> button.</td>
<td>Click <img src="image" alt="Expand Dialog" /></td>
</tr>
<tr>
<td><em>The Consolidate dialog box expands.</em></td>
<td></td>
</tr>
<tr>
<td>13. Select <strong>Add</strong>.</td>
<td>Click <img src="image" alt="Add" /></td>
</tr>
<tr>
<td><em>The reference is added to the <strong>All references</strong> list box.</em></td>
<td></td>
</tr>
<tr>
<td>14. Click the <strong>Collapse Dialog</strong> button in the <strong>Reference box.</strong></td>
<td>Click <img src="image" alt="Collapse Dialog" /></td>
</tr>
<tr>
<td><em>The Consolidate dialog box collapses.</em></td>
<td></td>
</tr>
<tr>
<td>15. Select the next worksheet or workbook containing a source area</td>
<td>Click the <strong>South</strong> tab</td>
</tr>
<tr>
<td>you want to consolidate.</td>
<td></td>
</tr>
<tr>
<td><em>The source area in the worksheet or workbook appears.</em></td>
<td></td>
</tr>
</tbody>
</table>
Lesson 6 - Consolidating Worksheets

**Steps** | **Practice Data**
--- | ---
16. Select the desired source area, including the row and column labels. *The range appears in the Consolidate dialog box.* | Drag across C13:G23
17. Click the **Expand Dialog** button. *The Consolidate dialog box expands.* | Click ![Expand Dialog](image)
18. Select **Add**. *The reference is added to the All references list box.* | Click ![Add](image)
19. Select additional source areas as desired. *The references are added to the All references list box.* | Follow the instructions shown below the table before continuing on to the next step
20. Select **OK**. *The Consolidate dialog box closes, and the consolidated data appears in the consolidation table.* | Click ![OK](image)

Add the range C13:G19 on the **Central** worksheet to the **All references** list box.

*Return to the table and continue on to the next step (step 20).*

After you have completed the consolidation, click the 2 row level symbol in the upper, left corner of the **All Regions** worksheet to expand the outline and view the consolidation table details.

Close **TOTALREG.XLSX**.

---

**CONSOLIDATING BY POSITION**

**Discussion**

You can consolidate information based on the position of the source data. This method is useful when the data appears in the same location or relative position on all the worksheets you want to consolidate. For example, if you have three identical worksheets for three different products in which only the data varies, you can consolidate based on position.

When you consolidate by position, you do not need to include the row or column headings when specifying the destination and source ranges.
Source data does not need to be located in identical locations; it can be arranged in the same relative location. For example, the **Phone**, **Hotel**, and **Meals** data can be located in columns B, C, and D in one source and in columns G, H, and I in another.

---

**Procedures**

1. Select the range for the consolidation table, not including row or column labels.
2. Select the **Data** tab, if necessary.
3. Select the **Consolidate** button in the **Data Tools** group.
4. Select the **Function** list.
5. Select the desired function.
6. Select the **Create links to source data** option.
7. Click the **Collapse Dialog** button in the **Reference** box.
8. Select the first worksheet you want to consolidate.
9. Select the desired source data.
10. Click the **Expand Dialog** button.

11. Select **Add**.

12. Add other source data ranges to the **All references** list box, as necessary.

13. Select **OK**.

### Step-by-Step

From the Student Data directory, open **WEEKLY.XLSX**. Consolidate by position.

View the **Central**, **Southeast**, and **Northeast** worksheets. Notice that the arrangement of the data is the same, even though it appears at different locations on each worksheet. Notice also that the column and row labels are different as well. Display the **By Week** worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the range for the consolidation table, not including row or column labels. <em>The range is selected.</em></td>
<td>Drag B8:D12</td>
</tr>
<tr>
<td>2. Select the <strong>Data</strong> tab, if necessary. <em>The Data tab is displayed.</em></td>
<td>Click <strong>Data</strong></td>
</tr>
<tr>
<td>3. Select the <strong>Consolidate</strong> button in the <strong>Data Tools</strong> group. <em>The Consolidate dialog box opens.</em></td>
<td>Click <strong>Consolidate</strong></td>
</tr>
<tr>
<td>4. Select the <strong>Function</strong> list. <em>A list of available functions is displayed.</em></td>
<td>Click <strong>Function</strong></td>
</tr>
<tr>
<td>5. Select the desired function. <em>The function is selected.</em></td>
<td>Click <strong>Average</strong></td>
</tr>
<tr>
<td>6. Select the <strong>Create links to source data</strong> option. <em>The Create links to source data option is selected.</em></td>
<td>Click <strong>Create links to source data</strong></td>
</tr>
<tr>
<td>7. Click the <strong>Collapse Dialog</strong> button in the <strong>Reference</strong> box. <em>The Consolidate dialog box collapses.</em></td>
<td>Click <strong>Collapse</strong></td>
</tr>
<tr>
<td>Steps</td>
<td>Practice Data</td>
</tr>
<tr>
<td>-------</td>
<td>---------------</td>
</tr>
</tbody>
</table>
| 8. Select the first worksheet you want to consolidate.  
*The worksheet appears.* | Click the **Central** tab |
| 9. Select the desired source data.  
*The data is selected.* | Drag B9:D13 |
| 10. Click the **Expand Dialog** button.  
*The Consolidate dialog box expands, and the selected range appears in the Reference box.* | Click |
| 11. Select **Add**.  
*The range is added to the All references list.* | Click **Add** |
| 12. Add other source data ranges to the All references list box, as necessary.  
*The references appear in the All references list.* | Follow the instructions shown below the table before continuing on to the next step |
| 13. Select **OK**.  
*The Consolidate dialog box closes, and the data is consolidated by position.* | Click **OK** |

Add the source data in D8:F12 on the **Southeast** worksheet. Then, add the source data in B7:D11 on the **Northeast** worksheet.

*Return to the table and continue on to the next step (step 13).*  
Close **WEEKLY.XLSX**.
EXERCISE

CONSOLIDATING WORKSHEETS

Task

Consolidate data from several worksheets by position and category.

1. Open YEAR7.XLSX.
2. Notice that the data in the Qtr 1 and Qtr 2 worksheets are identically arranged, but with different column and row headings.
3. Display the Semi worksheet.
4. Select the range B4:H8 as the consolidation table and then open the Consolidate dialog box.
5. Use the Sum function and create links to the source data.
6. Select and add the data in B3:H7 in both the Qtr 1 and Qtr 2 worksheets. Then, consolidate the data.
7. Display the Totals worksheet.
8. Select the range A3:E8 as the consolidation table and open the Consolidate dialog box.
9. Use the Sum function and create links to the source data. Because you are consolidating by category, select the Top row and Left column options.
10. Select the data in the range A3:H18 in the Semi worksheet and add it to the All references list box. (Hint: It does not matter if the table is expanded or collapsed.)
11. Select and add the data in A2:K7 in the Qtr 3 worksheet; select and add the data in A2:H11 in the Qtr 4 worksheet. Then, consolidate the data.
12. Close the workbook without saving it.
LESSON 7 - SOLVING PROBLEMS

In this lesson, you will learn how to:

- Use Solver
- Save a solution as a scenario
- Restore the original values
- Change a constraint
- Create a Solver report
- View solutions using scenarios
USING SOLVER

Discussion

Solver allows you to solve problems by working backwards; if you know the desired outcome, you can use Solver to find the variables that produce that result. For example, if you want maximum profits, you can use Solver to determine which combination of products will maximize the profit within the constraints of production and inventory.

You can specify the following items in the Solver Parameters dialog box:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Set Target cell</strong></td>
<td>The cell containing the formula you want Solver to use</td>
</tr>
<tr>
<td><strong>Equal To</strong></td>
<td>Whether to set the target cell to its maximum value, its minimum value, or a specified value</td>
</tr>
<tr>
<td><strong>By Changing cells</strong></td>
<td>The variable cells that Solver can change in order to reach the desired target cell value</td>
</tr>
<tr>
<td><strong>Subject to the Constraints</strong></td>
<td>You can set constraints or limitations on the changing cells, the target cell, or other cells in the worksheet</td>
</tr>
</tbody>
</table>

After you have entered all necessary information, you can solve the problem. Solver tries various inputs until it reaches an answer that satisfies all the constraints and other criteria. Each try is called an iteration. If a solution is not found within a default number of iterations, a message box asks if you want to continue.
Using Solver

In order to use Solver, the Solver add-in must be installed. In addition, the Solver Add-in option must be selected in the Add-Ins dialog box. The Add-Ins dialog box can be accessed via the Add-Ins page in the Excel Options dialog box by selecting the Go button.

You can continue to add constraints without closing the Add Constraint dialog box by selecting the Add button and entering the desired constraints. When you are finished, you can press the [Enter] key to close the Add Constraint dialog box.

Procedures

1. Select the Data tab.
2. Select the Solver button in the Analysis group.
3. Click the Collapse Dialog button in the Set Target Cell box.
4. Select the desired target cell.
5. Click the Expand Dialog button.
6. Under Equal To, select the desired option.
7. Click the Collapse Dialog button in the By Changing Cells box.
8. Select the changing cells.
9. Click the Expand Dialog button.
10. Under Subject to the Constraints, select Add.
11. Click the Collapse Dialog button in the Cell Reference box.
12. Select the range of cells to which you want to add a constraint.
13. Click the Expand Dialog button.
14. Select the operator list.
15. Select the desired operator.
16. Click the Collapse Dialog button in the Constraint box.
17. Select the range of cells you want to use as a constraint.
18. Click the Expand Dialog button.
19. Select OK.
20. Add additional constraints as desired.
21. Select Solve.

Step-by-Step

From the Student Data directory, open CAMP1.XLSX. Use Solver.

Solver must be installed on your computer in order to complete this step-by-step. Solver can be installed from the Add-Ins page in the Excel Options dialog box.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the Data tab.  
*The Data tab is displayed.* | Click Data |
| 2. Select the Solver button in the Analysis group.  
*The Solver Parameters dialog box opens.* | Click Solver |
<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>3. Click the <strong>Collapse Dialog</strong> button in the <strong>Set Target Cell</strong> box. The Solver Parameters dialog box collapses.</td>
<td>Click <strong>Set Target Cell</strong> <img src="image1" alt="Image" /></td>
</tr>
<tr>
<td>4. Select the desired target cell. The target cell is selected and appears in the collapsed Solver Parameters dialog box.</td>
<td>Click cell E15</td>
</tr>
<tr>
<td>5. Click the <strong>Expand Dialog</strong> button. The Solver Parameters dialog box expands and the target cell appears in the <strong>Set Target Cell</strong> box.</td>
<td><img src="image2" alt="Image" /></td>
</tr>
<tr>
<td>6. Under <strong>Equal To</strong>, select the desired option. The option is selected.</td>
<td>Click <img src="image3" alt="Image" /> <strong>Max</strong>, if necessary</td>
</tr>
<tr>
<td>7. Click the <strong>Collapse Dialog</strong> button in the <strong>By Changing Cells</strong> box. The Solver Parameters dialog box collapses.</td>
<td>Click <strong>By Changing Cells</strong> <img src="image4" alt="Image" /></td>
</tr>
<tr>
<td>8. Select the changing cells. The cells are selected and appear in the collapsed Solver Parameters dialog box.</td>
<td>Drag C4:D4</td>
</tr>
<tr>
<td>9. Click the <strong>Expand Dialog</strong> button. The Solver Parameters dialog box expands, and the changing cells appear in the <strong>By Changing Cells</strong> box.</td>
<td><img src="image5" alt="Image" /></td>
</tr>
<tr>
<td>10. Under <strong>Subject to the Constraints</strong>, select <strong>Add</strong>. The Solver Parameters dialog box closes, and the Add Constraint dialog box opens.</td>
<td>Click <img src="image6" alt="Image" /> <strong>Add</strong></td>
</tr>
<tr>
<td>11. Click the <strong>Collapse Dialog</strong> button in the <strong>Cell Reference</strong> box. The Add Constraint dialog box collapses.</td>
<td>Click <strong>Cell Reference</strong> <img src="image7" alt="Image" /></td>
</tr>
<tr>
<td>12. Select the range of cells to which you want to add a constraint. The cells are selected, and appear in the collapsed Add Constraint dialog box.</td>
<td>Drag E6:E12</td>
</tr>
<tr>
<td>Steps</td>
<td>Practice Data</td>
</tr>
<tr>
<td>-------</td>
<td>---------------</td>
</tr>
<tr>
<td>13. Click the <strong>Expand Dialog</strong> button. The Add Constraint dialog box expands, and the range appears in the Cell Reference box.</td>
<td>Click</td>
</tr>
<tr>
<td>14. Select the operator list. A list of available operators appears.</td>
<td>Click in the box to the right of the Cell Reference box</td>
</tr>
<tr>
<td>15. Select the desired operator. The operator appears in the box.</td>
<td>Click &lt;=</td>
</tr>
<tr>
<td>16. Click the <strong>Collapse Dialog</strong> button in the Constraint box. The Add Constraint dialog box collapses.</td>
<td>Click the Constraint</td>
</tr>
<tr>
<td>17. Select the range of cells you want to use as a constraint. The cells are selected, and appear in the collapsed Add Constraint dialog box.</td>
<td>Drag B6:B12</td>
</tr>
<tr>
<td>18. Click the <strong>Expand Dialog</strong> button. The Add Constraint dialog box expands, and the range appears in the Constraint box.</td>
<td>Click</td>
</tr>
<tr>
<td>19. Select <strong>OK</strong>. The Add Constraint dialog box closes, the Solver Parameters dialog box opens, and the constraint appears in the Subject to the Constraints list box.</td>
<td>Click <strong>OK</strong></td>
</tr>
<tr>
<td>20. Add additional constraints as desired. All constraints appear in the Subject to the Constraints list box.</td>
<td>Follow the instructions shown below the table before continuing on to the next step</td>
</tr>
<tr>
<td>21. Select <strong>Solve</strong>. The Solver Parameters dialog box closes, and the Solver Results dialog box opens.</td>
<td>Click <strong>Solve</strong></td>
</tr>
</tbody>
</table>

Add the following additional constraint:

\[ $D4 \geq 3 \]

Return to the table and continue on with the next step (step 21).
Move the Solver Results dialog box as necessary to view the results.

### SAVING A SOLUTION AS A SCENARIO

#### Discussion

Once Solver has arrived at a solution, you can view one of three available reports, restore the original values, or save the results as a scenario. Saving the results as a scenario allows you to view them at any later time; you can store the results of several solutions.

#### Procedures

1. Use Solver to solve a problem.
2. In the Solver Results dialog box, select ![Save Scenario](Save Scenario.png).
3. Type the desired scenario name.
4. Select ![OK](OK.png).

#### Step-by-Step

Save a Solver solution as a scenario.

If necessary, use Solver to solve a problem.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. In the Solver Results dialog box, select <strong>Save Scenario</strong>. &lt;br&gt; <em>The Solver Results dialog box closes, and the Save Scenario dialog box opens with the insertion point in the Scenario Name box.</em></td>
<td>Click ![Save Scenario](Save Scenario.png)</td>
</tr>
<tr>
<td>2. Type the desired scenario name. &lt;br&gt; <em>The text appears in the Scenario Name box.</em></td>
<td>Type <strong>First Try</strong></td>
</tr>
</tbody>
</table>
Steps | Practice Data
---|---
3. Select OK. The Save Scenario dialog box closes, the Solver solution is saved as a scenario, and the Solver Results dialog box opens. | Click OK

Restoring the Original Values

Discussion

Once Solver has found a solution, the suggested values are automatically inserted into the worksheet. You can keep the Solver values in the worksheet, or you can restore the original values.

If you choose to restore the original values, the Solver solution is lost, unless you have saved it as a scenario.

Procedures

1. If necessary, use Solver to solve a problem.
2. In the Solver Results dialog box, select the **Restore Original Values** option.
3. Select OK.

Step-by-Step

Restore the original values to a worksheet.

If necessary, use Solver to solve a problem.

Steps | Practice Data
---|---
1. In the Solver Results dialog box, select the **Restore Original Values** option. The **Restore Original Values** option is selected. | Click [Restore Original Values]
### Changing a Constraint

#### Discussion

Once you have used Solver in a worksheet, Excel retains the Solver settings. You can open the Solver Parameters dialog box and change those settings. This option allows you to change constraints and run Solver again.

#### Procedures

1. Select the Data tab, if necessary.

2. Select the Solver button in the Analysis group.

3. In the Subject to the Constraints list box, select the constraint you want to change.

4. Select Change.

5. Select the text you want to change.

6. Type the desired text.

7. Select OK.

#### Step-by-Step

Change a constraint.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Data tab, if necessary. The Data tab is displayed.</td>
<td>Click Data</td>
</tr>
<tr>
<td><strong>Steps</strong></td>
<td><strong>Practice Data</strong></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>
| 2. Select the **Solver** button in the **Analysis** group.  
*The Solver Parameters dialog box opens.* | Click ![Solver](image)
| 3. In the **Subject to the Constraints** list box, select the constraint you want to change.  
*The constraint is selected.* | Click $D4 \geq 3$
| 4. Select **Change**.  
*The Solver Parameters dialog box closes, and the Change Constraint dialog box opens.* | Click ![Change](image)
| 5. Select the text you want to change.  
*The text is selected.* | Press `[Tab]` twice
| 6. Type the desired text.  
*The text appears in the applicable box.* | Type 7
| 7. Select **OK**.  
*The Change Constraint dialog box closes, and the Solver Parameters dialog box opens with the edited constraint in the **Subject to the Constraints** list box.* | Click ![OK](image)

Select **Solve**. Move the Solver Results dialog box, if necessary, and notice that the changed constraint yields a different number. Save the scenario as **Second Try** and then restore the original values. Close the Solver Results dialog box.

## Creating a Solver Report

### Discussion

Solver provides three reports, each of which displays different information on how Solver reached the solution. The **Answer** report lists the original and final values of the target and the changing cells, as well as the effect of the constraints. The **Sensitivity** report indicates how sensitive the model is to changes in the changing cells and the constraints. The **Limits** report lists the upper and lower limits imposed on the target and changing cells.

You can choose any or all of the reports from the Solver Results dialog box. Excel inserts a new worksheet in the workbook for each report you select.
Procedures

1. Select the **Data** tab, if necessary.

2. Select the **Solver** button in the **Analysis** group.

3. Select **Solve**.

4. Select the desired reports from the **Reports** list box.

5. Select **OK**.

Step-by-Step

Create a Solver report.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>Data</strong> tab, if necessary. <em>The Data tab is displayed.</em></td>
<td>Click <strong>Data</strong></td>
</tr>
<tr>
<td>Steps</td>
<td>Practice Data</td>
</tr>
<tr>
<td>-------</td>
<td>---------------</td>
</tr>
<tr>
<td>2. Select the Solver button in the Analysis group. The Solver Parameters dialog box opens.</td>
<td>Click <img src="solver.png" alt="Solver" /></td>
</tr>
<tr>
<td>3. Select Solve. The Solver Parameters dialog box closes, and the Solver Results dialog box opens.</td>
<td>Click <img src="solve.png" alt="Solve" /></td>
</tr>
<tr>
<td>4. Select the desired reports from the Reports list box. The reports are selected.</td>
<td>Click <img src="answer.png" alt="Answer" /></td>
</tr>
<tr>
<td>5. Select OK. The Solver Results dialog box closes, and a worksheet is inserted for each report selected.</td>
<td>Click <img src="ok.png" alt="OK" /></td>
</tr>
</tbody>
</table>

Display the Answer Report 1 worksheet and review the report.

**Practice the Concept:** Display the Sheet1 worksheet. Then, create and review the Sensitivity and Limits reports (you can select both reports at the same time).

---

**VIEWING SOLUTIONS USING SCENARIOS**

![Discussion](discussion.png)

**Discussion**

If you have saved a Solver solution as a scenario, you can open and view the scenario just as you would open and view any scenario.

Displaying the saved scenario does not change any of the current selections in the Solver Parameters dialog box.

![Procedures](procedures.png)

**Procedures**

1. Select the Data tab, if necessary.
2. Select the What-If Analysis button in the Data Tools group.
3. Select the Scenario Manager command.
4. Select the desired scenario.
5. Select Show.


---

**Step-by-Step**

View a Solver solution saved as a scenario.

If necessary, display the Sheet1 sheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>Data</strong> tab, if necessary. <em>The Data tab is displayed.</em></td>
<td>Click <strong>Data</strong></td>
</tr>
<tr>
<td>2. Select the <strong>What-If Analysis</strong> button in the <strong>Data Tools</strong> group. <em>The What-If Analysis menu opens.</em></td>
<td>Click <strong>What-If Analysis</strong></td>
</tr>
<tr>
<td>3. Select the <strong>Scenario Manager</strong> command. <em>The Scenario Manager dialog box opens.</em></td>
<td>Click <strong>Scenario Manager</strong></td>
</tr>
<tr>
<td>4. Select the desired scenario. <em>The scenario is selected.</em></td>
<td>Click <strong>First Try</strong></td>
</tr>
<tr>
<td>5. Select Show. <em>The scenario appears in the worksheet.</em></td>
<td>Click <strong>Show</strong></td>
</tr>
<tr>
<td>6. Select Close. <em>The Scenario Manager dialog box closes.</em></td>
<td>Click <strong>Close</strong></td>
</tr>
</tbody>
</table>

**Practice the Concept:** Show the **Second Try** scenario.
Close CAMP1.XLSX.
EXERCISE

SOLVING PROBLEMS

Task

Use Solver with constraints.

1. Open SPORTS.XLSX.
2. Use Solver to find the maximum total profits from shipping both small and large packages. (Hint: Select cell E13 as the target cell and Max as the option to equal.)
3. Select the range C4:D4 as the cells to be changed.
4. Add the constraint that the stock used to create packages cannot exceed inventory. (Hint: Select the range E6:E10 as the cell reference, use the <= operator, and select the range B6:B10 as the constraint cells.)
5. Add a second constraint that at least five large packages must be created. (Hint: Select cell D4 as the cell reference, select the >= operator, and type 5 as the constraint.)
6. Solve for the answer. Save the new solution as a scenario named Best Case. Then, restore the original values to the worksheet.
7. Change the constraint in cell D4 to 7 and solve for the new result. Save the new solution as a scenario named Large Shipments and restore the original values to the worksheet.
8. Use the Scenario Manager to view the Large Shipments solution. Then, show the Best Case solution and close Scenario Manager.
9. Open Solver and solve using the defined parameters. Then, create and view an Answer report.
10. Close the workbook without saving it.
LESSON 8 -
WORKING WITH VIEWS

In this lesson, you will learn how to:

- Use views
- Create a normal view
- Create a custom view
- Display a view
- Delete a custom view
**USING VIEWS**

**Discussion**

You can use views to save worksheet and/or print settings. Once you have stored settings in a view, you can use that view to display and/or print the worksheet with the selected settings. As a result, you do not have to manually change worksheet settings each time you want to display or print the worksheet differently.

Views allow you to experiment with different looks for your worksheet. For example, if you need to print different print areas of a worksheet in separate print jobs on a regular basis, you can create a view for each print area. Then, when you need to print a particular print area, you can simply select the appropriate view.

Any view you create includes: the worksheet window size, position, and zoom setting; any hidden columns or rows; and all defined print settings.

**CREATING A NORMAL VIEW**

**Discussion**

The first view you create should be of the worksheet view before you change any settings. Creating a normal view allows you to easily restore the original worksheet settings. For example, if you zoom the third worksheet in a workbook to 200%, hide some columns and rows, and then save these settings as a view, you can use the normal view (if one has been saved) to quickly restore your default worksheet settings.

![Creating a normal view](image-url)
Procedures

1. Select the View tab.

2. Select the Custom Views button in the Workbook Views group.

3. Select Add.

4. Type the desired name for the normal view.

5. Select OK.

Step-by-Step

From the Student Data directory, open REPFIG1.XLSX. Create a normal view.

If necessary, display the Bonuses worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the View tab. The View tab is displayed.</td>
<td>Click View</td>
</tr>
<tr>
<td>2. Select the Custom Views button in the Workbook Views group. The Custom Views dialog box opens.</td>
<td>Click Custom Views</td>
</tr>
<tr>
<td>3. Select Add. The Custom Views dialog box closes, and the Add View dialog box opens with the insertion point in the Name box.</td>
<td>Click Add...</td>
</tr>
<tr>
<td>4. Type the desired name for the normal view. The text appears in the Name box.</td>
<td>Type Normal</td>
</tr>
<tr>
<td>5. Select OK. The Add View dialog box closes, and the normal view is saved.</td>
<td>Click OK</td>
</tr>
</tbody>
</table>
Creating a Custom View

Discussion

If you typically use display and print settings different from the normal view settings, you can create a custom view. For example, if you consistently view and print the third worksheet zoomed to 50% of normal magnification with a specific print area, you can create a view that stores these settings.

Views store display options (such as the cell pointer location and the size and position of the window), as well as print settings (such as the page setup options and print areas). When creating a view, you can choose to include the current print settings and/or the hidden row, column, and filter settings.

Views only apply to the worksheet in which they were created. You cannot apply a view created in one worksheet to another.

Because all the views for a workbook appear in the Custom Views dialog box, it is helpful to include the worksheet name with the custom view name.

Procedures

1. Modify the worksheet display and/or print settings as desired.
2. Select the View tab, if necessary.
3. Select the Custom Views button in the Workbook Views group.
4. Select Add...
5. Type the desired name for the view.
6. Select OK.

Step-by-Step

Create a custom view.
If necessary, display the **Bonuses** worksheet.

Hide columns B through E and deselect the **Gridlines** option on the **Sheet** page in the Page Setup dialog box.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the **View** tab, if necessary.  
*The View tab is displayed.* | Click **View** |
| 2. Select the **Custom Views** button in the **Workbook Views** group.  
*The Custom Views dialog box opens.* | Click **Custom Views** |
| 3. Select **Add.**  
*The Custom Views dialog box closes, and the Add View dialog box opens with the insertion point in the Name box.* | Click **Add...** |
| 4. Type the desired name for the view.  
*The text appears in the Name box.* | Type **Print Bonuses** |
| 5. Select **OK.**  
*The Add View dialog box closes, and the view is saved.* | Click **OK** |

---

**DISPalyING A VIEW**

**Discussion**

Once you have created a view, you can use the Custom Views dialog box to quickly switch to it. For example, if you have created a view in which the print settings are set to landscape with custom headers and footers, you can use the Custom Views dialog box to display the corresponding view.

**Procedures**

1. Select the **View** tab, if necessary.

2. Select the **Custom Views** button in the **Workbook Views** group.

3. Select the view you want to display.
4. Select ![Show](https://www.example.com).

### Step-by-Step

Display a view.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the View tab, if necessary. <em>The View tab is displayed.</em></td>
<td>Click View</td>
</tr>
<tr>
<td>2. Select the Custom Views button in the Workbook Views group. <em>The Custom Views dialog box opens.</em></td>
<td>Click <img src="https://www.example.com" alt="Custom Views" /></td>
</tr>
<tr>
<td>3. Select the view you want to display. <em>The view is selected.</em></td>
<td>Click Normal, if necessary</td>
</tr>
<tr>
<td>4. Select Show. <em>The Custom Views dialog box closes, and the corresponding view appears.</em></td>
<td>Click <img src="https://www.example.com" alt="Show" /></td>
</tr>
</tbody>
</table>

Practice the Concept: Display the Print Bonuses view.

### DELETING A CUSTOM VIEW

#### Discussion

If you no longer use a custom view, you can delete it. Deleting a custom view permanently removes it from the workbook.
Deleting a custom view

Procedures

1. Select the View tab, if necessary.

2. Select the Custom Views button in the Workbook Views group.

3. Select the view you want to delete in the Views list box.

4. Select Delete

5. Select Yes

6. Select Close

Step-by-Step

Delete a custom view.
<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>View</strong> tab, if necessary. <em>The View tab is displayed.</em></td>
<td>Click <strong>View</strong></td>
</tr>
<tr>
<td>2. Select the <strong>Custom Views</strong> button in the <strong>Workbook Views</strong> group. <em>The Custom Views dialog box opens.</em></td>
<td>Click ![Custom Views]</td>
</tr>
<tr>
<td>3. Select the view you want to delete in the <strong>Views</strong> list box. <em>The view is selected.</em></td>
<td>Click <strong>Print Bonuses</strong></td>
</tr>
<tr>
<td>4. Select <strong>Delete</strong>. <em>A Microsoft Office Excel message box opens, asking you to verify the deletion.</em></td>
<td>Click ![Delete]</td>
</tr>
<tr>
<td>5. Select <strong>Yes</strong>. <em>The Microsoft Office Excel message box closes, and the view is deleted.</em></td>
<td>Click ![Yes]</td>
</tr>
<tr>
<td>6. Select <strong>Close</strong>. <em>The Custom Views dialog box closes.</em></td>
<td>Click ![Close]</td>
</tr>
</tbody>
</table>

**Practice the Concept:** Delete the **Normal** view and close the Custom Views dialog box. Close **REPFIG1.XLSX**.
EXERCISE

WORKING WITH VIEWS

Task

Work with views.

1. Open REGION20.XLSX.
2. Display the Totals worksheet and create a normal view called Normal.
3. Hide columns B through D and select the Gridlines option. (Hint: Use the Sheet page in the Page Setup dialog box.)
4. Create a custom view called Print Totals.
5. Display the Normal view.
6. Display the Projections worksheet. Select a print range of B6:C14 and have rows 1 to 3 and column A repeat. (Hint: Use the Sheet page in the Page Setup dialog box.)
7. View the worksheet in print preview. Then, close print preview and create a custom view called Projections (Q1, Q2).
8. Delete the Print Totals custom view.
9. Close the workbook without saving it.
LESSON 9 -
USING AUDITING TOOLS

In this lesson, you will learn how to:

- Show formulas
- Display/Remove dependent arrows
- Display/Remove precedent arrows
- Use the Trace Error button
- Trace cells causing errors
- Use the Evaluate Formula button
- Use the Error Checking button
- Use the Watch Window
SHOWING FORMULAS

Discussion

The **Formula Auditing** group on the **Formulas** tab provides several tools that allow you to examine the association between cells and formulas in a worksheet. These tools help you avoid common formula and data-entry errors.

The **Show Formulas** button displays all the formulas in your worksheet instead of their results. With the formulas displayed, if you select a cell containing a formula, Excel applies a different color to each reference in the formula and displays a correspondingly colored border around each referenced cell.

You can quickly toggle between displaying formulas and displaying their results.

![Displaying worksheet formulas](image)

Procedures

1. Select the **Formulas** tab in the **Ribbon**.
2. To display worksheet formulas, select **Show Formulas** in the **Formula Auditing** group.
3. To highlight formula references, select a cell containing a formula.

4. To display formula results, select in the Formula Auditing group.

**Step-by-Step**

From the Student Data directory, open AUDIT.XLSX. Display worksheet formulas.

If necessary, display the Orders worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Formulas tab in the Ribbon. The Formulas tab is displayed.</td>
<td>Click Formulas</td>
</tr>
<tr>
<td>2. Select the Show Formulas button in the Formula Auditing group. The Show Formulas button is highlighted, worksheet column widths expand and cells containing formulas display the formula instead of the result.</td>
<td>Click Show Formulas</td>
</tr>
<tr>
<td>3. Select a cell containing a formula. The cell is selected, colors are applied to the references in the formula and correspondingly colored borders appear around the referenced cells.</td>
<td>Click cell E5</td>
</tr>
<tr>
<td>4. Select the Show Formulas button in the Formula Auditing group. The worksheet column widths return to their previous size, cells containing formulas display the result instead of the formula and highlighting is removed from the Show Formulas button.</td>
<td>Click Show Formulas</td>
</tr>
</tbody>
</table>
DISPLAYING/REMOVING DEPENDENT ARROWS

Discussion

You can locate all cells containing formulas that reference a specific cell. Cells that reference data in another cell are called the dependents of that cell. If a cell is referenced by multiple formulas, it has multiple dependents; if a cell is referenced by only one formula, it has a single dependent.

Displaying the dependents of a cell tells you which formulas are affected when you change the data in that cell. For example, if you are changing a shipping rate in an order entry worksheet, you may want to see which cells use that particular shipping rate in their formulas and, in turn, which orders will be affected by the change.

You can display just direct dependents of the active cell, or direct and indirect dependents. A direct dependent is a cell with a formula containing a direct reference to the active cell (for example, if cell C5 contains the formula =B5*10, cell C5 is a direct dependent of cell B5). An indirect dependent is a cell with a formula that refers to a another cell containing a formula that refers to the active cell, (continuing from the previous example, if cell D5 contains the formula =C5/2, cell D5 is an indirect dependent of cell B5, as well as a direct dependent of cell C5). Each intervening formula represents another level of dependency.

When you click the Trace Dependents button, Excel displays arrows pointing to the direct dependents of the active cell. If you click the button again, Excel displays additional arrows pointing to the first level of indirect dependents. Clicking the button again, displays arrows pointing to the second level of indirect dependents, and so on. If sound is not muted, your system will beep when there are no more additional levels.

Excel displays dependent cells with blue tracer arrows that lead from the active cell to each dependent. If a formula contains an error, the tracer arrows are red. If the dependent cell is on a different worksheet, a black arrow points to a worksheet symbol.
You can double-click a blue or red tracer arrow to go to the cell to which it points. To return, double-click the arrow again.

When you double-click a black arrow, the Go To dialog box opens and displays the address of the dependent cell. Select the displayed address and click OK to go to the cell. To return, press [Ctrl+G] to open the Go To dialog box, then press [Enter].

When you print a worksheet, displayed tracer arrows are printed.

You can click the left-hand part of the Remove Arrows button in the Formula Auditing group to remove all tracer arrows (both dependent and precedent). You can also click the arrow on the right-hand part of the Remove Arrows button and select Remove Dependent Arrows to remove tracer arrows one level at a time.
Procedures

1. Select the cell for which you want to view dependents.
2. Select the Formulas tab in the Ribbon.
3. To view direct dependents of the cell, select Trace Dependents in the Formula Auditing group.
4. To view the first level of indirect dependents, select Trace Dependents in the Formula Auditing group again.
5. To view additional levels of indirect dependents, select Trace Dependents in the Formula Auditing group, as necessary.
6. To remove all tracer arrows, select the left-hand part of the Remove Arrows button in the Formula Auditing group.

Step-by-Step

Display and remove dependent arrows.

If necessary, display the Orders worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell for which you want to view dependents. &quot;The cell is selected.&quot;</td>
<td>Click cell D5</td>
</tr>
<tr>
<td>2. Select the Formulas tab in the Ribbon. &quot;The Formulas tab is displayed.&quot;</td>
<td>Click Formulas</td>
</tr>
<tr>
<td>3. To view direct dependents of the cell, select the Trace Dependents button in the Formula Auditing group. Tracer arrows point to each directly dependent cell.</td>
<td>Click Trace Dependents</td>
</tr>
<tr>
<td>4. To view the first level of indirect dependents, select the Trace Dependents button in the Formula Auditing group again. Additional tracer arrows point to the first level of indirectly dependent cells.</td>
<td>Click Trace Dependents</td>
</tr>
</tbody>
</table>
### Steps

**5.** To view additional levels of indirect dependents, select the Trace Dependents button in the Formula Auditing group, as necessary. Additional tracer arrows point to the next level of indirectly dependent cells.

**6.** To remove all tracer arrows, select the left-hand part of the Remove Arrows button in the Formula Auditing group. All tracer arrows are removed from the worksheet.

### Practice Data

Click [Trace Dependents]

Click [Remove Arrows]

### Practice the Concept:

Select cell E8. Click the Trace Dependents button to display the direct dependents. Notice that four cells are direct dependents of the cell (cells F8, G8, H8, and E14). Click the Trace Dependents button again to display the first level of indirect dependents (cells F14 and H14). Click the Trace Dependents button again. Notice that no additional arrows are displayed; there are no more levels of dependent cells.

Click the arrow on the right-hand part of the Remove Arrows button and select Remove Dependent Arrows. Notice that Excel removes just one level of arrows. Select Remove Dependent Arrows again. Notice that Excel removes the next level of arrows.

### Displaying/Removing Precedent Arrows

#### Discussion

You can locate all the cells to which a formula refers. Cells that are referenced by a formula are called the precedents of the cell containing the formula. If a formula references multiple cells, it has multiple precedents, if a formula references only one cell, it has a single precedent.

Displaying the precedents of a formula tells you which cells provide data for the formula in the active cell. You can use this information to determine whether the formula refers to the correct data.

You can display just direct precedents of the formula in the active cell, or direct and indirect precedents. When a formula contains a reference to a cell or range, the cell or range is a direct precedent of the formula (for example, if cell D5 contains the formula =C5/2, cell C5 is a direct precedent of the formula in cell D5). When a formula contains a reference to a cell and the cell referenced contains a formula that references another cell, the second cell is an indirect precedent of the original formula (continuing from the previous example, if cell C5 contains the formula =B5*10, cell
B5 is an indirect precedent of cell D5, as well as a direct precedent of cell C5). Each intervening formula represents another level of precedence.

When you click the Trace Precedents button, Excel displays arrows pointing from the direct precedents of the formula to the active cell. Round dots on the tracer arrows indicate the position of the precedent cells. If you click the button again, Excel displays additional arrows pointing to the direct precedent cells; the dots on these arrows indicate the position of the first level of indirect precedents. Clicking the button again, displays arrows pointing to the second level of indirect precedents, and so on. If sound is not muted, your system will beep when there are no more additional levels.

Excel displays precedent cells with blue tracer arrows that lead from each precedent cell to the active cell. If a formula contains an error, a red tracer arrow indicates the precedent cell that is the direct cause of the error. If a precedent cell is on a different worksheet, a black arrow points from a worksheet symbol to the cell containing the formula that references the different sheet.

Displaying precedent cells

You can double-click a blue or red tracer arrow to go to the cell from which it points. To return, double-click the arrow again.

When you double-click a black arrow, the Go To dialog box opens and displays the address of the referenced cell or range. Select the displayed address and click OK to go to the cell or range. To return, press [Ctrl+G] to open the Go To dialog box, then press [Enter].
When you print a worksheet, displayed tracer arrows are printed.

You can click the left-hand part of the **Remove Arrows** button in the **Formula Auditing** group to remove all tracer arrows (both dependent and precedent). You can also click the arrow on the right-hand part of the **Remove Arrows** button and select **Remove Precedent Arrows** to remove tracer arrows one level at a time.

**Procedures**

1. Select the cell containing the formula for which you want to view the precedents.

2. Select the **Formulas** tab in the **Ribbon**.

3. To view direct precedents of the formula, select **Trace Precedents** in the **Formula Auditing** group.

4. To view the first level of indirect precedents, select **Trace Precedents** in the **Formula Auditing** group again.

5. To view additional levels of indirect precedents, select **Trace Precedents** in the **Formula Auditing** group, as necessary.

6. To remove all tracer arrows, select the left-hand part of the **Remove Arrows** button in the **Formula Auditing** group.

**Step-by-Step**

Display and remove precedent arrows.

If necessary, display the **Orders** worksheet.

<table>
<thead>
<tr>
<th><strong>Steps</strong></th>
<th><strong>Practice Data</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the cell containing the formula for which you want to view the precedents. <em>The cell is selected.</em></td>
<td>Click cell F5</td>
</tr>
</tbody>
</table>
Steps | Practice Data
--- | ---
2. Select the **Formulas** tab in the **Ribbon**. *The Formulas tab is displayed.*
| Click **Formulas**

3. To view direct precedents of the formula, select the **Trace Precedents** button in the **Formula Auditing** group. *Tracer arrows point from each direct precedent cell to the cell containing the formula.*
| Click **Trace Precedents**

4. To view the first level of indirect precedents, select the **Trace Precedents** button in the **Formula Auditing** group again. *Additional tracer arrows point from the first level of indirect precedents to the direct precedent cells.*
| Click **Trace Precedents**

5. To view additional levels of indirect precedents, select the **Trace Precedents** button in the **Formula Auditing** group, as necessary. *Additional tracer arrows point from the next level of indirect precedent cells.*
| Click **Trace Precedents**

6. To remove all tracer arrows, select the left-hand part of the **Remove Arrows** button in the **Formula Auditing** group. *All tracer arrows are removed from the worksheet.*
| Click **Remove Arrows**

**Practice the Concept:** Select cell H11. Click the **Trace Precedents** button to display the direct precedents. Notice that three cells are direct precedents of the formula in cell H11 (cells E11, F11, and G11). Notice also, the mixture of blue and red dots and lines. If these cells were not all in the same row, there would be three separate arrows pointing to cell H11, two blue ones and one red one. (The cell causing the error is the one with the red dot, cell F11).

Click the **Trace Precedents** button again to display the first level of indirect precedents. Notice that cell E11 has three direct precedents (cells A11, C11, and D11), which are the first level indirect precedents of cell H11. Notice also that cells F11 and G11 have arrows pointing from spreadsheet symbols; each of these cells has direct precedents in another sheet, which are also first level indirect precedents of cell H11.
Click the **Trace Precedents** button once more. Notice that cell C11 has a precedent in another sheet, this is also a second level indirect precedent of cell H11.

Click the arrow on the right-hand part of the **Remove Arrows** button and select **Remove Precedent Arrows**. Notice that Excel removes just one level of arrows. Select **Remove Precedent Arrows** again. Notice that Excel removes the next level of arrows. Select **Remove Precedent Arrows** one more time to remove the direct precedent arrows.

### Using the Trace Error Button

#### Discussion

Excel provides automatic error checking rules to help you correct possible errors in formulas. These predefined rules are designed to find common mistakes in formulas. The rules are described in the following table.

<table>
<thead>
<tr>
<th>Rule</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaluates to an error value</td>
<td>The formula does not use the expected syntax, arguments, or data types and the cell displays an error value such as #DIV/0, #NA, #NAME?, #NULL!, #NUM!, #REF!, or #VALUE!. Each error value has different causes and is resolved in different ways.</td>
</tr>
<tr>
<td>Text Date with 2-Digit Year</td>
<td>The cell contains a text date with a two-digit year that could be computed as the wrong century when used in formulas.</td>
</tr>
<tr>
<td>Number Stored as Text</td>
<td>Numbers formatted as text may not sort as expected.</td>
</tr>
<tr>
<td>Inconsistent Formula</td>
<td>The formula differs from the pattern of formulas in adjacent cells.</td>
</tr>
<tr>
<td>Formulas Omits Adjacent Cells</td>
<td>There is data in adjacent cells which is not included in the formula.</td>
</tr>
<tr>
<td>Unprotected Formula</td>
<td>The cell is unlocked. If worksheet protection is enabled, the formula will not be protected against inadvertent changes.</td>
</tr>
<tr>
<td>Formulas referring to empty cells</td>
<td>The formula references empty cells and may not produce the intended result.</td>
</tr>
<tr>
<td>Rule</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Inconsistent Calculated Column Formula</td>
<td>A cell in a calculated column in a table contains an entry which differs from the column formula.</td>
</tr>
</tbody>
</table>

Every cell with a formula identified by one of these rules as containing a potential error displays an error marker (green triangle) in its top-left corner. When you select a cell with an error marker, the Trace Error button appears beside the cell.

Pointing to the Trace Error button displays a ScreenTip that describes the possible error. Clicking the Trace Error button displays a menu headed with the name of the error type. The menu always contains options that let you ignore the error, edit the formula, or display the Error checking rules section on the Formulas page in the Excel Options dialog box. Additionally, depending on the type of error, it may also offer one or more of the following: additional help about the error; a suggested possible solution for automatically fixing the perceived problem; an option to show the individual steps in the calculation; or an option to display trace arrows indicating the source of the error.

Using the Trace Error button

The Trace Error button will not appear unless the Enable background error checking option is selected in the Error Checking section of the Formulas page in the Excel Options dialog box. The Error Checking section also lets you change the color used for the error marker triangle.
You should keep in mind that the error markers indicate possible errors, not definite errors. You should review the formula in the **Formula Bar** and use your judgment to decide whether the formula actually contains an error. If you decide that the formula is correct, you can use the **Ignore Error** option to remove the error marker from the cell. The error marker will not reappear for the cell unless you use the **Reset Ignored Errors** button in the **Error Checking** section of the **Formulas** page in the Excel Options dialog box.

### Procedures

1. Select any cell with a formula error marker in the top-left corner.
2. To view a description of the possible error, point to the **Trace Error** button beside the cell.
3. To display error checking options, select the expanded **Trace Error** button beside the active cell.
4. Select the desired command from the menu.

### Step-by-Step

Use the **Trace Error** button to correct a formula error.

If necessary, display the **Orders** worksheet.

<table>
<thead>
<tr>
<th><strong>Steps</strong></th>
<th><strong>Practice Data</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select any cell with a formula error marker in the top-left corner.</td>
<td>Click cell E7</td>
</tr>
<tr>
<td><em>The cell is selected and a Trace Error button appears beside the cell.</em></td>
<td></td>
</tr>
<tr>
<td>2. To view a description of the possible error, point to the Trace Error</td>
<td>Point to ✅</td>
</tr>
<tr>
<td><em>button beside the cell.</em></td>
<td></td>
</tr>
<tr>
<td>*The Trace Error button expands and a ScreenTip appears describing the</td>
<td></td>
</tr>
<tr>
<td><em>type of potential error identified.</em></td>
<td></td>
</tr>
</tbody>
</table>
Steps | Practice Data
--- | ---
3. To display error checking options, select the expanded Trace Error button beside the cell. A menu of available error checking options appears. | Click

4. Select the desired command from the menu. The error is corrected, the Trace Error button disappears, and the cell is no longer identified as containing an error. | Click Copy Formula from Above

Notice that the formula has been corrected to \(=C7*D7\).

**TRACING CELLS CAUSING ERRORS**

**Discussion**

If a formula results in an Error Value, such as #DIV/0, #NA, #NAME?, #NULL!, #NUM!, #REF!, or #VALUE!, you can determine whether the error is the result of an error in the formula itself or of an error in a preceding formula. For example, if the total cost formula in an order entry worksheet results in an Error Value, you can identify whether the problem is in the total cost formula itself or in a preceding formula referenced by the total, such as a price or shipping costs calculation. When you trace errors, Excel displays precedent tracer arrows pointing to the selected cell.

If only blue (or black) arrows point to the cell, then the problem is caused either by an incorrect reference in the current cell formula or by an incorrect data type in cells to which the formula refers (for example, text instead of numbers). You should examine the formula and its data to determine the exact problem.

Red tracer arrows indicate preceding formulas that result in an Error Value and Excel automatically selects the earliest preceding formula that produces an Error Value. You can then examine that formula and its data to determine the problem.

In addition to tracer arrows, the Trace Error button appears next to the active cell if it contains an error. Clicking the Trace Error button displays a list of error checking commands, including the reason the formula was flagged.
Tracing cells causing errors

You can also select the **Trace Error** command by clicking the arrow on the right-hand part of the **Error Checking** button in the **Formula Auditing** group on the **Formulas** tab.

You can remove the tracer arrows at any time by clicking the left-hand part of the **Remove Arrows** button in the **Formula Auditing** group on the **Formulas** tab.

**Procedures**

1. Select a cell that results in an **Error Value**.
2. Point to the **Trace Error** button beside the cell.
3. Select the expanded **Trace Error** button beside the cell.
4. Select **Trace Error** from the menu.
Step-by-Step

Trace cells causing an Error Value in the active cell.

If necessary, display the Orders worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select a cell that results in an Error Value. The cell is selected and the Trace Error button appears beside the cell.</td>
<td>Click cell H14</td>
</tr>
<tr>
<td>2. Point to the Trace Error button beside the cell. The Trace Error button expands.</td>
<td>Point to 🔄</td>
</tr>
<tr>
<td>3. Select the expanded Trace Error button beside the cell. A menu of available error checking options appears.</td>
<td>Click 🔄</td>
</tr>
<tr>
<td>4. Select Trace Error from the menu. Precedent tracer arrows appear on the worksheet and Excel selects the earliest preceding formula that results in an Error Value.</td>
<td>Click Trace Error</td>
</tr>
</tbody>
</table>

The red arrows indicate that the Error Value in cell H14 is caused by the Error Value in cell H11, which in turn is caused by the Error Value in cell F11. Cell F11 only has blue arrows pointing to it and is the starting point of the error; consequently, Excel has selected cell F11.

Examining both the formula in the cell and the data it refers to, the VLOOKUP formula in cell F11 refers to a two column table ("Lookup Tables"!$A$14:$B$23) but is trying to obtain its result from column 3 of the table; the formula should use a 2 rather than a 3 as its last argument.

Edit the VLOOKUP formula in the Formula Bar, change the final 3 to a 2 and press [Enter]. All worksheet cells are now correctly computed and the red tracer arrows no longer appear. Remove any remaining tracer arrows.
**Using the Evaluate Formula Button**

**Discussion**

When a formula contains nested expressions (such as an IF function that uses another function to calculate its logical_test argument) or a formula refers to a cell that contains a calculation, it can be difficult to understand how the final result is calculated. The Evaluate Formula dialog box lets you step through a formula, evaluating each expression in turn, to observe how the result is calculated.

When the Evaluate Formula dialog box opens, the formula is displayed in an Evaluation box with the first expression underlined. Clicking the Evaluate button in the dialog box displays the result of the underlined expression and the next expression is then underlined. Each click of the Evaluate button replaces the currently underlined expression with its resulting value, showing step-by-step how the final result of the formula is calculated.

Where the selected expression refers to a cell, a Step In button lets you display the cell entry in a separate Evaluation box, if desired. If the cell simply contains a value, the value is displayed in the separate Evaluation box. If the cell contains another formula, the formula is displayed in the separate Evaluation box and you can evaluate each of its expressions to determine the result of its formula. The Step Out button closes the separate Evaluation box and returns the cell value or the result of its formula to the main formula. You are not obliged to step in to a cell; you can use the Evaluate button to obtain its result and proceed to the next expression, if preferred.

If a formula in a cell you have stepped into contains a reference to another cell, you can use the Step In button to open yet another separate Evaluation box. You can Step In to as many levels as needed to view and evaluate precedent formulas in other cells. After evaluating each precedent formula, you then Step Out of the separate Evaluation box to return its result to the previous level until only one Evaluation box remains. When you have finished evaluating all expressions, the final result of the formula is displayed.
When you use the **Step In** button to display a precedent formula in a separate **Evaluation** box, you are not obliged to evaluate each individual expression within the precedent formula. You can click the **Step Out** button at any time to close the separate **Evaluation** box and return the result of the precedent formula to the originating formula.

When you **Step In** to a cell reference in a formula, Excel also selects the cell in the worksheet. If you are using **Evaluate Formula** to trace a problem and you discover that the problem is in the cell you have stepped into, you can **Close** the Evaluate Formula dialog box to edit the selected cell.

The Evaluate Formula feature can also be useful for relatively simple formulas. Because it shows the order in which expressions are calculated, it can help you understand when you need to insert brackets in a formula to control the order of calculation.
Procedures

1. Select the cell containing the formula to be evaluated.
2. Select the **Formulas** tab in the **Ribbon**.
3. Select \[\text{Evaluate Formula}\] in the **Formula Auditing** group.
4. To replace the underlined expression in the formula with its result, select \[\text{Evaluate}\].
5. To display the contents of an underlined cell reference in a separate **Evaluation** box, select \[\text{Step In}\].
6. Select \[\text{Step In}\] or \[\text{Step Out}\] for each expression within a precedent formula, as desired.
7. To return the result of a precedent formula to the original formula, select \[\text{Step Out}\].
8. Select \[\text{Step In}\], \[\text{Step Out}\], or \[\text{Restart}\] for each expression in turn, as desired, until the final result is displayed.
9. Select \[\text{Restart}\] to evaluate the formula again or \[\text{Close}\] to end the evaluation.

Step-by-Step

Use the **Evaluate Formula** button.

Display the **Orders** worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the cell containing the formula to be evaluated.  
*The cell is selected.* | Click cell E5 |
| 2. Select the **Formulas** tab in the **Ribbon**.  
*The **Formulas** tab is displayed.* | Click **Formulas** |
### Steps

| 3. | Select the **Evaluate Formula** button in the **Formula Auditing** group.  
   *The Evaluate Formula dialog box opens with the formula displayed in the Evaluation box and the first expression in the formula is underlined.* |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4. | To replace the underlined expression with its result, select the **Evaluate** button.  
   *The result of the selected expression appears in the formula and the next expression in the formula is underlined.* |
| 5. | To evaluate each expression in turn, select the **Evaluate** button repeatedly.  
   *Each underlined expression is replaced by its result until the final result is calculated.* |
| 6. | Select **Restart** to evaluate the formula again or **Close** to end the evaluation.  
   *The cell formula is redisplayed in the Evaluation box and the first expression in the formula is underlined.* |
| 7. | To display the contents of an underlined cell reference in the formula, select the **Step In** button.  
   *The selected cell reference is highlighted in blue in the first Evaluation box, the cell entry is displayed in a second Evaluation box and, if the entry is a formula, the first expression in the formula is underlined.* |
| 8. | Select **Evaluate** or **Step In** for each expression within the precedent formula, as desired.  
   *The expressions are calculated accordingly until the result of the precedent formula appears in the second Evaluation box.* |

### Practice Data

<table>
<thead>
<tr>
<th>Click <strong>Evaluate Formula</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Click <strong>Evaluate</strong></td>
</tr>
<tr>
<td>Follow the instructions shown below the table before continuing on to the next step</td>
</tr>
<tr>
<td>Click <strong>Restart</strong></td>
</tr>
<tr>
<td>Click <strong>Step In</strong></td>
</tr>
</tbody>
</table>

Follow the instructions shown below the table before continuing on to the next step.
Steps | Practice Data
---|---
9. Select **Step Out** to return to the original formula. The second **Evaluation** box closes, the italicized result of the precedent formula replaces the blue highlighted expression in the first **Evaluation** box and the next expression in the formula is underlined. | Click **Step Out**

10. Select **Evaluate, Step In** or **Step Out** for each expression in turn, as desired. Each underlined expression is calculated or stepped into accordingly until the final result of the formula is displayed. Follow the instructions shown below the table before continuing on to the next step.

11. Select **Restart** to evaluate the formula again or **Close** to end the evaluation. The **Evaluate Formula** dialog box closes. | Click **Close**

Click the **Evaluate** button to replace the underlined D5 expression with its result. Notice that the result (10) appears in italics and that the next stage in the calculation order (63.99*10) is underlined. Click the **Evaluate** button again. Notice that the underlined expression is calculated and the final result of the formula ($639.90) is displayed.

*Return to the table and continue on to the next step (step 6).*

Click the **Evaluate** button to replace the underlined B5 expression with its result. Notice that the result (“ST-2472”) appears in italics and that the whole precedent formula (VLOOKUP(“ST-2472”, ’Lookup Tables’!$A$3:$B$10,2)) is underlined. Click **Evaluate** once more. The precedent formula is calculated and its result is displayed in the second **Evaluation** box.

*Return to the table and continue on to the next step (step 9).*

Click the **Evaluate** button to replace the underlined D5 expression with its result. Notice that the result (10) appears in italics and that the next stage in the calculation order (63.99*10) is underlined. Because D5 simply contains a value there is no advantage to stepping in to this expression. If you stepped in to the expression, the value 10 would be shown in a separate **Evaluation** box and you would need to step out to return the value to the main **Evaluation** box. Click the **Evaluate** button again. Notice that the underlined expression is calculated and the final result of the formula ($639.90) is displayed.

*Return to the table and continue on to the next step (step 11).*
**USING THE ERROR CHECKING BUTTON**

**Discussion**

The **Error Checking** button in the **Formula Auditing** group on the **Formulas** tab helps you locate cells containing certain kinds of possible formula errors. It locates these cells based upon rules designed to find common mistakes in formulas. These rules include correcting dates entered with a 2-digit year, numbers entered as text, and formulas that are inconsistent for the region, as well as omitted cells, and formulas referring to empty cells. The error check also identifies unlocked cells containing formulas.

The **Error Checking** button is an efficient method of checking the entire sheet for potential errors. The Error Checking dialog box starts at the currently selected cell and steps from one potential error to the next. The Error Checking dialog box information changes, depending on the type of error encountered. You can automatically correct the formula error, ask for help in correcting the error, ignore the error, or edit the error in the Formula Bar. Excel notifies you when all the identified errors in the sheet have been corrected or ignored.

![Using the Error Checking dialog box](image_url)

The formula checking rules can be enabled or disabled as desired. Selecting the **Options** button in the Error Checking dialog box opens the Excel Options dialog box on the **Formulas** page; you can then select or deselect options in the **Error Checking Rules** section as desired.
You should keep in mind that the Error Checking dialog box is highlighting possible errors, not definite errors. You should review the formula in the Formula Bar and use your judgment to decide whether the formula actually contains an error. If you decide that the formula is correct, you can use the Ignore Error option to remove the error marker from the cell and proceed to the next potential error. The error marker will not reappear for the cell unless you use the Reset Ignored Errors button in the Error Checking section of the Formulas page in the Excel Options dialog box.

Similarly, you should not blindly accept the suggested solutions offered in the Error Checking dialog box, such as Copy Formula from Above. If you are not sure whether copying the formula will produce the correct result, click on the cell above and view the formula in the Formula Bar. The Error Checking dialog box will pause. When you are ready to continue, click the Resume button in the dialog box.

Procedures

1. Select the Formulas tab in the Ribbon.
2. Select the left-hand part of the Error Checking button in the Formula Auditing group.
3. Select the desired button in the Error Checking dialog box.
4. Continue reviewing errors in the worksheet.
5. To manually correct an error, select the Edit in Formula Bar button.
6. Make corrections to the formula as needed.
7. Select the Resume button in the Error Checking dialog box to continue reviewing errors.
8. When Excel notifies you that the error check is complete, select OK.
Step-by-Step

Use the Error Checking feature to identify and correct errors.

Display the **EastWest** worksheet. Select cell A1.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>Formulas</strong> tab in the Ribbon. The <strong>Formulas</strong> tab is displayed.</td>
<td>Click <strong>Formulas</strong></td>
</tr>
<tr>
<td>2. Select the left-hand part of the <strong>Error Checking</strong> button in the <strong>Formula Auditing</strong> group. The Error Checking dialog box opens and the first cell containing a potential error is selected.</td>
<td>Click <img src="image" alt="Error Checking" /></td>
</tr>
<tr>
<td>3. Select the desired button in the Error Checking dialog box. The error is corrected, and the next cell containing an error is selected.</td>
<td>Click <strong>Convert XX to 20XX</strong></td>
</tr>
<tr>
<td>4. Select the desired button in the Error Checking dialog box. The error is corrected, and the next cell containing an error is selected.</td>
<td>Click <strong>Convert to Number</strong></td>
</tr>
<tr>
<td>5. Continue reviewing errors in the worksheet. The errors are corrected.</td>
<td>Follow the instructions shown below the table before continuing on to the next step</td>
</tr>
<tr>
<td>6. To manually correct an error, select the <strong>Edit in Formula Bar</strong> button. The formula is selected, and the insertion point appears in the Formula Bar.</td>
<td>Click <strong>Edit in Formula Bar</strong></td>
</tr>
<tr>
<td>7. Make corrections to the formula as needed. The corrections are made to the formula.</td>
<td>Edit the formula to read =SUM(E14:E18)</td>
</tr>
<tr>
<td>8. Select the <strong>Resume</strong> button in the Error Checking dialog box to continue reviewing errors. A Microsoft Office Excel message box opens, informing you that the error check is complete.</td>
<td>Click <strong>Resume</strong></td>
</tr>
</tbody>
</table>
For the Inconsistent Formula error in cell H7, the suggested solution is to copy the formula from the cell above. Move the dialog box, if necessary, click cell H6 in the worksheet and view the formula in the Formula Bar. Notice that in this instance, copying the formula will produce the result you want. Click the Resume button in the dialog box, then click the Copy Formula from Above button.

For the Inconsistent Formula error in cell E9, click cell D9 and view the formula in the Formula Bar. Click Resume, then click Copy Formula from Left.

For the Unprotected Formula error in cell H9, click the Lock Cell button.

For the Divide by Zero Error in cell I15, move the dialog box, if necessary, to view cells I14:I18. Notice that there are similar errors in cells I16 through I18; the problem was caused by not using an absolute reference in the formula in cell I14 before copying it to cells I15:I18. Click cell I14. Click in the Formula Bar at the end of the formula, press [F4] to change the H19 reference in the formula into an absolute reference (SH$19) then click the Enter button (green tick) in the Formula Bar. Use the Fill Handle at the bottom-right corner of cell I14 to copy the formula into cells I15:I18. Select the Resume button in the Error Checking dialog box.

For the Formula Omits Adjacent Cells error in cell D19, click the Update Formula to Include Cells button, then click cell D19 and view the formula in the Formula Bar to check that it now refers to the correct range. Click the Resume button.

Return to the table and continue on to the next step (step 6).
worksheet; you can even monitor cells from another workbook. To add cells from another workbook to the Watch Window, that workbook must be open.

The Watch Window is a task pane. You can resize it and drag it to any desired position in the window or dock it along any edge. When the Watch Window is docked, you can drag the border between the Watch Window and the worksheet to adjust the relative sizes of the Watch Window and the worksheet. You can also resize the columns within the Watch Window and you can work in the worksheet while the Watch Window is open.

You can delete cells from the Watch Window by first selecting the cell you want to delete in the Watch Window; then you can either select the **Delete Watch** button or press the `[Delete]` key.

Double-clicking an entry in the Watch Window selects the corresponding cell in the worksheet.

Monitored cells are not removed from the Watch Window when you close it. While this is useful when monitoring the same cells over a period of time, if you no longer wish to monitor the selected cells, remember to delete them from the Watch Window before closing it.
Procedures

1. Select the **Formulas** tab in the **Ribbon**.

![Watch Window](image1)

2. Select **Watch Window** in the **Formula Auditing** group.

3. Select the cell or range you want to add to the Watch Window.

4. Select **Add Watch...** in the Watch Window.

5. Select **Add**.

6. To delete an entry from the Watch Window, select the entry in the Watch Window.

7. Select **Delete Watch** in the Watch Window.

8. To close the Watch Window, select **Watch Window** in the **Formula Auditing** group.

Step-by-Step

Use the Watch Window.

Display the **Orders** worksheet.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the **Formulas** tab in the **Ribbon**.  
*The Formulas tab is displayed.* | Click **Formulas** |
| 2. Select the **Watch Window** button in the **Formula Auditing** group.  
*The Watch Window window opens.* | Click **Watch Window** |
| 3. Select the cell or range you want to add to the Watch Window.  
*The range is selected.* | Drag to select A6:H6 |
### Steps

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. Select the <strong>Add Watch</strong> button in the Watch Window.</td>
<td>Click <img src="image" alt="Add Watch" /></td>
</tr>
<tr>
<td><em>The Add Watch dialog box opens, with the selected range displayed.</em></td>
<td></td>
</tr>
<tr>
<td>5. Select <strong>Add</strong>.</td>
<td>Click <img src="image" alt="Add" /></td>
</tr>
<tr>
<td><em>The Add Watch dialog box closes, and the selected cells appear in the Watch Window.</em></td>
<td></td>
</tr>
<tr>
<td>6. To delete an entry from the Watch Window, select the entry in the Watch Window.</td>
<td>Click the cell B6 entry in the Watch Window</td>
</tr>
<tr>
<td><em>The entry is highlighted.</em></td>
<td></td>
</tr>
<tr>
<td>7. Select the <strong>Delete Watch</strong> button in the Watch Window.</td>
<td>Click <img src="image" alt="Delete Watch" /></td>
</tr>
<tr>
<td><em>The entry is removed from the Watch Window.</em></td>
<td></td>
</tr>
<tr>
<td>8. To close the Watch Window, select the <strong>Watch Window</strong> button in the <strong>Formula Auditing</strong> group.</td>
<td>Click <img src="image" alt="Watch Window" /></td>
</tr>
<tr>
<td><em>The Watch Window closes.</em></td>
<td></td>
</tr>
</tbody>
</table>

### Practice the Concept: Open the Watch Window. Notice that the entries you are watching were not removed from the Watch Window when you closed it.

If necessary, resize the Watch Window to display all the formula information. Notice that both cell formulas and their results are displayed.

Reposition the Watch Window as necessary, then select the range A10:H10 and add it to the Watch Window. Delete the B10 entry from the Watch Window.

Select the **Lookup Tables** sheet. Dock the Watch Window along the bottom edge of the worksheet and drag the border between the worksheet and the Watch Window, as necessary, until you can just see all the entries in the Watch Window.

In the worksheet, change cell E7 (the Discount Scale for SportsCity) to 2. Notice in the Watch Window that the values for cells G6 (the Discount amount) and H6 (the Total Cost of the order) change accordingly. *(Hint: When you are trying to monitor changes in more than one cell, you may find it useful to use [Ctrl+Z] (undo) and [Ctrl+Y] (redo) to toggle the value you have changed back and forth while you view the values in the Watch Window).*

In the worksheet, change cell B8 to $200. Notice in the Watch Window that the values for cells C6, E6, G6, H6, C10, E10, F10, and H10 all change.
Double-click the D6 entry in the Watch Window. Notice that Excel switches to the Orders worksheet and selects the corresponding cell.

Select the Delete Watch button in the Watch Window. Notice that the selected entry is removed from the Watch Window.

Select all the entries in the Watch Window by clicking the first entry, holding the [Shift] key, and clicking the last entry. Then, use the Delete Watch button to delete them all. Close the Watch Window.
Close AUDIT.XLSX.
EXERCISE

USING AUDITING TOOLS

Task

Use the Formula Auditing tools.

1. Open AUDITEX.XLSX.
2. Display the Bonuses worksheet, if necessary.
3. Display the Formulas tab.
4. Display all dependent cells for cell E8. Then, remove one level of dependent arrows.
5. Display all precedent cells for cell I14. Then, remove one level of precedent arrows.
6. Remove all displayed tracer arrows.
7. Select cell F14, which contains an error. Use the Trace Error button that appears beside the cell to display its tracer arrows. (Notice that the error is caused by an incorrect reference in the formula in cell F9. Notice also that similar errors occur in cells F10 through F13. All these errors were caused by copying the formula in cell F8 without using an absolute reference.)
8. Select cell F8 and correct the formula to =E8/E$14.
9. Copy the formula in cell F8 to the range F9:F13 to correct this series of errors.
10. Display the Checking worksheet. Select cell B14 and point to the Trace Error button beside the cell to display a ScreenTip describing the error. Read the formula in the Formula Bar to understand the problem.
11. Click the Trace Error button and use the suggested solution to correct the formula. Read the formula in the Formula Bar again to check that the suggested solution produced the desired result.
12. Select cell A1. Then use the Error Checking button in the Formula Auditing group to correct the errors in cells B4, B5, E11, and D14.
13. Evaluate all expressions in the formula in cell I9. Use Step In and Step Out, as desired, to view cells referenced by the main formula.
15. Display the **Bonuses** worksheet. Change the Bonus percentage in cell B5 to 7%. Observe the changes in the Watch Window.

16. Delete all cells from the Watch Window, then close the Watch Window.

17. Close the workbook without saving it.
LESSON 10 - USING MACROS

In this lesson, you will learn how to:

- Define macros
- Change macro security settings
- Add a folder as a Trusted Location
- Open a workbook containing macros
- Run a macro
- Use a shortcut key
- Display the Developer tab
- Open the Visual Basic Editor window
- Use the Visual Basic Editor window
DEFINING MACROS

Discussion

A macro is a program you can create to automatically perform frequently used operations. It contains all the commands, mouse movements, and user actions necessary to complete a task. Macros can save you considerable time because you can use them to automate repetitive, time-consuming tasks. You can use macros for simple tasks you perform frequently or for complex tasks that require consistency.

Excel macros are written in the Visual Basic for Applications (VBA) programming language. The series of commands used in a macro is known as a procedure. Each procedure is given a name that is used to execute the macro.

Each procedure is saved in a device called a module. This module is attached to the worksheet in which the macro is stored. A module is hidden and cannot be seen in the normal window view. You must use the Visual Basic Editor to select a module in order to view and edit the macro commands.

All procedures begin with the word Sub (an abbreviation of subroutine), followed by the name of the macro and an open and a closing parenthesis, and end with the words End Sub. Commands entered between these two lines are macro statements. When you record a procedure, Visual Basic translates the menu choices and keystrokes into the macro statements. Comments may also be inserted into a procedure to add clarity. If you know Visual Basic, you can write a procedure yourself instead of using the macro recorder.

When you save a workbook containing macros, Excel prompts you to save it as a Macro-Enabled Workbook file type. A Macro-Enabled Workbook has all the same features as a standard workbook but can also store macros. The workbook file is saved with the file extension .XLSM instead of .XLSX. This lets you see when you are about to open a file that may contain macros; if the file extension is .XLSX, you know that the file cannot contain macros. If your copy of Windows is set not to show file extensions, you can still differentiate between Macro-Enabled Workbooks and standard workbooks: the file icon for a Macro-Enabled Workbook has an Exclamation Mark (!) on the icon.

A macro can be stored in a specific workbook or in the Personal Macro Workbook. Macros stored in a specific workbook can only be used when that workbook is open. The Personal Macro Workbook automatically opens when you start Excel. As a result, macros stored in the Personal Macro Workbook are available to all files.
CHANGING MACRO SECURITY SETTINGS

Discussion

When you open a workbook containing macros, you could unknowingly install a virus on your computer. The powerful VBA language can run almost any command on your computer, so it is possible to write a macro containing malicious code that activates itself when a workbook is opened and spreads a virus on your computer. Various security settings in Excel provide protection against malicious macros. The default setting is to disable unknown macros until you indicate that you trust the macros.

You can change the default settings and choose how you want Excel to respond to macros on the Macro Settings page in the Trust Center, which is accessed from the Trust Center page in the Excel Options dialog box.

The Trust Center in Excel also contains a list of Trusted Locations. These are folders on your computer’s hard drive (such as the Templates folder) that are treated as trusted sources for opening files. Any workbook stored in these locations is automatically trusted and macros contained in them will not be disabled when you open the workbook. You can add more folders to the list of Trusted Locations and store your own macro-driven workbooks in those folders so that you can run them without interference.

The default setting is Disable all macros with notification, which checks every Excel workbook you open that is not stored in a Trusted Location to see if it contains macros. If it finds any macros, it disables them before opening the workbook and a Message Bar appears above the workbook containing a Security Warning notifying you that macros have been disabled together with an Options button. Clicking the Options button opens the Microsoft Office Security Options dialog box which gives you a number of choices depending on the situation, such as whether the macros have a valid Digital Signature (an electronic, encrypted, secure stamp of authentication). It always contains the option Enable this content, which lets you enable the macros contained in the workbook. The macros are only enabled until you close the workbook; next time you open the workbook the Security Warning will reappear.

Developers of commercially available Excel workbooks containing macros can register with a Certificate Authority which issues Digital Certificates. The Certificate Authority keeps track of who is assigned a certificate and digitally signs certificates to verify their validity. It also tracks which certificates have been revoked or have expired. The developer then attaches this Digital Signature to workbooks containing macros to authenticate them.

In the Macro Settings section of the Trust Center, you can choose to Disable all macros except digitally signed macros. This option does not automatically accept all macros with Digital Signatures but only those Digital Signatures that you choose to add to the list of Trusted Publishers in the Trust Center.
You can also choose to **Disable all macros without notification**. This option disables all macros not stored in a **Trusted Location** (including those with trusted digital signatures) but does not display the **Security Warning** in the **Message Bar**.

The final option in the **Macro Settings** section lets you **Enable all macros**. This option allows all macros to run without any security checking and is not generally recommended.

The **Developer Macro Settings** section on the **Macro Settings** page contains just one option: **Trust access to the VBA project object model**. By default, the option is not selected; this is to make it more difficult for unauthorized programs to build self-replicating code to propagate viruses. To allow macro code to access the VBA object model programmatically, the user running the code must explicitly grant access by enabling this option.

![Changing macro security settings](image)

When you open the **Trust Center** from the Excel Options dialog box to make changes to the **Macro Settings**, the changes apply to macros in Excel only; the macro settings are not changed for other Office programs.
### Procedures

1. Select the **Office** button.
2. Select **Excel Options**.
3. Select **Trust Center** from the list in the left-hand pane.
4. Select **Trust Center Settings** in the right-hand pane.
5. Select **Macro Settings** from the list in the left-hand pane.
6. Select the desired option in the **Macro Settings** section of the page.
7. Select **OK**.
8. Select **OK**.

### Step-by-Step

Change macro security settings.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the **Office** button.  
*The Office menu opens.* | Click |
| 2. Select the **Excel Options** button.  
*The Excel Options dialog box opens.* | Click **Excel Options** |
| 3. Select **Trust Center** from the list in the left-hand pane.  
*A page with information about privacy and security appears in the right-hand pane.* | Click **Trust Center** |
| 4. Select the **Trust Center Settings** button in the right-hand pane.  
*The Trust Center dialog box opens.* | Click **Trust Center Settings** |
| 5. Select **Macro Settings** from the list in the left-hand pane.  
*The Macro Settings page is displayed in the right-hand pane.* | Click **Macro Settings** |
6. Select the desired option in the **Macro Settings** section of the page.
   The desired option is selected.

   Practice Data: Click **Disable all macros except digitally signed macros**

7. Select **OK**.
   The selection is confirmed and the Trust Center dialog box closes.

   Practice Data: Click **OK**

8. Select **OK**.
   The Excel Options dialog box closes.

   Practice Data: Click **OK**

---

**Practice the Concept:** Open the Trust Center again and change the Macro Settings to **Disable all macros with notification**.

---

### Adding a Folder as a Trusted Location

**Discussion**

The Macro-Enabled Workbooks that you save containing macros that you have created do not have a Digital Signature. When you open your Macro-Enabled Workbooks, the default security settings for macros will disable the macros and display a Security Warning in the Message Bar.

You could change the macro security settings to enable all macros but this, of course, creates a security risk. A far better method is to store all your Macro-Enabled Workbooks in a specific folder and add the folder to your **Trusted Locations** list. Files stored in your Trusted Locations are not security-checked by Excel and open without any annoying prompts.

When you designate a folder as a Trusted Location, you can specify that subfolders within the designated folder should also be seen as **Trusted Locations**. This makes it easy to have a structured folder system for storing your macros. If you have Macro-Enabled Workbooks stored in folders in different locations, you can add each of those folders to the Trusted Locations list.

You can also add a description to a folder when designating it as a Trusted Location to remind yourself about the contents of the folder. The description is visible only in the Trusted Locations list.
Adding a folder as a Trusted Location

To remove a folder from the Trusted Locations list, open the Trust Center and display the Trusted Locations page. In the list of Trusted Locations, select the folder you wish to remove and click the Remove button.

To make changes to a folder in the Trusted Locations list, open the Trust Center and display the Trusted Locations page. In the Trusted Locations list, select the folder you wish to change and click the Modify button. You can designate a different folder by changing the path, allow or disallow subfolders, and change the description.

You can also disable all Trusted Locations. In the Trusted Locations page in the Trust Center, select the Disable all Trusted Locations. Only files signed by Trusted Publishers will be trusted option.

Preferably, the folders you add to the Trusted Locations list should be on the local hard disk of your computer. If the folder resides on a network drive, it should be a folder with limited access. You should not designate a public folder on a network drive as a Trusted Location. The more people who have access to the folder, the more danger there is that someone could tamper with your macros.
Procedures

1. Select the Office button.
2. Select Excel Options.
3. Select Trust Center from the list in the left-hand pane.
4. Select Trust Center Settings... in the right-hand pane.
5. Select Trusted Locations from the list in the left-hand pane.
6. Select Add new location...
7. Select Browse...
8. Select the double arrow at the left of the Address bar.
9. Select the drive in which the folder you want to add to the Trusted Locations list is located.
10. Open the folder you want to add to the Trusted Locations list.
11. Select the left-hand part of the OK button.
12. Select Subfolders of this location are also trusted, if required.
13. Select the Description box.
14. Type a description of the folder or its contents.
15. Select OK.
16. If the folder you added resides on a network drive, select Allow Trusted Locations on my network (not recommended).
17. Select OK.
18. Select OK.

Step-by-Step

Add a folder to the Trusted Locations list.
### Steps

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>Office</strong> button. &lt;br&gt;The <strong>Office</strong> menu opens.</td>
<td>Click</td>
</tr>
<tr>
<td>2. Select the <strong>Excel Options</strong> button. &lt;br&gt;The <strong>Excel Options</strong> dialog box opens.</td>
<td>Click Excel Options</td>
</tr>
<tr>
<td>3. Select <strong>Trust Center</strong> from the list in the left-hand pane. &lt;br&gt;A page with information about privacy and security appears in the right-hand pane.</td>
<td>Click Trust Center</td>
</tr>
<tr>
<td>4. Select the <strong>Trust Center Settings</strong> button in the right-hand pane. &lt;br&gt;The <strong>Trust Center</strong> dialog box opens.</td>
<td>Click Trust Center Settings...</td>
</tr>
<tr>
<td>5. Select <strong>Trusted Locations</strong> from the list in the left-hand pane. &lt;br&gt;The <strong>Trusted Locations</strong> page is displayed in the right-hand pane.</td>
<td>Click Trusted Locations</td>
</tr>
<tr>
<td>6. Select the <strong>Add new location</strong> button. &lt;br&gt;The Microsoft Office <strong>Trusted Location</strong> dialog box opens.</td>
<td>Click Add new location...</td>
</tr>
<tr>
<td>7. Select the <strong>Browse</strong> button. &lt;br&gt;The <strong>Browse</strong> dialog box opens.</td>
<td>Click Browse...</td>
</tr>
<tr>
<td>8. Select the double arrow at the left of the <strong>Address bar</strong>. &lt;br&gt;A list of available drives and common folders appears.</td>
<td>Click &lt;&lt;</td>
</tr>
<tr>
<td>9. Select the drive in which the folder you want to add to the <strong>Trusted Locations</strong> list is located. &lt;br&gt;A list of available folders appears.</td>
<td>Click the student data drive</td>
</tr>
<tr>
<td>10. Open the folder you want to add to the <strong>Trusted Locations</strong> list. &lt;br&gt;The folder opens.</td>
<td>Double-click to open the student data folder</td>
</tr>
<tr>
<td>11. Select the left-hand part of the <strong>OK</strong> button. &lt;br&gt;The <strong>Browse</strong> dialog box closes and the selected drive and folder appear in the <strong>Path</strong> box in the Microsoft Office <strong>Trusted Location</strong> dialog box.</td>
<td>Click OK</td>
</tr>
</tbody>
</table>
Lesson 10 - Using Macros

Steps | Practice Data
--- | ---
12. Select the **Subfolders of this location are also trusted** option, if required. *The option is selected.* | Click ☑ Subfolders of this location are also trusted to select it.
13. Select the **Description** box. *The insertion point appears in the Description box.* | Click in the **Description** box.
14. Type a description of the folder or its contents. *The text appears in the Description box.* | Type **My macros**
15. Select **OK.** *The Microsoft Office Trusted Location dialog box closes and the folder name appears in the User Locations section of the Trusted Locations list.* | Click **OK**
16. If the folder you added resides on a network drive, select the **Allow Trusted Locations on my network (not recommended)** option. *The option is selected, if necessary.* | Click ☑ Allow Trusted Locations on my network (not recommended) to select it, if necessary.
17. Select **OK.** *The Trust Center dialog box closes.* | Click **OK**
18. Select **OK.** *The Excel Options dialog box closes.* | Click **OK**

**Practice the Concept:** Open the **Trust Center** and display the **Trusted Locations** page. Select the student data folder in the list and click the **Remove** button.

## OPENING A WORKBOOK CONTAINING MACROS

### Discussion

The default macro security setting in Excel checks every workbook you open that is not stored in a **Trusted Location** to see if it contains macros. If it finds any macros, it disables them before opening the workbook and a **Message Bar** appears above the workbook containing a **Security Warning** notifying you that macros have been disabled together with an **Options** button.

Clicking the **Options** button opens the Microsoft Office Security Options dialog box which contains further information about the workbook, such as whether it has a valid **Digital Signature** (an electronic, encrypted, secure stamp of authentication). The
dialog box always contains at least two options. If you select **Help protect me from unknown content (recommended)**, the macros will remain disabled. If you select **Enable this content**, the macros contained in the workbook are enabled. The macros are only enabled until you close the workbook, next time you open the workbook the Security Warning will reappear.

The Microsoft Office Security Options dialog box displays a variety of different warnings and notes depending on whether or not the workbook has a digital signature and whether the digital signature is valid or expired.

If the workbook has a valid and current **Digital Signature**, a third option appears in the dialog box: **Trust all documents from this publisher**. Selecting this option adds the publisher to your **Trusted Publishers** list in the **Trust Center**.

---

**Opening a workbook containing macros**

![Opening a workbook containing macros](image)

- The **More Information** link in the Microsoft Office Security Options dialog box opens Excel Help and displays a page which describes the security options in detail.

- The **Open the Trust Center** link in the Microsoft Office Security Options dialog box opens the **Trust Center**, where you can view the list of **Trusted Publishers** and remove publishers from the list, if required. You can also view and edit the list of **Trusted Locations**, change security settings for macros and control whether the **Message Bar** is displayed when content is blocked.
In order for this topic to work properly, macro security must be set to the default setting. To check and, if necessary, set the correct option, select the Office button, the Excel Options button, the Trust Center page, and the Trust Center Settings button. Select the Macro Settings section, then select the Disable all macros with notification option and OK twice.

**Procedures**

1. Select the Office button.
2. Select \[image\] from the Office menu.
3. Select the double arrow \[image\] at the left of the Address bar.
4. Select the drive in which the workbook you want to open is located.
5. Open the folder in which the workbook you want to open is located.
6. Select the file name of the workbook you want to open.
7. Select the left-hand part of the Open button \[image\].
8. Select \[image\] in the Message Bar.
9. Select the Enable this content option.
10. Select \[image\].

**Step-by-Step**

Open a workbook containing macros.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the Office button.  
  *The Office menu opens.*                                                                 | Click \[image\]       |
| 2. Select Open from the Office menu.  
  *The Open dialog box opens.*                                                                 | Click \[image\] Open   |
**Steps** | **Practice Data**
--- | ---
3. Select the double arrow at the left of the **Address bar**. 
   *A list of available drives and common folders appears.* | **Click <**
4. Select the drive in which the workbook you want to open is located. 
   *A list of available folders appears.* | **Click the student data drive**
5. Open the folder in which the workbook you want to open is located. 
   *The contents of the folder appear.* | **Double-click to open the student data folder**
6. Select the file name of the workbook you want to open. 
   *The file name is highlighted in the list and appears in the File name box.* | **Scroll as necessary and click CITY56.XLSM**
7. Select the left-hand part of the **Open** button. 
   *The Open dialog box closes, the workbook opens and the Message Bar opens above the workbook with a Security Warning displayed.* | **Click**
8. Select the **Options** button in the Message Bar. 
   *The Microsoft Office Security Options dialog box opens.* | **Click Options...**
9. Select the desired option. 
   *The desired option is selected.* | **Click [Enable this content]**
10. Select **OK**. 
    *The Microsoft Office Security Options dialog box closes, the Message Bar closes and the macros are enabled.* | **Click OK**

---

**RUNNING A MACRO**

**Discussion**

Macros are saved in modules, and both are saved with the workbook in which they were written. Before you can run a macro, the workbook in which it is saved must be open. When you select a macro from the Macro dialog box, Excel performs its commands in sequence.
Macros in any open workbook can be run from any other open workbook.

You can also run macros by clicking the **Macros** button in the **Code** group on the **Developer** tab.

In order for this topic to work properly, macro security must be set to the default setting. To check and, if necessary, set the correct option, select the **Office** button, the **Excel Options** button, the **Trust Center** page, and the **Trust Center Settings** button. Then, select the **Disable all macros with notification** option and **OK** twice.

### Procedures

1. Open a workbook containing macros and enable the macros.
2. Select the **View** tab in the **Ribbon**.
3. Select the top part of the **Macros** button in the **Macros** group.
4. Select the macro you want to run.
5. Select Run.

Step-by-Step

Run a macro.

If necessary, enable the macros.

The macro used in the following step-by-step opens the NEWYORK.XLSX workbook but does not contain a file path. As a result, it will look in the current folder for the file. If you opened the CITY56.XLSM file by selecting it from the recently used files list, the current folder may not be the folder that contains the NEWYORK.XLSX file. Make sure that your current folder is set for the path containing the student data files by using the Open command in the Office menu to open the CITY56.XLSM file.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the View tab in the Ribbon. The View tab is displayed.</td>
<td>Click View</td>
</tr>
<tr>
<td>2. Select the top part of the Macros button in the Macros group. The Macro dialog box opens.</td>
<td>Click</td>
</tr>
<tr>
<td>3. Select the macro you want to run. The macro is selected.</td>
<td>Click NewYork</td>
</tr>
<tr>
<td>4. Select Run. The Macro dialog box closes, and the macro runs.</td>
<td>Click Run</td>
</tr>
</tbody>
</table>

Close the NEWYORK.XLSX workbook.

Using a Shortcut Key

Discussion

Excel performs macro commands in sequence. If a macro has a shortcut key assigned to it, you can run the macro by pressing the shortcut key. A shortcut key consists of pressing the [Ctrl] key and a key assigned to the macro. If an uppercase letter has been assigned, you need to press the [Ctrl] key, the [Shift] key, and the letter assigned to the macro.
The workbook containing the macro must be open.

Macro shortcut keys are case-sensitive. If the shortcut key is the [Ctrl] key plus the uppercase letter N, you must press [Ctrl+Shift+N] or the macro will not run.

In order for this topic to work properly, macro security must be set to the default setting. To check and, if necessary, set the correct option, select the Office button, the Excel Options button, the Trust Center page, and the Trust Center Settings button. Then, select the Disable all macros with notification option and OK twice.

Procedures

1. Open a workbook containing macros and enable the macros.
2. Press the shortcut key assigned to the macro.

Step-by-Step

Use a shortcut key to run a macro.

If necessary, enable the macros.

The shortcut key for this macro is the lowercase letter n.

The macro used in the following step-by-step opens the NEWYORK.XLSX workbook but does not contain a file path. As a result, it will look in the current folder for the file. If you opened the CITY56.XLSM file by selecting it from the recently used files list, the current folder may not be the folder that contains the NEWYORK.XLSX file. Make sure that your current folder is set for the path containing the student data files by using the Open command in the Office menu to open the CITY56.XLSM file.

Steps | Practice Data
--- | ---
1. Press the shortcut key assigned to the macro.  
   The macro runs. | Press [Ctrl+n]
Close the NEWYORK.XLSX workbook.

**DISPLAYING THE DEVELOPER TAB**

**Discussion**

The Developer tab provides a variety of buttons that can be used when creating, editing, or running macros.

The Macros button opens the Macro dialog box so that you can run a macro. The Record Macro button allows you to record a macro. The Visual Basic button opens the Visual Basic Editor window, in which you can edit an existing macro or write new macros.

The Macro Security button opens the Trust Center on the Macro Settings page so that you can set the security level you want to use when opening workbooks containing macros.

Other buttons are used for automating forms and working with HTML tags and scripts for workbooks saved as web pages.

![Developer tab](image)

**Procedures**

1. Select the Office button.
2. Select Excel Options.
3. Select Show Developer tab in the Ribbon.
4. Select OK.
5. Select the Developer tab.


## Step-by-Step

Display the **Developer** tab.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>Office</strong> button.</td>
<td><img src="Click.png" alt="Click" /></td>
</tr>
<tr>
<td><em>The Office menu opens.</em></td>
<td></td>
</tr>
<tr>
<td>2. Select the <strong>Excel Options</strong> button.</td>
<td>![Click](Excel Options.png)</td>
</tr>
<tr>
<td><em>The Excel Options dialog box opens with</em></td>
<td></td>
</tr>
<tr>
<td><em>the Popular page displayed in the</em></td>
<td></td>
</tr>
<tr>
<td><em>right-hand pane.</em></td>
<td></td>
</tr>
<tr>
<td>3. Select the <strong>Show Developer tab in the</strong></td>
<td>![Click](Show Developer tab in the Ribbon.png) to select it</td>
</tr>
<tr>
<td><em>Ribbon</em> option.</td>
<td></td>
</tr>
<tr>
<td><em>The option is selected.</em></td>
<td></td>
</tr>
<tr>
<td>4. Select <strong>OK</strong>.</td>
<td><img src="OK.png" alt="Click" /></td>
</tr>
<tr>
<td><em>The Excel Options dialog box closes and</em></td>
<td></td>
</tr>
<tr>
<td><em>the Developer tab appears on the</em></td>
<td></td>
</tr>
<tr>
<td><em>Ribbon.</em></td>
<td></td>
</tr>
<tr>
<td>5. Select the <strong>Developer</strong> tab.</td>
<td><img src="Developer.png" alt="Click" /></td>
</tr>
<tr>
<td><em>The Developer tab is displayed.</em></td>
<td></td>
</tr>
</tbody>
</table>

### Opening the Visual Basic Editor Window

#### Discussion

Excel macros are written in the Visual Basic for Applications (VBA) programming language. Macros are saved in modules, which in turn are saved in the worksheet in which the macros were written. Modules cannot be viewed in the normal window view; they can only be displayed in the Visual Basic Editor window. You can create, edit, view and run macros from this window.

The Visual Basic Editor window houses several task panes and windows. The Project Explorer is displayed at the top-left of the work area; it displays the various Microsoft Excel Objects and Modules in a tree format. The tree can be expanded or collapsed in much the same way as files and folders are expanded or collapsed in Windows Explorer.

The Properties Window is displayed at the bottom-left of the work area. It shows the properties of the Microsoft Excel Object or Module that is currently selected in the Project Explorer.
Both the Project Explorer and the Properties Window are task panes and can be dragged to any position within the Visual Basic Editor window or docked against any side of the window.

One or more Code windows may be displayed to the right of the Project Explorer and Properties Window task panes. Code windows display the programming code attached to a Microsoft Excel Object or Module. For example, when a Module Code window is displayed, all the details of your macros are displayed including the macro name, description, keyboard shortcut, macro code and any additional comments. In the Code window, you can view, edit, write and run macros. Code windows can be moved, resized, minimized or maximized within the Visual Basic Editor window.

You can open the Code window for any Object or Module by double-clicking its name in the Project Explorer.

The Visual Basic Editor window

- If the Project Explorer or the Properties Window is not open in the Visual Basic Editor window, you can click the Project Explorer or Properties Window button on the Standard toolbar in the Visual Basic Editor window to open them. You can also select the View menu and choose Project Explorer or Properties Window.

- You can also open any Code window from the View menu but you must first select the name of the Object or Module in the Project Explorer.
The Visual Basic Editor window can also be opened by pressing the [Alt+F11] key combination. You can use the [Alt+F11] key combination to toggle between Excel and the Visual Basic Editor.

To close the Visual Basic Editor window, select the File menu and click Close and Return to Microsoft Excel. You can also use the Close button at the right-hand end of the Title bar in the Visual Basic Editor window.

Procedures

1. Open a workbook containing macros and enable the macros.
2. Select the Developer tab.
3. Select in the Code group.

Step-by-Step

Open the Visual Basic Editor window.

If necessary, enable the macros and show the Developer tab in the Ribbon.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Developer tab. The Developer tab is displayed.</td>
<td>Click Developer</td>
</tr>
<tr>
<td>2. Select the Visual Basic button in the Code group. The Visual Basic Editor window opens.</td>
<td>Click Visual Basic</td>
</tr>
</tbody>
</table>

Practice the Concept: Maximize the Module 1 Code window. Press [Alt+F11] to switch to the Excel window. Then, display the Visual Basic Editor window again.
USING THE VISUAL BASIC EDITOR WINDOW

Discussion

The Visual Basic Editor window displays the Project Explorer window, the Properties window, and the Code window. Each of these windows can be opened and closed independently. When a closed window is reopened, it opens at its previous size and location.

You can also press the [Ctrl+R] key combination to open the Project Explorer window, the [F4] key to open the Properties window, and the [F7] key to open the Code window.

Procedures

1. Open a workbook containing macros and enable the macros.
2. To close the Project Explorer task pane, select the Close button on its Title bar.
3. To open the Project Explorer task pane, select the Project Explorer button on the Standard toolbar.
4. To close the Properties Window task pane, select the Close button on its Title bar.
5. To open the Properties Window task pane, select the Properties Window button on the Standard toolbar.
6. To close a maximized Code window, select its Close button at the right-hand end of the Menu bar.
7. To expand an item in the Project Explorer task pane, select the plus sign to the left of the item.
8. To open the Code window for a Module, double-click the module name.

Step-by-Step

Use the Visual Basic Editor window.
If necessary, enable the macros and open the Visual Basic Editor window.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the <strong>Close</strong> button on the Title bar of the Project Explorer task pane. <em>The Project Explorer task pane closes.</em></td>
<td>Click ✗ on the Project Explorer task pane</td>
</tr>
<tr>
<td>2. Select the <strong>Project Explorer</strong> button on the <strong>Standard</strong> toolbar. <em>The Project Explorer task pane opens.</em></td>
<td>Click <img src="image" alt="Project Explorer" /></td>
</tr>
<tr>
<td>3. Select the <strong>Close</strong> button on the Title Bar of the Properties Window task pane. <em>The Properties Window task pane closes.</em></td>
<td>Click ✗ on the Properties Window task pane</td>
</tr>
<tr>
<td>4. Select the <strong>Properties Window</strong> button on the <strong>Standard</strong> toolbar. <em>The Properties Window task pane opens.</em></td>
<td>Click <img src="image" alt="Properties Window" /></td>
</tr>
<tr>
<td>5. Select the <strong>Code window</strong> <strong>Close</strong> button. <em>The Code window closes.</em></td>
<td>Click ✗ at the right-hand end of the Menu bar</td>
</tr>
<tr>
<td>6. Select the plus sign to the left of the <strong>Modules</strong> folder in the Project Explorer window. <em>The folder expands to display its contents.</em></td>
<td>Click <img src="image" alt="Modules" /> to the left of Modules</td>
</tr>
<tr>
<td>7. Double-click the module you want to display. <em>The Code window opens with the contents of the module displayed.</em></td>
<td>Double-click <strong>Module 1</strong></td>
</tr>
</tbody>
</table>

Notice that the module contains two macros, **NewYork** and **Atlanta**, which open **NEWYORK.XLSX** and **ATLANTA.XLSX** respectively.

Close the Visual Basic Editor window. Close **CITY56.XLSM**.

Villanova UNIT Training©
EXERCISE

USING MACROS

Task

Use macros in a workbook.

1. Open EXER81.XLSM and enable the macros.

2. Use the Macros button in the Macros group on the View tab to run the Sort.RegionSales macro. Notice that the macro sorts the data, first by region in ascending order, and then by annual sales in descending order.

3. Use the Macros button in the Code group on the Developer tab to run the Sort.LastName macro. Notice that the macro sorts the data by last name in ascending order.

4. Use the [Ctrl+r] shortcut key to run the Subtotals.Region macro. Notice that the macro adds an Outline and Subtotals to the sorted data.

5. Open the Visual Basic Editor window.

6. Display the Module1 module in the Code window, if necessary. Scroll as necessary to view the Subtotals.Remove macro.

7. Close the Visual Basic Editor window. Then, run the Subtotals.Remove macro.

8. Hide the Developer tab.

9. Close the workbook without saving it.
LESSON 11 -
RECORDING MACROS

In this lesson, you will learn how to:

- Record a macro
- Assign a shortcut key
- Use relative references
- Delete a macro
RECORDING A MACRO

Discussion

A macro executes a series of previously recorded and saved actions. You can create macros for repetitious tasks you perform frequently or to automate tasks that involve a lengthy number of steps. When using macros, you can be assured that the task is performed in exactly the same way each time; therefore, macros can ensure consistency and accuracy, as well as save you time.

Before you create a macro, it is a good idea to make sure that you know all the steps needed to perform the desired procedure so that you can record the macro properly. The steps you carry out are not recorded as simple keystrokes; they are translated into the Visual Basic for Application programming language, which translates your steps into macro statements. When you record a macro, Excel creates a module containing the Visual Basic language statements used to define and create the macro. A module can contain more than one macro.

You use the macro recorder to record a macro which captures each keystroke and menu selection, with some exceptions. For example, navigation on the Ribbon is not included in the recorded steps. If you make a typing error and press the [Backspace] key to correct it, neither the error nor the [Backspace] command is recorded. Likewise, if you select the Cancel button to close a dialog box, neither the command that opened the dialog box nor the Cancel command is recorded.

While the macro recorder is running, the Record Macro button in the Code group on the Developer tab changes into the Stop Recording button. A Stop Recording button also appears on the left-hand side of the status bar.

During the macro recording process, you name the macro and enter descriptive information into the Record Macro dialog box. This information appears as a comment in the module.

You should assign a name to the macro that identifies what the macro will accomplish. Although a macro name must begin with a letter, it can contain both letters and numbers. A macro name cannot contain any spaces or symbols, but you can use an underscore (_) character to separate words. If you use a macro name that is also a cell reference, an error message may appear to inform you that the macro name is invalid.

In addition to naming the macro, you can select where you want to store it. Macros can be stored in the current workbook, a new workbook or the Personal Macro Workbook. Macros stored in a current or new workbook are only available when that workbook is open. If you want a macro to be available whenever you use Excel, you can store it in the Personal Macro Workbook. This is saved in the XLStart folder and will open automatically whenever Excel starts.
If you record a macro to the **Personal Macro Workbook**, you are prompted to save the changes when you exit Excel.

To use the **Developer** tab, you should ensure that the **Show Developer in the Ribbon** option is selected. This option is located in the **Popular** page of the Excel Options dialog box.

You can also open the Record Macro dialog box by selecting the bottom part of the **Macros** button on the **View** tab and then clicking **Record Macro**.

Some words are reserved and cannot be used in a macro name. Invalid words are usually macro commands, such as **Date** or **If**.

### Procedures

1. Select the **Developer** tab on the **Ribbon**.
2. Select ![Record Macro](Record Macro.png) in the **Code** group.
3. Type the desired macro name.
4. Select the Shortcut key box, if desired.
5. Type the desired shortcut key character.
6. Select the Store macro in list.
7. Select the desired location in which to store the macro.
8. Select the Description box.
9. Type the desired macro description.
10. Select OK.
11. Perform the steps in the procedure you want to automate.
12. When you have finished recording macro steps, click Stop Recording in the Code group.

Step-by-Step

From the Student Data directory, open CITY57.XLSM. Record a macro.

If necessary, enable the macros.

Make sure that your current folder is set for the path containing the student data files.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Developer tab on the Ribbon. &lt;br&gt;The Developer tab is displayed.</td>
<td>Click Developer</td>
</tr>
<tr>
<td>2. Select the Record Macro button in the Code group. &lt;br&gt;The Record Macro dialog box opens and the suggested text in the Macro name box is highlighted.</td>
<td>Click Record Macro</td>
</tr>
<tr>
<td>3. Type the desired macro name. &lt;br&gt;The text appears in the Macro name box.</td>
<td>Type Chicago</td>
</tr>
<tr>
<td>4. Select the Shortcut key box, if desired. &lt;br&gt;The insertion point is placed in the Shortcut key box.</td>
<td>Press [Tab]</td>
</tr>
</tbody>
</table>
### Steps

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 5. Type the desired shortcut key character.  
*The character appears in the **Shortcut key** box.* | Type *e* |
| 6. Select the **Store macro in** list.  
*A list of available options is displayed.* | Click **Store macro in** |
| 7. Select the desired location in which to store the macro.  
*The location appears in the **Store macro in** box.* | Click **This Workbook** |
| 8. Select the **Description** box.  
*The insertion point is placed in the **Description** box.* | Press **Tab** |
| 9. Type the desired macro description. *The text appears in the **Description** box.* | Type **Opens the Chicago workbook** |
| 10. Select **OK**.  
*The Record Macro dialog box closes. The **Record Macro** button in the **Code** group changes to the **Stop Recording** button and a **Stop Recording** button is also displayed on the status bar.* | Click **OK** |
| 11. Perform the steps in the procedure that you want to automate.  
*The procedure is carried out and recorded.* | Follow the instructions shown below the table before continuing on to the next step |
| 12. When you have finished recording macro steps, click the **Stop Recording** button in the **Code** group.  
*The macro recorder stops. The **Record Macro** button displays again and the macro recording is complete.* | Click **Stop Recording** |

Perform the following steps to create the macro:

Click the **Office** button and then select **Open** from the **Office** menu. Double-click **CHICAGO.XLSX** in the student data folder.

*Return to the table and continue on to the next step (step 12).*

Close the **CHICAGO.XLSX** workbook. Click the **Visual Basic** button in the **Code** group to open the **Visual Basic** editor. Double-click **Module2** in the **Project - VBAProject** task pane. View the macro in the **CITY57.XLSM - Module2 (Code)**
pane. Notice that the macro has recorded the filename path to the CHICAGO.XLSX workbook. (You may need to resize the pane to see the entire filename path). Select File on the menu bar and then click Close and Return to Microsoft Excel.

Use the [Ctrl+e] shortcut key to run the Chicago macro. Then, close the CHICAGO.XLSX workbook.

ASSIGNING A SHORTCUT KEY

Discussion

You can assign a shortcut key to a macro, either when you create the macro or after the macro has been created. After a shortcut key has been assigned to a macro, the macro runs whenever the shortcut key is pressed. All macro shortcut keys must begin with the [Ctrl] key followed by any character.

If you assign a shortcut key while the macro is being created, it appears as a comment in the module containing the Visual Basic language statements used to define and create the macro. When a shortcut key is assigned to an existing macro, however, this comment is not automatically created. You can edit the macro text, however, to manually add the comment.

Assigning a shortcut key to a macro

Shortcut keys are assigned as you are recording a macro by entering the desired shortcut key in the Record Macro dialog box.
To use the Developer tab, you should ensure that the Show Developer in the Ribbon option is selected. This option is located in the Popular page of the Excel Options dialog box.

You should avoid assigning macro shortcut keys that are the same as the standard Excel shortcuts such as [Ctrl+O] and [Ctrl+S]. The macro shortcut key will override any equivalent Excel shortcut key while the workbook containing the macro is open.

Procedures

1. Select Macros in the Code group on the Developer tab.
2. Select the macro to which you want to assign a shortcut key from the Macro list.
3. Select Options…
4. Type the desired shortcut key character.
5. Select OK
6. Select Cancel

Step-by-Step

Assign a shortcut key to a macro.

If necessary, enable the macros and display the Developer tab.

The macro used in the following step-by-step does not contain a file path. As a result, it will look in the folder that contains the CITY57.XLSM data file. Make sure that your current folder is set for the path containing the student data files.
Lesson 11 - Recording Macros

Steps | Practice Data
--- | ---
1. Select the **Macros** button in the **Code** group on the **Developer** tab.  
   *The Macro dialog box opens with the first macro name highlighted in the Macro name box.* | Click [Macros]

2. Select the macro to which you want to assign a shortcut key from the **Macro** list.  
   *The macro is selected.* | Click **Atlanta**, if necessary

3. Select the **Options** button.  
   *The Macro Options dialog box opens with the insertion point in the Shortcut key box.* | Click [Options…]

4. Type the desired shortcut key character.  
   *The character appears in the Shortcut key box.* | Type **a**

5. Select **OK**.  
   *The Macro Options dialog box closes and the shortcut key is assigned to the macro.* | Click [OK]

6. Select **Cancel**.  
   *The Macro dialog box closes.* | Click [Cancel]

Use the [Ctrl+a] shortcut key to run the **Atlanta** macro. Then, close the **ATLANTA.XLSX** workbook.

**USING RELATIVE REFERENCES**

**Discussion**

Absolute and relative references refer to how a macro records cell locations. When you use absolute references while recording a macro, Excel records absolute, or exact, cell references. For example, if you enter the label **Annual Sales Report** in cell A4 while recording a macro with absolute references, that label will always appear in cell A4 whenever you run the macro, no matter which cell is selected on the worksheet.

When you use relative references while recording a macro, Excel records cell references relative to the active cell. For example, if you type the label **Annual Sales Report** in the cell to the left of the active cell while recording a macro with relative references, that label will always be entered into the cell to the left of whichever cell is active when you run the macro. Relative references give a macro more flexibility.
Procedures

1. Select the desired cell in the worksheet.
2. Click **Record Macro** in the **Code** group on the **Developer** tab.
3. Type the desired macro name.
4. Select the **Description** box.
5. Type the desired macro description.
6. Select **OK**.
7. Click **Use Relative References** in the **Code** group on the **Developer** tab.
8. Perform the steps in the procedure that you want to automate.
9. When you have finished recording macro steps, click **Stop Recording** in the **Code** group.

Step-by-Step

Use relative references to record a macro.
If necessary, enable the macros and display the **Developer** tab.

<table>
<thead>
<tr>
<th><strong>Steps</strong></th>
<th><strong>Practice Data</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the desired cell in the worksheet.</td>
<td>Click A14</td>
</tr>
<tr>
<td><em>The cell is selected.</em></td>
<td></td>
</tr>
<tr>
<td>2. Click the <strong>Record Macro</strong> button in the <strong>Code</strong> group on the <strong>Developer</strong> tab.</td>
<td>Click <img src="image" alt="Record Macro" /></td>
</tr>
<tr>
<td><em>The Record Macro dialog box opens with the suggested name highlighted in the Macro name box.</em></td>
<td></td>
</tr>
<tr>
<td>3. Type the desired macro name.</td>
<td>Type <strong>Company</strong></td>
</tr>
<tr>
<td><em>The new name appears in the Macro name box.</em></td>
<td></td>
</tr>
<tr>
<td>4. Select the <strong>Description</strong> box.</td>
<td>Click in the <strong>Description</strong> box</td>
</tr>
<tr>
<td><em>The insertion point is placed in the Description box.</em></td>
<td></td>
</tr>
<tr>
<td>5. Type the desired macro description.</td>
<td>Type <strong>Runs the Company label</strong></td>
</tr>
<tr>
<td><em>The text appears in the Description box.</em></td>
<td></td>
</tr>
<tr>
<td>6. Select <strong>OK</strong>.</td>
<td>Click <img src="image" alt="OK" /></td>
</tr>
<tr>
<td><em>The Record Macro dialog box closes. The Record Macro button in the Code group changes to the Stop Recording button and a Stop Recording button is also displayed on the status bar.</em></td>
<td></td>
</tr>
<tr>
<td>7. Click the <strong>Use Relative References</strong> button in the <strong>Code</strong> group on the <strong>Developer</strong> tab.</td>
<td>Click <img src="image" alt="Use Relative References" /></td>
</tr>
<tr>
<td><em>The Use Relative References button is selected and highlighted.</em></td>
<td></td>
</tr>
<tr>
<td>8. Perform the steps in the procedure that you want to automate.</td>
<td>Follow the instructions shown below the table before continuing on to the next step</td>
</tr>
<tr>
<td><em>The procedure is carried out and recorded.</em></td>
<td></td>
</tr>
<tr>
<td>9. When you have finished recording macro steps, click the <strong>Stop Recording</strong> button in the <strong>Code</strong> group.</td>
<td>Click <img src="image" alt="Stop Recording" /></td>
</tr>
<tr>
<td><em>The macro recorder stops. The Record Macro button displays again and the macro recording is complete.</em></td>
<td></td>
</tr>
</tbody>
</table>
Display the **Home** tab and format the current cell with bold and italics using the appropriate buttons in the **Font** group. Also, change the font size to **14**. Then, type **Worldwide Sporting Goods** into the current cell and press **[Enter]**. Redisplay the **Developer** tab.

*Return to the table and continue on with the next step (step 9).*

Select cell A14 and display the **Home** tab. Click the **Clear** button in the **Editing** group and then select **Clear All** to remove the format and contents of cell A14.

Select cell A1. Redisplay the **Developer** tab and click the **Macros** button in the **Code** group. Select the **Company** macro in the Macro dialog box and click **Run**. Notice that the label is entered in cell A1 because the macro was recorded to run with reference to the currently selected cell in the worksheet.

Click the **Visual Basic** button in the **Code** group to open the **Visual Basic** editor. Double-click the last recorded module in the **Project - VBAProject** task pane to view the **Company** macro. Select **File** on the menu bar and then click **Close and Return to Microsoft Excel**. Click the **Use Relative References** button in the **Code** group on the **Developer** tab to deselect it.

### DELETING A MACRO

#### Discussion

If you no longer use a macro, you can delete it. Since macros can invoke other macros, you should make sure that the macro you are deleting is not needed by any other macro.

If a macro is stored in a workbook, the workbook must be open in order to delete the macro. If the macro is stored in the **PersonalMacro Workbook**, which automatically loads as a hidden window every time you open Excel, the **Personal Macro Workbook** window must be unhidden before the macro can be deleted.

---

*You can unhide hidden workbooks, such as the **Personal Macro Workbook**, by selecting the **Unhide** button in the **Window** group on the **View** tab. If **Unhide** is unavailable, then the workbook does not contain hidden workbook windows.*

*If you delete a macro from the **Personal Macro Workbook**, Excel prompts you to save the changes to the **Personal Macro Workbook** when you exit Excel.*
You can also open the Macro dialog box by selecting the bottom part of the Macros button on the View tab and then clicking View Macros.

Procedures

1. Open the workbook containing the macro you want to delete and enable the macros.
2. Select the Developer tab on the Ribbon.
3. Click Macros in the Code group.
4. Select the macro you want to delete from the Macro list.
5. Select Delete.
6. Select Yes.

Step-by-Step

Delete a macro.

If necessary, enable the macros.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Developer tab on the Ribbon.</td>
<td>Click Developer, if necessary</td>
</tr>
<tr>
<td>The Developer tab is displayed.</td>
<td></td>
</tr>
<tr>
<td>2. Click the Macros button in the Code</td>
<td>Click Macros</td>
</tr>
<tr>
<td>group.</td>
<td></td>
</tr>
<tr>
<td>The Macro dialog box opens.</td>
<td></td>
</tr>
<tr>
<td>3. Select the macro you want to delete</td>
<td>Click Company</td>
</tr>
<tr>
<td>from the Macro list.</td>
<td></td>
</tr>
<tr>
<td>The macro is selected and the name is</td>
<td></td>
</tr>
<tr>
<td>displayed in the Macro name box.</td>
<td></td>
</tr>
</tbody>
</table>
### Steps

<table>
<thead>
<tr>
<th>Steps</th>
</tr>
</thead>
</table>
| 4. Select **Delete**.  
*The Macro dialog box closes and a Microsoft Office Excel message box opens asking you to confirm the deletion.* |

<table>
<thead>
<tr>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click <a href="#">Delete</a></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Steps</th>
</tr>
</thead>
</table>
| 5. Select **Yes**.  
*The Microsoft Office Excel message box closes and the macro is deleted.* |

<table>
<thead>
<tr>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click <a href="#">Yes</a></td>
</tr>
</tbody>
</table>

Close **CITY57.XLSM**.
EXERCISE

RECORDING MACROS

Task

Record Macros.

1. Open EXER82.XLSM and enable the macros.
2. Display the Last Year worksheet, if necessary.
3. In cell A1, begin recording a macro named Today. Assign [Ctrl+m] as a shortcut key. Store the macro in the current workbook.
4. Using relative references, perform the following steps to record the Today macro:
   - Add bold and a red font color to the active cell.
   - Change the font size to 12.
   - Type =today().
   - Press [Enter].
5. Stop recording the macro. Clear all contents and formatting from cell A1. (Hint: Use the Clear button in the Editing group on the Home tab.)
6. Select cell A4 and run the macro.
7. View the macro in the Visual Basic editor; display Module3, if necessary. Then, close and return to Excel.
8. Display the Projections worksheet. Use the shortcut key to run the macro in cell A1.
9. Delete the Today macro.
10. Deselect the Use Relative References button and close the workbook without saving it.
LESSON 12 - EDITING MACROS

In this lesson, you will learn how to:

- Write a new macro
- Enter macro comments
- Copy macro commands
- Edit macro commands
- Type macro commands
- Run a macro from the Code window
**Writing a New Macro**

**Discussion**

New macros can be created by recording the steps or typing the commands directly into the Code windows in the Visual Basic Editor. To write a new macro, you must begin by typing the word `Sub`, followed by the macro name and a set of parentheses `()`. Excel automatically supplies the `End Sub` command. All steps in the procedure must be entered between the `Sub` and `End Sub` commands.

New macros can be inserted in an existing module or added to a new module. You can insert a macro above or below any existing macro; the position of the macro in the module does not affect its performance.

- You can create a new module in a Visual Basic project by selecting the **Insert** menu and then clicking **Module**. A new module with an incremental number, such as **Module3**, will then be inserted into the current project.

- When you insert a macro above or below an existing macro in a module, the Visual Basic Editor automatically inserts horizontal lines between macros to separate them.
To use the Developer tab, you should ensure that the Show Developer in the Ribbon option is selected. This option is located in the Popular page of the Excel Options dialog box.

Procedures

1. Display the Developer tab on the Ribbon.

2. Select Visual Basic in the Code group.

3. Display or insert the module in which you want to write the new macro.

4. Resize or maximize the Code window, as desired.

5. Position the insertion point above or below any existing macros in the Code window.

6. Type Sub.

7. Press [Spacebar].

8. Type the macro name, followed by a set of parentheses ( ( ) ).

9. Press [Enter].

Step-by-Step

From the Student Data directory, open CITY58.XLSM. Write a new macro.

If necessary, enable the macros.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Display the Developer tab on the Ribbon. The Developer tab is displayed.</td>
<td>Click Developer</td>
</tr>
<tr>
<td>2. Select the Visual Basic button in the Code group. The Visual Basic Editor opens.</td>
<td>Click Visual Basic</td>
</tr>
</tbody>
</table>
Steps | Practice Data
---|---
3. Display or insert the module in which you want to write the new macro. *The desired module Code window is displayed.* | Double-click **Module1** in the **Project Explorer** task pane, if necessary.

4. Resize or maximize the Code window, as desired. *The Code window is maximized.* | Click ![Maximize](maximize.png)

5. Position the insertion point above or below any existing macros in the Code window. *The insertion point is placed below the existing macros in the Code window.* | Press **[Ctrl+End]**

6. Type **Sub**. *The Sub command appears at the insertion point.* | Type **Sub**

7. Press **[Spacebar]**. *A space appears after the Sub command.* | Press **[Spacebar]**

8. Type the macro name, followed by a set of parentheses ( () ). *The text and parentheses appear at the insertion point.* | Type **Open_Cities()**

9. Press **[Enter]**. *The insertion point moves to the next line, the End Sub command appears on the line below the insertion point and a horizontal line appears above the Sub command. The macro name is displayed in the macro name list at the top of the Code window.* | Press **[Enter]**

---

**ENTERING MACRO COMMENTS**

**Discussion**

When macros are first recorded, the recording process documents the name, date, and other macro descriptions in the Code window. This information appears before the actual macro statements. If you make a change to a macro, you should go into the documentation area of the macro and edit this information as necessary. For example, if you add a shortcut key to a macro after the macro has been recorded, this
information is not automatically added to the macro documentation. You must manually add a line to document the change.

In longer and more complex macros, you can add comments to the various macro steps. Comments can explain what a step or series of steps should accomplish. This information can be helpful if the macro must be modified or debugged at a later date.

Comments must begin with a single apostrophe (‘). Excel treats any line that begins with a single apostrophe as a comment. Comments can appear either on a separate line or adjacent to a macro statement. Comments appear in black text when they are entered and change to green text as soon as you move to another line.

![Entering macro commands in the Code window](image)

**Procedures**

1. Open the workbook containing a macro to which you want to add a comment and enable the macros.

2. Display the **Developer** tab on the **Ribbon**.

3. Select **Visual Basic** in the **Code** group to open the Visual Basic Editor.

4. Display the module containing the macro in which you want to add the comment.
5. Resize or maximize the Code window, as desired.
6. Position the insertion point where you want to add the comment.
7. Type an apostrophe (').
8. Press [Spacebar].
9. Type the desired comment text.
10. Press [Enter].

**Step-by-Step**

Enter macro comments.

If necessary, enable the macros. Open the Visual Basic Editor using the Visual Basic button in the Code group on the Developer tab and then display the CITY58.XLSM - Module1 (Code) window using the Project Explorer task pane.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Position the insertion point where you want to add the comment.</td>
<td>Click in the blank line below Sub Open_Cities(), if necessary</td>
</tr>
<tr>
<td><em>The insertion point is placed in the desired location in the Code window.</em></td>
<td></td>
</tr>
<tr>
<td>2. Type an apostrophe (').</td>
<td>Type '</td>
</tr>
<tr>
<td><em>The apostrophe ('') appears in the line.</em></td>
<td></td>
</tr>
<tr>
<td>3. Press [Spacebar].</td>
<td>Press [Spacebar]</td>
</tr>
<tr>
<td><em>A space appears after the apostrophe ('').</em></td>
<td></td>
</tr>
<tr>
<td>4. Type the desired comment text.</td>
<td>Type This macro opens three city files</td>
</tr>
<tr>
<td><em>The comment text appears in the Code window.</em></td>
<td></td>
</tr>
<tr>
<td>5. Press [Enter].</td>
<td>Press [Enter]</td>
</tr>
<tr>
<td><em>The insertion point moves to the next line and the comment text turns green.</em></td>
<td></td>
</tr>
</tbody>
</table>

**Practice the Concept:** Double-click Module2 in the Project Explorer task pane to display the CITY58.XLSM - Module2 (Code) window. In the Company macro, click in the blank line above the Selection.Font.Bold = True command. Type an apostrophe ('), then press [Spacebar]. Type the following comment: Formats the current cell. Press [Enter].

Scroll down, if necessary and click at the end of the End With command in the Company macro; press [Enter] to insert a new line. Then, type the following
comment: 'Inserts the company name. Click in the next line to see the comment color change to green.

COPYING MACRO COMMANDS

Discussion

There may be times when you want to combine macros to operate them more efficiently. For example, if three different macros open three separate workbooks, you can combine the three macros to open all three workbooks in one step. You can combine macros by copying and pasting the command statements from one macro to another. Macro commands can be copied from other macros, other modules, or even other workbooks.

Copying macro commands saves the time it takes to write or re-record a macro, or insert new macro steps.

If an error occurs while running a macro, the Microsoft Visual Basic dialog box opens. You can select the End button to stop running the macro. If you select the Debug button, the Visual Basic Editor opens with the command that created the error highlighted. You can fix the error and click the Continue button on the Standard toolbar to continue the macro, or you can click the Reset button on the Standard toolbar to stop debugging the macro.

Procedures

1. Open a workbook containing macros and enable the macros.
2. Display the Developer tab on the Ribbon.
3. Select in the Code group to open the Visual Basic Editor.
4. Display the module containing the macro text you want to copy.
5. Select the macro text you want to copy.
6. Click the Copy button on the Visual Basic Editor Standard toolbar.
7. Position the insertion point where you want to paste the macro text.
8. Click the **Paste** button on the Visual Basic Editor **Standard** toolbar.

---

**Step-by-Step**

Copy macro commands between modules.

If necessary, enable the macros. Open the Visual Basic Editor using the **Visual Basic** button in the **Code** group on the **Developer** tab. Then, open the CITY58.XLSM - Module1 Code window using the **Project Explorer** task pane and display the **NewYork** macro.

Make sure that your current folder is set for the path containing the student data files.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Select the macro text you want to copy.  
*The macro text is highlighted as you drag.* | Drag to select the **Workbooks.Open**  
Filename:= "NEWYORK.XLSX" text |
| 2. Release the mouse button.  
*The desired macro text is selected.* | Release the mouse button |
| 3. Select the **Copy** button on the Visual Basic Editor **Standard** toolbar.  
*The text is copied to the Clipboard.* | Click |
| 4. Position the insertion point where you want to paste the macro text.  
*The insertion point is placed in the desired location.* | Scroll as necessary and click in the **Open_Cities** macro in the line above the **End Sub** command |
| 5. Select the **Paste** button on the Visual Basic Editor **Standard** toolbar.  
*The copied macro text is pasted at the insertion point.* | Click |

Press [**Home**] to place the insertion point to the left of the inserted macro text  
**Workbooks.Open Filename:= "NEWYORK.XLSX"** and press [**Tab**] to indent it.  
Press [**End**] and then press [**Enter**].

**Practice the Concept:** Select the text in the **Atlanta** macro that opens the **ATLANTA.XLSX** workbook, copy and paste it into the **Open_Cities** macro, on the line below the step that opens the **NEWYORK.XLSX** workbook.
Then, double-click Module2 in the Project Explorer task pane to display the CITY58.XLSM - Module2 Code window. Select and copy the step in the Chicago macro that opens the CHICAGO.XLSX workbook. Paste the macro text into the Open_Cities macro in the Module1 Code window, on a line below the step that opens the ATLANTA.XLSX workbook. All three macro commands should appear above the End Sub command.

Select the File menu and then Close and Return to Microsoft Excel. Click the Macros button in the Code group on the Developer tab and run the Open_Cities macro in the CITY58.XLSM workbook. Then, run the Close_Cities macro to close the NEWYORK.XLSX, ATLANTA.XLSX, and CHICAGO.XLSX workbooks.

EDITING MACRO COMMANDS

Discussion

Depending on the changes you need to make, it may be easier to edit the macro text rather than to delete the entire macro and record it again. You can use standard editing techniques to edit macro text.

You can rename a macro by changing the macro name in the Sub command. You may want to change any associated comments, descriptions, or shortcut keys. Changing comment text in the Code window does not change the text or shortcut key in the macro itself. To carry out these changes, you must use the Options button in the Macro dialog box to open the Macro Options dialog box.

You can rename a module in the Visual Basic Editor by selecting the module in the Project Explorer task pane and typing the desired name into the (Name) box in the Properties window.

Be careful when editing command statements; if you do not use the correct syntax and punctuation, the macro may not work correctly or may not run at all.
Procedures

1. Open a workbook containing macros you want to edit and enable the macros.
2. Select the Developer tab on the Ribbon.
3. Select Macros in the Code group.
4. Select the macro you want to edit from the Macro list.
5. Select Edit.
6. Select the text you want to edit.
7. Edit the selected text.

Step-by-Step

Edit macro commands manually.

If necessary, enable the macros. Make sure that your current folder is set for the path containing the student data files.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Developer tab on the Ribbon. The Developer tab is displayed.</td>
<td>Click Developer, if necessary</td>
</tr>
<tr>
<td>2. Select the Macros button in the Code group. The Macro dialog box opens.</td>
<td>Click Macros</td>
</tr>
<tr>
<td>3. Select the macro you want to edit from the Macro list. The desired macro is selected and is displayed in the macro name box.</td>
<td>Click Open_Cities</td>
</tr>
<tr>
<td>4. Select the Edit button. The Macro dialog box closes and the Visual Basic Editor opens with the insertion point placed in the selected macro.</td>
<td>Click Edit</td>
</tr>
</tbody>
</table>
### Typing Macro Commands

#### Discussion

In addition to recording macro commands, you can also type them directly into a macro procedure. Macro procedures work with statements and objects. Some macro steps, such as macro statements, cannot be recorded. For example, you may want to create a condition that executes specific macro commands if the condition is true and other commands if the condition is false.

Objects are the elements upon which the macro acts, such as a workbook, worksheet, range, cell, or chart. A collection is a set of multiple objects. The worksheets in a workbook are part of a collection. You use VBA (Visual Basic for Applications) commands to change the properties of an object or collection or to apply a method (action) to an object. The properties and methods differ depending upon the object.

When entering a command, you must use the proper syntax. The syntax of a command is the structure of its elements. As you type commands in the Visual Basic Editor, lists with available properties or methods may appear.

There is often more than one way to perform the same action. For example, you can record a macro that deletes the contents in the range A6:F12 in the Sales worksheet. The steps could be recorded using the following selection methods since you would be manually selecting the worksheet and range.
Sheets("SALES").Select
Range("A6:F12").Select
Selection.ClearContents

You can accomplish the same result by writing a command, such as the one shown below, that applies a method to the specified object. The object is the range A6:F12 in the Sales worksheet. The command assumes that the correct workbook is open and can be written without the Worksheets property if the command to activate the worksheet was issued in a previous step.

Worksheets("SALES").Range("A6:F12").ClearContents

If a group of commands can be recorded, it might be easier to record the macro steps in a separate macro and then copy and paste them into the larger macro, than it would be to write them directly into the Code window.

The Visual Basic Editor provides a variety of help features. As you type a command, pop-up help may appear. The Auto Quick Info box and the Auto List Member list both provide help with either parameters or available methods and properties. The syntax of your command is automatically verified when you move to another line.

Procedures

1. Open a workbook containing macros and enable the macros.
2. Display the Developer tab on the Ribbon.
3. Select in the Code group to open the Visual Basic Editor.
4. Open the module containing the macro to which you want to add a command.
5. Position the insertion point where you want to add the command.
6. Press [Enter].
7. Type the desired command.
8. To enter a property or method, type a period (.)
9. Type the desired property or method, or select it from the Auto List Member list.
10. Press [Enter].
Step-by-Step

From the Student Data directory, open CITY59.XLSM.
Type a macro command.

If necessary, enable the macros. Open the Visual Basic Editor using the Visual Basic button in the Code group on the Developer tab. Then, open the CITY59.XLSM - Module1 Code window using the Project Explorer task pane to display the Open_Cities macro.

Make sure that your current folder is set for the path containing the student data files.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Position the insertion point where you want to insert the command. &lt;br&gt;The insertion point is placed in the desired location.</td>
<td>Click at the end of the Workbooks.Open Filename:= &quot;CHICAGO.XLSX&quot; text</td>
</tr>
<tr>
<td>2. Press [Enter]. &lt;br&gt;A new line is inserted.</td>
<td>Press [Enter]</td>
</tr>
<tr>
<td>3. Type the desired command. &lt;br&gt;As you enter the text, an Auto Quick Info box appears until you complete typing the command.</td>
<td>Type Workbooks(&quot;CITY59.XLSM&quot;)</td>
</tr>
<tr>
<td>4. To enter a property or method, type a period (.). &lt;br&gt;An Auto List Member list displays available properties or methods.</td>
<td>Type a period (.)</td>
</tr>
<tr>
<td>5. Type the desired property or method, or select it from the Auto List Member list. &lt;br&gt;The property or method appears in the command.</td>
<td>Double-click Activate</td>
</tr>
<tr>
<td>6. Press [Enter]. &lt;br&gt;The command syntax is verified.</td>
<td>Press [Enter]</td>
</tr>
</tbody>
</table>

Press [Alt+F11] to switch to the worksheet window. Select the Macros button in the Code group and run the Open_Cities macro. When the macro finishes running, the CITY59.XLSM workbook should be active and the TAMPA.XLSX, ATLANTA.XLSX, and CHICAGO.XLSX workbooks have opened. Then, run the Close_Cities macro. All workbooks should close, except the CITY59.XLSM workbook.
RUNNING A MACRO FROM THE CODE WINDOW

Discussion

During the editing process, a macro can be run directly from the Visual Basic Editor using the Run Sub/UserForm button on the Standard toolbar or the corresponding command located in the Run menu.

You must position the insertion point in the Code window, within the text of the macro you want to run. Otherwise, the Macros dialog box will open from which you must select and run the desired macro.

You can also run a macro directly from the Visual Basic Editor by pressing the [F5] key.

Procedures

1. Open a workbook containing macros and enable the macros.
2. Display the Developer tab on the Ribbon.
3. Select Visual Basic in the Code group to open the Visual Basic Editor.
4. Open the module containing the macro you want to run.
5. Click in the macro text you want to run.
6. Click the Run Sub/UserForm button on the Visual Basic Editor Standard toolbar.

Step-by-Step

Run a macro from the Code window.

Press [Alt+F11] to toggle to the Visual Basic Editor window and double-click Module3 in the Project Explorer task pane.

If necessary, scroll to view the Urban_Update macro in Module3. Notice that it performs the following actions: opens the TAMPA.XLSX, ATLANTA.XLSX, and
**CHICAGO.XLSX** workbooks; runs the **CleanUp** macro; consolidates the data from the three workbooks into the **CITY59.XLSM** workbook; formats the consolidated data range; runs the **Company** macro; and then closes the **TAMPA.XLSX**, **ATLANTA.XLSX**, and **CHICAGO.XLSX** workbooks.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Click in the macro text you want to run.  
_The insertion point is placed in the desired macro text._ | Click after the text **Range(“C6:F9”).Select** in the **Urban_Update** macro |
| 2. Click the **Run Sub/UserForm** button on the Visual Basic Editor **Standard** toolbar.  
_The macro commands are executed._ | Click ![button](button.png) |

Close the Visual Basic Editor. View the consolidated data in the **CITY59.XLSM** workbook.

Notice the consolidated value in cell C6 of **10,703**. Open the **ATLANTA.XLSX** workbook and change cell C7 from **3250** to **9250**. Then, close and save the changes you made to the **ATLANTA.XLSX** workbook.

Click the **Macros** button in the **Code** group on the **Developer** tab. Run the **Urban_Update** macro and notice that the consolidated value in cell C6 changes to **16,703**.  
Close **CITY59.XLSM**.
**EXERCISE**

**EDITING MACROS**

**Task**

Create and edit macros using the Visual Basic Editor.

1. Open EXER83.XLSM and enable the macros.

2. View the workbook in **Print Preview** layout. Notice the portrait page orientation and then close **Print Preview**.

3. Open the **Visual Basic Editor** and open the **Module1** Code window, if necessary. Maximize the Code window.

4. Press [Enter] twice to move the **Sub Sort.RegionSales()** down two lines.

5. Press [Ctrl+Home] to navigate to the top of the Code window. Then type **Sub Sort.RegionLastN()**, and press [Enter].

6. Enter the following comment: **Sorts by region and then by last name.** Then, press [Enter]. *(Hint: Remember to begin the comment with an apostrophe.)*

7. Copy all the macro steps from the **Sort.RegionSales** macro to the **Sort.RegionLastN** macro.

8. In the **Sort.RegionLastN** macro, change the cell reference in `Key2:=Range(“J6”)` to A6 and change the `Order2:=xlDescending` argument to `Order2:=xlAscending`.

9. Insert a blank line above the **End Sub** command and type the following macro command in the line:

10. Run the **Sort.RegionLastN** macro from the Visual Basic Editor. *(Hint: Remember to place the insertion point within the desired macro.)*

11. Close the Visual Basic Editor.

12. Notice that the list in the **Last Year** worksheet is sorted first by region and then by last name. View the list in **Print Preview** and notice the landscape page orientation.

13. Select the **Projections** worksheet and run the **Sort.RegionLastN** macro.
14. Delete the **Sort RegionLastN** macro.

15. Close the workbook without saving it.
LESSON 13 - CREATING MACRO BUTTONS

In this lesson, you will learn how to:

- Use a macro button
- Create a macro button
- Copy a macro button
- Format a macro button
- Move/Size a macro button
- Delete a macro button
- Add a macro to the Quick Access Toolbar
- Change a QAT macro button image
- Delete a macro button from the QAT
Using a Macro Button

Discussion

You can create a button in your worksheet that will run an associated macro. Macro buttons make it easier to run your macros and can help even an inexperienced user perform complex worksheet tasks.

You can draw a macro button anywhere on the worksheet, as well as create as many macro buttons as needed to perform different worksheet tasks.

Like other graphic objects, macro buttons are not stored in cells; they lie over the worksheet.

Procedures

1. Open a workbook containing a macro button and enable the macros.
2. Click the desired macro button.

Step-by-Step

From the Student Data directory, open CITY60.XLSM.

Use a macro button.

If necessary, enable the macros.

The macro used in the following step-by-step opens the ATLANTA.XLSX workbook but does not contain a file path. As a result, it will look in the current folder for the file. If you opened the CITY60.XLSM file by selecting it from the recently used files list, the current folder may not be the folder that contains the ATLANTA.XLSX file. Make sure that your current folder is set for the path containing the student data files by using the Open command in the Office menu to open the CITY60.XLSM file.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Click the desired macro button.</td>
<td>Click [Open Atlanta] on the worksheet</td>
</tr>
<tr>
<td><em>The associated macro runs.</em></td>
<td></td>
</tr>
</tbody>
</table>
Close ATLANTA.XLSX.

CREATING A MACRO BUTTON

Discussion

The Insert button in the Controls group on the Developer tab is used to create a macro button. You drag to draw a button of the required size and position on the worksheet. After you have drawn the macro button, you must assign a macro to it.

When the macro button is clicked, the macro attached to it runs.

Creating a macro button

You can resize a macro button at any time and reposition it anywhere on the worksheet.

Procedures

1. Open the desired workbook and enable the macros.
2. Select the Developer tab.
3. Select \begin{itemize}
\item \textbf{in the Controls group.}
\end{itemize}

4. Select the \textbf{Button (Form Control) button \begin{itemize}
\item \textbf{in the Form Controls section of the Controls gallery.}
\end{itemize}

5. Drag as needed to create the button.

6. Release the mouse button.

7. Select the macro you want to assign to the button.

8. Select \begin{itemize}
\item \textbf{.}
\end{itemize}

9. Select the existing button text.

10. Type the desired button text.

11. Select any cell in the worksheet to deselect the button.

\section*{Step-by-Step}

Create a macro button.

If necessary, enable the macros and make sure that your current folder is set for the path containing the student data files.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Developer tab. \textit{The Developer tab is displayed.}</td>
<td>Click Developer</td>
</tr>
<tr>
<td>2. Select the Insert button in the Controls group. \textit{The Controls gallery opens.}</td>
<td>Click Insert</td>
</tr>
<tr>
<td>3. Select the Button (Form Control) button in the Form Controls section of the Controls gallery. \textit{The mouse pointer changes into a crosshair when positioned in the worksheet.}</td>
<td>Click</td>
</tr>
<tr>
<td>4. Drag as needed to create the button. \textit{An outline appears as you drag.}</td>
<td>Drag from the top-left corner of cell H4 to the bottom-right corner of cell I4</td>
</tr>
</tbody>
</table>
Steps | Practice Data
--- | ---
5. Release the mouse button.
*The button appears on the worksheet, and the Assign Macro dialog box opens.* | Release the mouse button
6. Select the macro you want to assign to the button.
*The macro name appears in the Macro name box.* | Click **Chicago**
7. Select **OK**.
*The Assign Macro dialog box closes and the macro is assigned to the button.* | Click **OK**
8. Select the button text.
*The button text is selected.* | Drag to select the **Button 2** text
9. Type the desired button text.
*The text appears on the button.* | Type **Open Chicago**
10. Select any cell in the worksheet to deselect the button.
*The button is deselected.* | Click anywhere in the worksheet

Click the macro button to test it. Notice that the **Chicago** workbook opens. Close **CHICAGO.XLSX**.

**COPYING A MACRO BUTTON**

**Discussion**

You can create new macro buttons quickly by copying existing ones. This option is useful in adding several similar macro buttons to a worksheet. When you copy a macro button, the pasted macro buttons will always be the same size and shape as the copied macro button.

After copying the button, you can assign a different macro to the copy and then edit the text on the button.
Procedures

1. Open the desired workbook and enable the macros.
2. Hold [Ctrl] and drag the macro button you want to copy to where you want the copy to appear.
3. Release the mouse button.
5. Select a cell in the worksheet to deselect the macro button copy.
6. Right-click the macro button copy.
7. Select Assign Macro from the shortcut menu.
8. Select the desired macro from the list in the Assign Macro dialog box.
9. Select OK.
10. Select the text you want to edit on the macro button copy.
11. Type the desired text.
12. Select a cell in the worksheet to deselect the macro button copy.

Step-by-Step

Copy a macro button.

If necessary, enable the macros, show the Developer tab in the Ribbon and make sure that your current folder is set for the path containing the student data files.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Hold [Ctrl] and drag the macro button you want to copy to where you want the copy to appear. The mouse pointer changes into an arrow with a plus sign and an outline appears as you drag.</td>
<td>Hold [Ctrl] and drag the Open Chicago button into cell H6</td>
</tr>
<tr>
<td>2. Release the mouse button. A copy of the macro button appears on the worksheet.</td>
<td>Release the mouse button</td>
</tr>
</tbody>
</table>
**Steps** | **Practice Data**
---|---
*The mouse pointer returns to its normal shape.* | Release the [Ctrl] key

4. Select any cell in the worksheet to deselect the macro button copy.  
*The macro button copy is deselected.* | Click anywhere in the worksheet

5. Right-click the macro button copy.  
*A shortcut menu opens.* | Right-click the Open Chicago copy

6. Select **Assign Macro** from the shortcut menu.  
*The Assign Macro dialog box opens.* | Click **Assign Macro**

7. Select the desired macro from the list in the Assign Macro dialog box.  
*The macro name appears in the Macro name box.* | Click **NewYork**

8. Select **OK**.  
*The Assign Macro dialog box closes.* | Click **OK**

9. Select the text you want to edit on the macro button copy.  
*The text is selected.* | Double-click **Chicago** on the Open Chicago copy

10. Type the desired text.  
*The text appears on the macro button.* | Type **New York**

11. Select a cell in the worksheet to deselect the macro button.  
*The macro button is deselected.* | Click anywhere in the worksheet

Click the **Open New York** macro button to test it. Notice that the **NEWYORK.XLSX** workbook opens.  
Close **NEWYORK.XLSX**.

---

**FORMATTING A MACRO BUTTON**

**Discussion**

You can format text on a macro button in much the same way as you format text on a worksheet. You can select the entire button and apply the same formats to all the button text, or you can select individual characters in the button text to format.
To select a macro button for editing, hold the [Ctrl] key as you click the macro button; otherwise, the macro attached to the button runs when you click it.

Procedures

1. Select the Home tab in the Ribbon.
2. To format all the button text, hold [Ctrl] and click the macro button you want to format.
4. Format the macro button text as desired.
5. Select any cell in the worksheet to deselect the macro button.
6. To format selected macro button text, hold [Ctrl] and click the macro button you want to format.
8. Select the text you want to format.
9. Format the selected text as desired.
10. Select a cell in the worksheet to deselect the macro button.

Step-by-Step

Format a macro button.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Home tab in the Ribbon. The Home tab is displayed.</td>
<td>Click Home</td>
</tr>
<tr>
<td>2. To format all the button text, hold [Ctrl] and click the macro button you want to format. The mouse pointer changes into an arrow with a plus sign and the macro button is selected.</td>
<td>Hold [Ctrl] and click the Open Chicago button</td>
</tr>
<tr>
<td>3. Release the [Ctrl] key. The mouse pointer returns to its normal shape.</td>
<td>Release the [Ctrl] key</td>
</tr>
</tbody>
</table>
### Steps

<table>
<thead>
<tr>
<th>Steps</th>
</tr>
</thead>
</table>
| 4. Format the macro button text as desired. 
   *All the macro button text is formatted accordingly.* |
| 5. Select any cell in the worksheet to deselect the macro button. 
   *The macro button is deselected.* |
| 6. To format selected macro button text, hold [Ctrl] and click the macro button you want to format. 
   *The mouse pointer changes into an arrow with a plus sign and the macro button is selected.* |
   *The mouse pointer returns to its normal shape.* |
| 8. Select the text you want to format. 
   *The text is selected.* |
| 9. Format the selected text as desired. 
   *The text is formatted.* |
| 10. Select any cell in the worksheet to deselect the macro button. 
   *The macro button is deselected.* |

### Practice Data

<table>
<thead>
<tr>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. Click ✋ in the Font group</td>
</tr>
<tr>
<td>5. Click anywhere in the worksheet</td>
</tr>
<tr>
<td>6. Hold [Ctrl] and click the Open Atlanta button</td>
</tr>
<tr>
<td>7. Release the [Ctrl] key</td>
</tr>
<tr>
<td>8. Double-click Atlanta</td>
</tr>
<tr>
<td>9. Click ✎ in the Font group</td>
</tr>
<tr>
<td>10. Click anywhere in the worksheet</td>
</tr>
</tbody>
</table>

#### Practice the Concept:
Bold the text on all macro buttons; then italicize the city names on each macro button.

---

## MOVING/SIZING A MACRO BUTTON

### Discussion

You can move or size a macro button in the same way that you move or size any graphic object on a worksheet. In order to move or size a macro button, you must first select it.

---

To select a macro button, hold the [Ctrl] key as you click it; otherwise, the macro attached to the macro button runs when you click it.
Procedures

1. To move a macro button, hold [Ctrl] and click the macro button to select it.
2. Release the [Ctrl] key.
3. Point to the shaded border around the button and drag the selected macro button to the desired location on the worksheet.
4. Release the mouse button.
5. Select a cell in the worksheet to deselect the macro button.
6. To size a macro button, hold [Ctrl] and click the macro button to select it.
8. Drag any sizing handle in the desired direction.
9. Release the mouse button.
10. Select a cell in the worksheet to deselect the macro button.

Step-by-Step

Move and size a macro button.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. To move a macro button, hold [Ctrl] and click the macro button to select it. &lt;br&gt;&lt;i&gt;The mouse pointer changes into an arrow with a plus sign and the macro button is selected.&lt;/i&gt;</td>
<td>Hold [Ctrl] and click the &lt;b&gt;Open Atlanta&lt;/b&gt; button</td>
</tr>
<tr>
<td>2. Release the [Ctrl] key. &lt;br&gt;&lt;i&gt;The mouse pointer returns to its normal shape.&lt;/i&gt;</td>
<td>Release the [Ctrl] key</td>
</tr>
<tr>
<td>3. Point to the shaded border around the button and drag the selected macro button to the desired location on the worksheet. &lt;br&gt;&lt;i&gt;The mouse pointer changes into a four-headed arrow and an outline of the button appears as you drag.&lt;/i&gt;</td>
<td>Drag the &lt;b&gt;Open Atlanta&lt;/b&gt; button below the &lt;b&gt;Open New York&lt;/b&gt; button</td>
</tr>
</tbody>
</table>
### Steps

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. Release the mouse button.</td>
<td>Release the mouse button</td>
</tr>
<tr>
<td><em>The button moves to the new location.</em></td>
<td></td>
</tr>
<tr>
<td>5. Select any cell in the worksheet to deselected the macro button.</td>
<td>Click anywhere in the worksheet</td>
</tr>
<tr>
<td><em>The macro button is deselected.</em></td>
<td></td>
</tr>
<tr>
<td>6. To size a macro button, hold [Ctrl] and click the macro button to</td>
<td>Hold [Ctrl] and click the Open Chicago button</td>
</tr>
<tr>
<td>select it.</td>
<td></td>
</tr>
<tr>
<td><em>The mouse pointer changes into an arrow with a plus sign and the macro button is selected.</em></td>
<td></td>
</tr>
<tr>
<td>7. Release the [Ctrl] key.</td>
<td>Release the [Ctrl] key</td>
</tr>
<tr>
<td><em>The mouse pointer returns to its normal shape.</em></td>
<td></td>
</tr>
<tr>
<td>8. Drag any sizing handle in the desired direction.</td>
<td>Drag the top-center selection handle of the Open Chicago button to the top of row 3</td>
</tr>
<tr>
<td><em>The mouse pointer changes into a two-headed arrow and an outline appears as you drag.</em></td>
<td></td>
</tr>
<tr>
<td>9. Release the mouse button.</td>
<td>Release the mouse button</td>
</tr>
<tr>
<td><em>The size of the macro button changes accordingly.</em></td>
<td></td>
</tr>
<tr>
<td>10. Select a cell in the worksheet to deselect the macro button.</td>
<td>Click anywhere in the worksheet</td>
</tr>
<tr>
<td><em>The macro button is deselected.</em></td>
<td></td>
</tr>
</tbody>
</table>

### DELETING A MACRO BUTTON

#### Discussion

You can delete any macro button you no longer need. Deleting unused macro buttons protects you from accidentally running a macro that is outdated or no longer works correctly.

When you delete a macro button, only the button is deleted; the macro remains available and can be viewed and edited in the Visual Basic Editor or run from the Macros button in the View or Developer tab.
Procedures

1. Hold [Ctrl] and click the macro button you want to delete.
2. Release the [Ctrl] key.
3. Press [Delete].

Step-by-Step

Delete a macro button.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Hold [Ctrl] and click the macro button you want to delete.</td>
<td>Hold [Ctrl] and click the Open Chicago button</td>
</tr>
<tr>
<td>The mouse pointer changes into an arrow with a plus sign and the macro button is selected.</td>
<td></td>
</tr>
<tr>
<td>2. Release the [Ctrl] key.</td>
<td>Release the [Ctrl] key</td>
</tr>
<tr>
<td>The mouse pointer returns to its normal shape.</td>
<td></td>
</tr>
<tr>
<td>3. Press [Delete].</td>
<td>Press [Delete]</td>
</tr>
<tr>
<td>The macro button is deleted.</td>
<td></td>
</tr>
</tbody>
</table>

Adding a Macro to the Quick Access Toolbar

Discussion

You can add a macro to the Quick Access Toolbar (QAT). When you add a macro to the QAT, Excel assigns a special custom button to the macro. If you add several macros to the QAT, the button for each macro will look the same. You can easily modify the buttons and assign a different image to each button from a gallery of button designs.

When you add a macro to the QAT, you can choose whether to add it to the QAT for all documents or just for the current document. Macros stored in the Personal Macro Workbook are available for all documents and can be added to the QAT for all documents. Macros stored in an individual workbook are only available when that workbook is open; you should add these macros to the QAT for the specific workbook in which they are used.
Adding a macro to a Quick Access Toolbar

If you add a macro that is stored in a specific workbook to the Quick Access Toolbar for all documents, although the macro button will always be visible on the QAT, you will only be able to run the macro when the workbook in which it is stored is open. If you try to run the macro in a workbook that does not contain the macro, a Microsoft Office Excel message will open warning you that the macro cannot run.

Procedures

1. Select the **Customize Quick Access Toolbar** button to the right of the Quick Access Toolbar.
2. Select **More Commands**.
3. Select the **Choose commands** from list.
4. Select **Macros**.
5. If the macro is stored in a specific workbook, select the **Customize Quick Access Toolbar** list.
6. Select the name of the workbook in which the macro is stored.
7. Select <Separator> from the Commands list, if desired.

8. Select [Add >>].

9. Select the desired macro from the Commands list.

10. Select [Add >>].

11. Continue adding macros to the QAT, as desired.

12. Select [OK].

---

**Step-by-Step**

Add a macro to the Quick Access Toolbar.

If necessary, enable the macros.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Customize Quick Access Toolbar button to the right of the Quick Access Toolbar. The Customize Quick Access Toolbar menu opens.</td>
<td>Click ✎</td>
</tr>
<tr>
<td>2. Select More Commands. The Excel Options dialog box opens with the Customize page displayed.</td>
<td>Click More Commands</td>
</tr>
<tr>
<td>3. Select the Choose commands from list. A list of available options is displayed.</td>
<td>Click Choose commands from ✏</td>
</tr>
<tr>
<td>4. Select Macros. A list of available macros is displayed in the Commands list.</td>
<td>Click Macros</td>
</tr>
<tr>
<td>5. If the macro is stored in a specific workbook, select the Customize Quick Access Toolbar list. A list of open workbooks is displayed.</td>
<td>Click Customize Quick Access Toolbar ✏</td>
</tr>
<tr>
<td>6. Select the name of the workbook in which the macro is stored. The workbook name appears in the Customize Quick Access Toolbar box.</td>
<td>Click For CITY60.XLSM</td>
</tr>
</tbody>
</table>
### Steps

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>7. Select <code>&lt;Separator&gt;</code> from the Commands list, if desired. <em>The <code>&lt;Separator&gt;</code> is selected.</em></td>
<td>Click <code>&lt;Separator&gt;</code></td>
</tr>
<tr>
<td>8. Select the <strong>Add</strong> button. <em>The <code>&lt;Separator&gt;</code> is added to the Customize Quick Access Toolbar For CITY60.XLSM list.</em></td>
<td>Click <strong>Add</strong></td>
</tr>
<tr>
<td>9. Select the desired macro from the Commands list. <em>The macro name is selected.</em></td>
<td>Click <strong>Company</strong></td>
</tr>
<tr>
<td>10. Select the <strong>Add</strong> button. <em>The macro is added to the Customize Quick Access Toolbar For CITY60.XLSM list.</em></td>
<td>Click <strong>Add</strong></td>
</tr>
<tr>
<td>11. Continue adding macros to the QAT, as desired. <em>The macros are added to the Customize Quick Access Toolbar For CITY60.XLSM list.</em></td>
<td>Follow the instructions shown below the table before continuing on to the next step</td>
</tr>
<tr>
<td>12. Select <strong>OK</strong>. <em>The Excel Options dialog box closes and the macro buttons and separator appear in the Quick Access Toolbar.</em></td>
<td>Click <strong>OK</strong></td>
</tr>
</tbody>
</table>

Add the **Urban_Update** macro to the Quick Access Toolbar For CITY60.XLSM.

*Return to the table and continue on to the next step (step 12).*

Point to each of the macro buttons in the Quick Access Toolbar. Notice that the name of each macro appears in a ScreenTip.

Select cell A1 in the worksheet, if necessary, then click the **Company** macro button on the Quick Access Toolbar. Notice that the macro inserts the company name into the selected cell.

---

**Changing a QAT Macro Button Image**

#### Discussion

When a macro is added to the Quick Access Toolbar, the same button image is displayed for each macro. You can easily change the button image; Excel provides numerous images from which you can choose.
When you hover the mouse pointer over a macro button, a **ScreenTip** displays the name of the macro. You can change the text that appears in the **ScreenTip**, if desired, to describe what the macro does.

![Changing a button image](image)

You can also open the **Customize** page in the Excel Options dialog box by right-clicking on a button in the **Quick Access Toolbar** and selecting **Customize Quick Access Toolbar**.

**Procedures**

1. Select the **Customize Quick Access Toolbar** button to the right of the **Quick Access Toolbar**.
2. Select **More Commands**.
3. If the macro is stored in a specific workbook, select the **Customize Quick Access Toolbar** list.
4. Select the name of the workbook in which the macro is stored.
5. Select the desired macro from the **Customize Quick Access Toolbar** list.
6. Select Modify...

7. Select the desired button image.

8. Select the text in the Display name box.

9. Type the desired display name for the macro button.

10. Select OK.

11. Select OK.

**Step-by-Step**

Change a button image.

If necessary, enable the macros in the workbook.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Select the Customize Quick Access Toolbar button to the right of the Quick Access Toolbar. The Customize Quick Access Toolbar menu opens.</td>
<td>Click <img src="image1.png" alt="Click" /></td>
</tr>
<tr>
<td>2. Select More Commands. The Excel Options dialog box opens with the Customize page displayed.</td>
<td>Click More Commands</td>
</tr>
<tr>
<td>3. If the macro is stored in a specific workbook, select the Customize Quick Access Toolbar list. A list of open workbooks is displayed.</td>
<td>Click Customize Quick Access Toolbar <img src="image2.png" alt="Click" /></td>
</tr>
<tr>
<td>4. Select the name of the workbook in which the macro is stored. The workbook name appears in the Customize Quick Access Toolbar box.</td>
<td>Click For CITY60.XLSM</td>
</tr>
<tr>
<td>5. Select the desired macro from the Customize Quick Access Toolbar list. The macro button is selected</td>
<td>Click <img src="image3.png" alt="Click" /> Company</td>
</tr>
<tr>
<td>6. Select the Modify button. The Modify Button dialog box opens displaying a gallery of available button images.</td>
<td>Click Modify...</td>
</tr>
</tbody>
</table>
Lesson 13 - Creating Macro Buttons  
Excel 2007 - Lvl 4

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>7. Select the desired button image. The button image is selected.</td>
<td>Click 🏡 (row 2, column 8)</td>
</tr>
<tr>
<td>8. Select the text in the <strong>Display name</strong> box. The text is selected.</td>
<td>Double-click the text <strong>Company</strong></td>
</tr>
<tr>
<td>9. Type the desired display name for the macro button. The text is</td>
<td>Type <strong>Insert company name in current cell</strong></td>
</tr>
<tr>
<td>displayed in the <strong>Display name</strong> box.</td>
<td></td>
</tr>
<tr>
<td>10. Select OK. The Modify Button dialog box closes and the selected</td>
<td>Click <img src="ok.png" alt="OK" /></td>
</tr>
<tr>
<td>button image and display name appear in the **Customize Quick</td>
<td></td>
</tr>
<tr>
<td>Access Toolbar For <strong>CITY60.XLSM</strong> list.</td>
<td></td>
</tr>
<tr>
<td>11. Select OK. The Excel Options dialog box closes and the new</td>
<td>Click <img src="ok.png" alt="OK" /></td>
</tr>
<tr>
<td>button image is displayed on the <strong>Quick Access Toolbar</strong>.</td>
<td></td>
</tr>
</tbody>
</table>

**Practice the Concept:** Right-click on any button on the **Quick Access Toolbar** and select **Customize Quick Access Toolbar**. Select **For CITY60.XLSM** from the **Customize Quick Access Toolbar** list. Click on the **Urban_Update** macro, then click the **Modify** button.

Select the **Modify** button and click on the multisheet symbol (row 10, column 5). Change the **Display name** text to **Consolidate the latest sales data**. Click **OK** to close the Modify Button dialog box, then click **OK** to close the Excel Options dialog box.

Click the **Consolidate the latest sales data** button on the **Quick Access Toolbar**.

---

**DELETING A MACRO BUTTON FROM THE QAT**

**Discussion**

You can remove a button easily from the **Quick Access Toolbar** if you no longer use it. If the button is attached to the **Quick Access Toolbar** for a specific workbook, the workbook must be open to remove the button.
When you delete a macro button from the Quick Access Toolbar, only the button is deleted; the macro remains available and can be viewed and edited in the Visual Basic Editor or run from the Macros button in the View or Developer tab.

Procedures

1. Right-click the button on the Quick Access Toolbar that you want to remove.

2. Select Remove from the Quick Access Toolbar.

Step-by-Step

Delete a macro button from the Quick Access Toolbar.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Practice Data</th>
</tr>
</thead>
</table>
| 1. Right-click the button on the Quick Access Toolbar that you want to remove.  
  The Quick Access Toolbar shortcut menu appears. | Right-click  ![Image](image.png) |
| 2. Select Remove from the Quick Access Toolbar.  
  The selected button is removed from the Quick Access Toolbar. | Click Remove from the Quick Access Toolbar |

Practice the Concept: Right-click on the Insert company name in current cell macro button and select Remove from the Quick Access Toolbar from the shortcut menu. Then, right-click on the Separator and remove it from the Quick Access Toolbar.

Close CITY60.XLSM.
EXERCISE

CREATING MACRO BUTTONS

Task

Create and work with macro buttons on a worksheet.

1. Open EXER85.XLSM and enable the macros.
2. Display the Developer tab.
3. Draw a macro button in the range M7:M8. Assign the Eastern macro to the macro button and change the button text to Eastern Sales. Then, italicize the button text.
4. Deselect the Eastern Sales button and click it to test it.
5. Drag to create a copy of the Eastern Sales button. Position its top, left corner at the top, left corner of cell M3. Assign the ClearOutput macro to the macro button and rename it Clear Report.
6. Deselect the Clear Report button and then click it to test it.
7. Move the Clear Report button below the Eastern Sales button, aligning it to the top, left corner of cell M10.
8. Widen the Clear Report button to extend across both columns M and N.
9. Delete both macro buttons.
11. Customize the Quick Access Toolbar For EXER85.XLSM and add the Eastern, Southern, and Western macros.
12. Modify the Western macro button and assign the left-pointing arrow symbol (row 2, last column) to the button. Change the Display name to Western Sales.
13. Modify the Eastern and Southern macros in a similar way, assigning appropriate arrow symbols to the buttons and changing the Display name to add the word Sales.
14. Remove all three macro buttons from the Quick Access Toolbar.
15. Close the workbook without saving it.
INDEX

Absolute references
  using in macros, 198, 199
Comments
  entering for macros, 208, 209, 210
Custom views
  creating, 128
  deleting, 130, 131
  displaying, 129, 130
Data
  consolidating, 102
  consolidating by category, 102, 103, 104
  consolidating by position, 106, 107, 108
  limiting the precision of numbers, 38, 39, 40
  solving problems, 112, 113, 114
Data series
  creating date, 5, 6
  creating growth trend, 9, 10, 11
  creating linear, 3, 4
  using a stop value, 7, 8
  using the Series command, 2
Data tables, 92
  creating one-variable, 93, 94, 95
  creating two-variable, 96, 97
  placing formulas in, 92
Error Checking button
  using, 156, 157, 158
Errors
  checking, 145, 147
  tracing cells causing error values, 148, 149, 150
  using the Trace Error button, 145, 147
  using the Trace Error command, 148, 149, 150
Evaluate Formula button
  using, 151, 153
Formula auditing
  display dependent arrows, 138, 140
  display formula results in worksheet cells, 136, 137
  display formulas in worksheet cells, 136, 137
  display precedent arrows, 141, 143
  remove all arrows, 138, 140, 141, 143
  remove dependent arrows, 138, 140
  remove precedent arrows, 141, 143
  trace dependents, 138, 140
  trace precedents, 141, 143
  using the Error Checking button, 156, 157, 158
  using the Evaluate Formula button, 151, 153
  using the Trace Error button, 145
using the Trace Error command, 148, 149, 150
using the Watch Window, 159, 161

Functions
HLOOKUP, 14, 18, 20
IF, 21, 23, 24, 28, 29, 30
ISERROR, 28, 29, 30
LOOKUP, 14
nested IF, 25, 26
ROUND, 36, 37
using AND conditions with IF, 31, 32
using lookup, 14
using OR conditions with IF, 33, 34
VLOOKUP, 14, 16, 17

Goal Seek, 86, 87

Macro-Enabled Workbooks, 168

Macros
adding a folder to the Trusted Locations list, 172, 174
adding a macro button to the Quick Access Toolbar, 234, 235, 236
assigning a shortcut key, 196, 197
changing macro security settings, 169, 171
changing the button image, 237, 238, 239
copying a macro button, 227, 228
copying commands, 211, 212
creating a macro button, 225, 226
defining, 168
deleting, 201, 202
deleting a macro button, 233, 234
deleting a macro button from the Quick Access Toolbar, 240, 241
displaying the Developer tab, 183, 184
ingoing commands, 213, 214
entering comments, 208, 209, 210
formatting button text, 229, 230
moving a macro button, 231, 232
opening a workbook containing, 176, 178
opening the Visual Basic Editor window, 184, 186
recording, 192, 193, 194
resizing a macro button, 231, 232
running, 179, 180, 181
running from the Code window, 218
typing commands, 215, 216, 217
using a macro button, 224
using a shortcut key, 181, 182
using absolute references, 198, 199
using relative references, 198, 199
using the Visual Basic Editor window, 187, 188
writing new, 206, 207

Numbers
limiting precision, 38, 39, 40
rounding, 36, 37

PivotChart reports
creating, 66, 67, 68
PivotTable reports
   adding new fields, 54, 55
   changing the summary function, 52, 53
   creating report filter pages, 62, 63, 64
   deleting fields, 61, 62
   filtering, 49, 50
   formatting, 65, 66
   hiding items, 59, 60
   moving fields, 56, 57
   publishing to the Web, 69, 70
   refreshing, 51, 52
   selecting a page field item, 49, 50
   unhiding items, 59, 60
PivotTables
   adding fields to a PivotTable, 46, 47
   Creating a PivotTable, 44, 45
   Expand and Collapse buttons, 58, 59
Quick Access Toolbar
   customizing, 234, 235, 236, 237, 238, 239
   removing a macro button, 240, 241
Relative references
   using in macros, 198, 199, 200
Report filter pages
   creating, 62, 63, 64
Scenario Manager, 76
Scenarios
   creating, 76, 77, 78
   creating a summary report, 83, 84, 85
   displaying, 79, 80
   editing, 81, 82
   saving Solver solutions as, 117
   using the Scenario Manager, 76
   viewing Solver solutions, 122, 123
Shortcut keys
   assigning to macros, 196, 197
Solver, 112
   changing constraints, 119
   creating reports, 120, 121
   restoring the original values, 118
   saving a solution as a scenario, 117
   using, 113, 114
   viewing solutions using scenarios, 122, 123
Trace Error button
   using, 145, 147
Views
   creating custom, 128
   creating normal, 126, 127
   deleting custom, 130, 131
   displaying, 129, 130
   using, 126
Watch Window
using, 159, 161
What-if analysis, 76
Workbooks
  consolidating, 102
  opening those containing macros, 178
Worksheets
  consolidating, 102
  consolidating by category, 102, 103, 104
  consolidating by position, 106, 107, 108