Microsoft Excel 2016

Level 3

Instructor Edition

- ✓ Maps to Microsoft Office Specialist (MOS) objectives for exam for Excel 2016
- ✓ Realistic, hands-on exercises
- ✓ Downloadable ancillaries at 30bird.com

Excel 2016 Level 3

Instructor Edition

30 Bird Media
510 Clinton Square
Rochester NY 14604
www.30Bird.com

Excel 2016 Level 3

Instructor Edition

CEO, 30 Bird Media: Adam A. Wilcox

Series designed by: Clifford J. Coryea, Donald P. Tremblay, and Adam A Wilcox

Managing Editor: Donald P. Tremblay

Instructional Design Lead: Clifford J. Coryea

Copyeditor: Robert S. Kulik

Keytester: Kurt J. Specht

Table of Contents

Introduction

Welcome to *Excel 2016 Level 3*. This course builds on the concepts and skills of our Level 1 and Level 2 courses to provide advanced tools for solving real-world problems in Microsoft Excel 2016: lookup and decision-making functions, auditing and error-handling, array functions, date and text functions, importing and exporting, the Power Pivot Data Model, what-if-analysis, and macros. The three levels of our Excel 2016 courses map to the objectives of the Microsoft Office Core and Expert exams for Excel 2016. Objective coverage is marked throughout the course, and you can download an objective map for the series from http://www.30bird.com.

You will benefit most from these courses if you want to use Excel 2016 to perform real-world tasks such as getting information from large amounts of data from sources inside and outside of Excel, creating output that varies according to conditions, manipulating dates and text, and automating repetitive tasks. If you intend to take a Microsoft Office Specialist or Expert exam for Excel, this course will complete your coverage of all the objectives for both exams.

The course assumes you know how to use a computer, that you're familiar with Microsoft Windows, and that you've taken the Level 1 and Level 2 courses or have equivalent experience with Excel. The exercises are more detailed and complex than those in the previous levels.

After you complete this course, you will know how to:

- Use function to make decisions and to return information from arrays and tables

- Find and trap errors, control formula options, and use array formulas and functions

- Perform calculations on dates, manipulate text, and use some statistical and financial functions

- Import or create a connection to external data, use Power Pivot and cube functions to access information in the Power Pivot Data Model, and export Excel worksheets to other formats

- Perform what-if analyses by using a watch window, scenarios, and the Goal Seek utility; and use some of the tools available in the Analysis Toolpak

- Record and run macros, and create a simple form

This is the third and final course in this series.

Course setup

To complete this course, each student and instructor needs to have a computer running Excel 2016. Setup instructions and activities are written assuming Windows 10; however, with slight modification the course will work using Windows XP Service Pack 3, Windows Vista Service Pack 1, Windows 7, or Windows 8 or 8.1.

Hardware requirements for Windows 10 course setup include:

- 1 GHz or faster processor

- 2 GB RAM

- 25 GB total hard drive space (50 GB or more recommended)

- DirectX 10 (or later) video card or integrated graphics, with a minimum of 128 MB of graphics memory

- Monitor with 1280x800 or higher resolution

Software requirements include:

- Windows 10 (or alternative as above)

- Microsoft Excel 2016 or any Microsoft Office 2016 edition including Excel (you will need one of the editions that includes PowerPivot in order to teach the exercises that cover that topic)

- The Excel 2016 Level 3 data files and PowerPoint slides, available at http://www.30bird.com

- Microsoft Access 2016 for one exercise in the chapter on importing and exporting, and one in the chapter on macros and forms (which can also be skipped or demonstrated by the instructor)

Network requirements include:

- An internet connection (you will need this to be able to complete the synthesis exercise for the chapter on importing and exporting)

Because the exercises in this course include viewing and changing some Excel defaults, it's recommended that you begin with a fresh installation of the software. But this is certainly not necessary. Just be aware that if you are not using a fresh installation, some exercises might work slightly differently and some screens might look slightly different. In particular, the Power Pivot and Analysis Toolpak add-ins are loaded during the course, two buttons are added to the Quick Access toolbar, and the Developer ribbon tab is shown.

1. Install Windows 10, including all recommended updates and service packs. Use a different computer and user name for each student.

2. Install Microsoft Excel 2016, using all defaults during installation.

3. Update Excel or Office using Windows Update.

4. Copy the Excel 2016 Level 3 data files to the Documents folder.

Chapter 1: Logical and Lookup Functions

You will learn:

- About using functions to make decisions, how to construct IF functions, and how to use SUMIFS to calculate sums based on criteria
- How to use lookup and reference functions to return values from lists based on conditions and positions

Learning time: 75 minutes

Module A: Decision-making functions

Decision-making functions test a logical condition, then return a value or take an action. For example, you might want to know the sales figures that result if a certain condition is true. Excel has many functions, such as IF, AND, and SUMIFS, that evaluate conditions before returning a value.

You will learn how to:

- Use IF to return one value when a condition is true, and another when that condition is false

- Nest IF functions or use other logical functions such as AND and OR to evaluate more complex decision scenarios

- Use SUMIF and SUMIFS to calculate totals only when one or more conditions are met

The IF function

The IF function does what it sounds like it does: it tests a condition, and *if* that condition is true, it returns one value; if not, it returns another or does nothing. Consider this table.

1 The first thing an IF function does is to test a *condition*. Here, it would ask, "Is the sales amount greater than or equal to the quota?"

2 The *value if true* here is $1,000. In plain language, this means "pay a bonus of $1,000 if the rep made quota."

3 The *value if false* here is zero.

The IF function can do what that table did with a single function. The function's syntax looks like this.

IF(logical_test, value_if_true, [value_if_false])

In practice, it looks like this.

The logical test "B2>B1" tests whether the sales met the quota. If so, the value if true, 1000, is returned. If not, the value if false, 0, is returned.

Building an IF function

When you enter functions with multiple arguments, use Excel's screen tips to help you. With functions that test a condition, such as IF, you might want to build the condition as a separate formula first, just to be sure you understand how to write the logical_test argument.

Exam Objective: MOS Excel Core 4.2.1

1. Select the cell where you want the function.

2. Type =IF(.

   As soon as Excel recognizes the function, it displays a screen tip that tells you which arguments you need to enter. The first is a logical test.

   =IF(
   IF(**logical_test**, [value_if_true], [value_if_false])
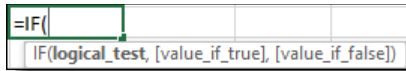
3. Type an expression that will result in TRUE or FALSE, then type a comma.
   If you need help, it's often a good idea to construct this expression in a cell before trying to create the IF function.

   =IF(B2>B1,
   IF(logical_test, **[value_if_true]**, [value_if_false])
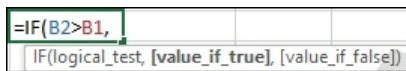
   The next argument, value_if_true, is highlighted.

4. Enter the value you want if the logical_test argument is TRUE, then type a comma.

5. Enter the value you want if the logical_test argument is FALSE.

6. Type a closing parentheses, then press **Enter**.

Exercise: Using IF to determine sales commissions

Exam Objective: MOS Excel Core 4.2.1

Do This	How & Why
1. Open `Decisions`.	From the `Logical and Lookup Functions` data folder. The first worksheet contains sales information for several sales reps. It also shows the quota for each sales rep, and has commission rates.
2. In D10:D19, enter expressions to test whether the rep made quota.	
a) Select D10:D19.	
b) Type =, then click B10.	You need to test whether the sales total, in B10, is greater than or equal to the quota for this rep.
c) Type >=, then click C10.	The condition should look like this.
d) Press **Ctrl+Enter**.	To enter the formula in the entire selected range. The cells display "TRUE" or "FALSE," depending on the total sales and quota values for each rep.
3. Use IF to give the reps a 4% commission only if they made quota.	
a) Select E10:E19.	
b) Type =IF(.	A screen tip appears, prompting you to enter the logical text argument.
c) Type B10>=C10,.	Include the comma. This is the expression you used in column D.
d) Type 4%*, then click B10.	The function should look like this, so far.
	`=IF(B10>=C10,4%*B10`
e) Type a comma, then type 0.	To enter the value_if_false argument. That is, to give no commission if the rep does not make quota.
f) Type a closing parentheses, then press **Ctrl+Enter**.	To enter the function in the selected range.
4. Save the workbook as `My Decisions`.	

My Decisions with commission functions using IF

More complex decisions

If you want to make a series of decisions in a function, you can *nest* functions together. That simply means using a function as an argument in another function. For example, you might want to test a condition using IF, then, for the value_if_false argument, test another condition with another IF function.

You can also use *logical functions* such as AND and OR to test conditions in combination.

Nesting IF functions

To nest an IF function within another IF function, simply use a second IF function as one of the value_if arguments (or both). Here's an example of how that might look.

IF(logical_test1, value_if_true,IF(logical_test2,value_if_true,value_if_false))

You can nest as many IF functions as you like, although when you nest several, it's likely that there's a better way to accomplish your goal.

AND, OR, and NOT

Use AND and OR to test combinations of conditions in a single expression. Here is the syntax for these functions.

Exam Objective: MOS Excel Expert 3.1.1

=AND(logical1, logical2,...)
=OR(logical1, logical2,...)

AND returns TRUE only when *all* of the logical arguments evaluate to TRUE. OR returns TRUE when *any one* of the logical argument evaluates to TRUE.

NOT returns TRUE when the expression in its argument is false. This can also be useful.

Here are some examples.

	A	B	C
1	7	=OR(A1<8,A2>10)	TRUE
2	9	=OR(A2<8,A3>10)	FALSE
3	7	=AND(A3<8,A4>10)	TRUE
4	11	=AND(A4>8,A4>10)	TRUE
5	10	=NOT(A5<10)	TRUE

The OR function in B1 returns TRUE because, even though 7 is not greater than 10 (the second condition), it is less than 8. OR requires only one condition to be met. Neither condition is met in B2, so that OR function returns FALSE.

B3 returns FALSE because, even though 7 is less than 8, it is not greater than 10. For AND to return TRUE, both conditions must be met, as they are in B4.

B5 returns TRUE because the condition in the NOT function, A5<10, is FALSE. NOT returns the opposite of the logical expression's value.

Exercise: Using nested functions to make more complex decisions

`My Decisions` is open.

Exam Objective: MOS Excel Expert 3.1.1, 3.1.2

Do This	How & Why
1. Observe the commission-rate information.	In D3:E4. Reps should get one commission rate, 4%, if they meet quota, but a higher rate if they reach 110% of quota.
2. Use a nested IF function to give the correct commission rates.	
a) In cell E10, select the second argument of the IF function.	This is the value_if_true argument. You'll replace it with a function that will give a higher commission rate if the rep has reached 110% of quota, or the lower rate otherwise. =IF(B10>=C10,4%*B10,0) IF(logical_test, [value_if_true], [value_if_false])
b) Type or enter `IF(B10>=(1.1*C10), $E$4*B10,$E$3*B10).`	It's probably best to click the cells to enter the cell references. Note that the references to the commission rates in cells $E$3 and $E$4 should be absolute, because you don't want those to change when you copy the formula.
c) Enter the formula.	It should look like this. The rep still gets no commission because he hasn't met the quota, which is the logical test of the outer IF function. =IF(B10>=C10,IF(B10>(1.1*C10),$E$4*B10,$E$3*B10),0) |	D	E	F	G
---	---	---	---	---
ta	Met Quota?	Comm.		
000	FALSE	$ -			
3. Note the commission amount for Hernandez.	It is $624. This is based on the original IF function, which gives a 4% commission to all reps who meet quota. But Hernandez's sales total is well over 110% of quota.
4. Copy the formula in cell E10 to E11:E19.	Hernandez now gets the higher commission rate, 5%, and has a commission of $780.

Do This	How & Why
5. Observe the team quota.	Let's say that management wants to encourage teamwork by giving commissions only when the entire team goal is reached. Now, to get any commission, a rep has to meet her own quota *and* the team has to meet their goal.
6. Edit the commission formula to have the outer IF function test both team and individual quotas. a) Select E10:E19. b) Select the first logical_test argument, as shown. c) Type or enter `AND(B10>=C10,$B$20>=$E$2)`. d) Press **Ctrl**+**Enter**.	You'll enter all the formulas in one step. `=IF(B10>=C10,IF(B10>(1.1*C10),$E$4*B10,$E$3*B10),0)` Again, you should use the mouse to enter the cell references, taking care to make the references to $B$20 (the grand total sales for the team) and $E$2 (the team quota) absolute. The AND function will test that both conditions are true. That is, both that the rep has met her individual goal, and that the team has met theirs. To enter the function in the range. Nothing seems to change, because the team goal has been met.
7. Change the team goal in E2 to $140,000.	Because the team goal hasn't been met, none of the reps get commission.
8. Change the team goal to $130,000.	The reps who have made quota once again receive commission, because the team goal has been met. And the reps who have exceeded 110% of their goal get a higher commission rate. The one formula makes a series of decisions.
9. Save the workbook.	

Commissions calculated using nested IF functions and AND

	Team quota	$ 130,000
	100% comm:	4%
	110% comm:	5%

Sales and Commission

2014

Rep	Total Sales	Quota	Met Quota?	Comm.
Blackwell	$ 3,046	$ 4,000	FALSE	$ -
Daniels	$ 9,207	$ 8,500	TRUE	$ 368
Franklin	$ 12,539	$ 12,000	TRUE	$ 502
Hernandez	$ 15,600	$ 13,000	TRUE	$ 780
Lloyd	$ 23,392	$ 22,000	TRUE	$ 936
McCanney	$ 20,067	$ 22,000	FALSE	$ -
Patterson	$ 7,113	$ 6,500	TRUE	$ 285
Sanchez	$ 12,905	$ 10,000	TRUE	$ 645
Schiller	$ 19,240	$ 20,000	FALSE	$ -
Westlein	$ 16,846	$ 18,000	FALSE	$ -
Total	**$ 139,955**			

Conditional calculation

When you have a list of data, you might want to perform calculations for only certain values in the list. For example, you might want to know the sum of sales only for invoices to customers in a certain region. You could use subtotals, or perhaps a pivot table, to get that information, but you can also use *conditional functions*. Excel includes several of these.

- SUMIF

- SUMIFS

- COUNTIF

- COUNTIFS

- AVERAGEIF

- AVERAGEIFS

The functions without an "S" calculate according to a single condition, while the functions with an "S" can use many conditions in an AND relationship.

Using SUMIF

SUMIF calculates the sum of a range of values based upon whether a condition you specify is met. For example, you might want the sum of rep sales totals if the rep is in a particular region. You define three arguments for the SUMIF function (COUNTIF and AVERAGEIF work in the same way).

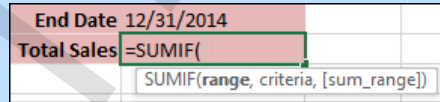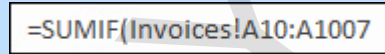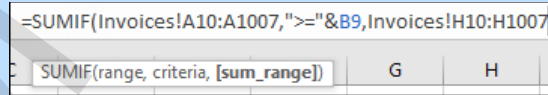**Exam Objective:** MOS Excel Core 4.2.2, 4.2.3,4.2.4

- The *range argument* is the range that you want to test in your criterion. In the example here, that would be the range containing the region data.

- The *criteria argument* is an expression about the range values that will evaluate to TRUE or FALSE. For example, ="Eurozone" would test whether a value in the range is equal to the text "Eurozone."

    If you're going to combine a cell reference with operators in the criteria argument, you will need to put operators in quotation marks, and concatenate that with the reference by using an ampersand (&).

- The *sumrange argument* is the range containing the values for which you want to calculate the sum (for which the criterion is met). In this example, that would be the sales totals.

Exercise: Using SUMIF to conditionally sum values

`My Decisions` is open.

Exam Objective: MOS Excel Core 4.2.2

Do This	How & Why
1. Observe the Invoices worksheet.	It contains a large list of Java Tucana invoices for 2014.
2. Activate the **Sales by Date Ranges** worksheet.	This is set up to provide sales totals based on start and end dates. First, you'll use SUMIF to calculate the sales totals after the start date.
3. In cell B11, type =SUMIF(.	Excel's screen tip tells you the first argument is range. This is the range containing the values you want to test with your criterion, or the dates. End Date 12/31/2014 Total Sales =SUMIF(SUMIF(**range**, criteria, [sum_range])
4. Activate the **Invoices** worksheet, then select A10:A1007.	Select A10, hold **Ctrl**+**Shift**, and press the Down Arrow key. To insert a reference to this range in the function. =SUMIF(Invoices!A10:A1007
5. Type, ">="&B9.	To move to and enter the criteria argument. Note that you need to concatenate the operators, which need quotations marks, with the cell reference by using the ampersand (&). B9 is the cell with the start date.
6. Type a comma; then, on the Invoices worksheet, select H10:H1007.	To enter the sum_range argument, the sales totals. =SUMIF(Invoices!A10:A1007,">="&B9,Invoices!H10:H1007 SUMIF(range, criteria, [**sum_range**]) G H
7. Type a closing parenthesis, then press **Enter**.	The function calculates the total sales for dates after 3/1/14.
8. Enter a different 2014 start date.	In B9. Validation will prevent a date that's not in 2014. The sales total changes as you change start dates.
9. Save the workbook.	

Using SUMIFS

SUMIFS calculates a sum when two or more conditions are true. The tricky part of using SUMIFS is that its syntax is opposite that of SUMIF, with the range to calculate coming first, and the criteria ranges coming at the end, after their associated criteria. Thus, the syntax looks like this.

Exam Objective: MOS Excel Expert 3.1.3

SUMIFS(sum_range,criteria_range1,criteria1,...)

- *sum_range* is the range containing the values on which you want to calculate.
- *criteria_range* is the range containing the values to test.
- *criteria* is the test.

To add more conditions, you add more pairs of critieria_range and criteria arguments. Note that AVERAGEIFS and COUNTIFS have the same syntax and work in the same way.

Exercise: Using SUMIFS to calculate based on a date range

`My Decisions` is open, and the Sales by Date Ranges worksheet is active.

Exam Objective: MOS Excel Expert 3.1.3

Do This	How & Why
1. Observe the End Date.	Now, you'd like the function to calculate for a date range. This means really two criteria: after one date, and before another. For more than one criterion, you'll need the SUMIFS function.
2. In B11, type =SUMIFS(.	To begin to enter the function. Notice that the sum_range argument comes first, unlike SUMIF, in which it comes last.
3. Type or enter Invoices! H10:H1007.	The easiest way is to activate the **Invoices** worksheet, then select the range. You don't need to worry about using a relative reference here because you won't be copying this formula to another location.
4. Type or enter ,Invoices! A10:A1007.	Don't forget the comma! To enter the first criteria_range argument. This is the date column on the Invoices worksheet.
5. Type ,">="&B9.	To enter the first criterion: that the date be at least the start date (in cell B9). So far, the function looks like the one shown below. The screen tip is showing that you can enter more pairs of criteria_range and criteria arguments. =SUMIFS(Invoices!H10:H1007,Invoices!A10:A1007,">="&B9 SUMIFS(sum_range, criteria_range1, **criteria1**, [criteria_range2, criteria2], ...)
6. Type or enter the criteria_range2 and criteria2 arguments, as shown.	This criteria_range argument is the same: the dates on the Invoices worksheet. The criteria argument uses "<=" and refers to the end date cell, B10. >="&B9,Invoices!A10:A1007,"<="&B10

Do This	How & Why
7. Type a closing parenthesis and press **Enter**.	To enter the function.
8. Experiment with different start and end dates in 2014. What happens if your end date is before your start date?	
9. Save and close the workbook.	

Using SUMIFS

Assessment: Decision-making functions

In the IF function, the logical_test argument must evaluate to TRUE or FALSE. True or false?

- **True**
- False

You can use another IF function as the value_if_true argument of an IF function, but not as the value_if_false argument. True or false?

- True
- **False**

Which of the following are differences between SUMIF and SUMIFS? Choose all correct answers.

- SUMIF can take more than one criterion, while SUMIFS takes a single criterion.
- **SUMIFS can take more than one criterion, while SUMIF takes a single criterion.**
- SUMIFS cannot be used with numeric criteria.
- **The sum_range argument comes first in SUMIFS, but last in SUMIF.**

Module B: Lookup and reference functions

Lookup functions find a value in a range, then return a value from another location in that range. Reference functions can give you location information for values in ranges. You can combine these types of functions to perform powerful information gathering.

You will learn how to:

- Use a lookup function to find an exact match for a value and then return another value

- Use a lookup function to find an approximate match in a range, and then return another value

- Use INDEX and MATCH together to look up values based on two variables

Lookup functions

When you look up a value in a table, you're doing exactly what a lookup function does. You find a lookup value in a column, then scan across its row to find an associated value in another column. In Excel, the VLOOKUP function does exactly that. The HLOOKUP function is similar, but it looks across the first row for the lookup value, then scans down that column to find the value you want returned. Lookup functions turn out to be useful in an enormous variety of Excel applications.

Using VLOOKUP

To use the VLOOKUP function, you need to tell it what to look for, where to look, from which column to return a value, and how to look for the lookup value.

Exam Objective: MOS Excel Expert 3.2.1; 3.2.2

1. Create a lookup table.
   To use VLOOKUP, the table must have the lookup values in the first column. If you want to use a range lookup (more on this shortly), you should sort the table in ascending order by the first column. There can be any number of columns, but note the relative position of the column from which you want to return values.

2. Select a cell, and insert the VLOOKUP function.

   VLOOKUP(lookup_value, table_array, col_index_num, [range_lookup])

   - *lookup_value* is a reference to the cell to look for in the first column of the table_array. You usually want to use a relative reference here so that the lookup value changes when you copy the function.

   - *table_array* is the lookup table, including the lookup column and any columns from which you want to return values. It is best to use an absolute reference for table_array, so that the table stays the same when you copy the function. You can also name the range for the table_array argument, and then use the name in the function.

   - *col_index_number* is the relative position of the column from which you want to return values. For example, if the lookup column is column B, and the values you want to return are in column E, then column E is the fourth column in the lookup table, so you would enter 4 for the col_index_number argument.

   - *range_lookup* determines whether you want to look for approximate matches within the first column or exact matches.

3. Enter the function.

HLOOKUP is very similar, except that you are looking up values in a row, and returning them from another row.

Exercise: Using VLOOKUP to return the sales total for a rep

Exam Objective: MOS Excel Expert 3.2.2

Do This	How & Why
1. Open `Lookups`.	From the `Logical and Lookup Functions` data folder. The Customers worksheet is active.
2. Pick a customer name from the drop-down list in cell A4.	The rep name will change in B4.
3. Select B4.	The cell contains a VLOOKUP function that looks for the selected customer, then returns the rep name from the second column. The last argument is FALSE because the type of lookup here is an exact match, not a range lookup. With an exact match, VLOOKUP finds the first entry that exactly matches the lookup value, and returns the value in the same row from the second column.
4. Activate the **Reps** worksheet.	This worksheet contains some sales rep information. You'll enter a lookup function in J10 to find total sales for the rep who is selected in cell I10.
5. Name A10:B19 `RepSales`.	Select the range, click in the Name box, type `RepSales`, and press **Enter**. To create a name for the lookup table you'll use. Using names for lookup tables can make your lookup functions easier to enter and understand.
6. In J10, enter a VLOOKUP function to return the sales total for the rep selected in I10.	
a) Select J10, then type `=vlookup(`.	To begin to enter the function. The first argument is lookup_value.
b) Click I10, then type a comma.	To enter the lookup value (the rep selected in I10). The next argument is table_array.
c) Type `RepSales`, then type a comma.	To indicate that the function should look for the lookup value in the RepSales table. The next argument is col_index_num, the column from which you want to return a value.
d) Type `2`, then type a comma.	You want to return a value from the second column in the table, the sales total column.
e) Type `FALSE`.	To indicate that you want to find an exact match for the lookup value. `=vlookup(I10,RepSales,2,FALSE`
f) Type a closing parenthesis, then press **Enter**.	To enter the function. It returns the correct sales total for Blackwell, $3,046.
Continued...	

Do This	How & Why
7. Select a different sales rep from the I10 drop-down list.	The VLOOKUP function returns the correct sales total for the selected rep.
8. Save the workbook as My Lookups.	

Looking up a sales rep total

```
=VLOOKUP(I10,RepSales,2,FALSE)
```

G	H	I	J
		Rep	Total
		Blackwell	$ 3,046

Range lookups

Sometimes, such as when you're looking for a name, you want to find exact matches for a lookup value. The other kind of lookup is a range lookup, in which you want to find an approximate match for a value within a range. A tax table is a good example of this kind of lookup.

A part of a tax table.

If line 43 (taxable income) is—		And you are—			
At least	But less than	Single	Married filing jointly *	Married filing sepa- rately	Head of a house- hold
			Your tax is—		
3,000					
3,000	3,050	303	303	303	303
3,050	3,100	308	308	308	308
3,100	3,150	313	313	313	313
3,150	3,200	318	318	318	318
3,200	3,250	323	323	323	323

Here, you would look to see where your taxable income falls within a range of values (in the first two columns).

With a lookup function, you put the ranged values in ascending order in a single column, such as in a grade table.

0	F
65	D
73	C
82	B
92	A

The structure of the lookup table is important. If a value is greater than or equal to the value in the first cell, but less than the value in the second cell of the first column, the function will return a value from the first row. So, here, if the value is less than 65, the grade would be "F." A value greater than or equal to the second value in the first column and less than the third would return the value from the second row ("D").

To use a range lookup, set the range_lookup argument value to TRUE.

Exercise: Using a range lookup to determine commissions

`My Lookups` is open, and the Reps worksheet is active.

Do This	How & Why
1. Observe the Grades worksheet.	This worksheet uses VLOOKUP (in column C) to find the student's numeric grade in the Grades table (E2:F6) and then return the correct grade. This is a range lookup because, in most cases, the numeric grades must be found between the values in the Grades table.
2. Activate the **Reps** worksheet, and observe the % Quota column.	This column calculates the percentage of quota a rep has achieved by dividing total sales by quota.
3. Observe L10:M13.	This range contains a lookup table for commission percentage. Up to 100% quota, the rep receives no commission. From 100% to 110%, the commission is 4%, and so on.
4. Define QuotaComm as a name referring to L10:M13.	Select the range, enter `QuotaComm` in the Name box. Again, naming lookup tables makes your functions easier to work with. Do not include the row 9 headings in the range that you name. The lookup range should be only the values.
5. In E10:E19, enter a VLOOKUP function to determine commission percentage.	
a) Select E10:E19.	
b) Type or enter the function as shown.	Note that you can use the value 1 instead of TRUE for any logical argument, such as range_lookup. Similarly, you can use 0 instead of FALSE.
	`=VLOOKUP(D10,QuotaComm,2,1)`
c) Press **Ctrl**+**Enter**.	To enter the function throughout the selected range. The commission rates vary, depending on each rep's quota percentage. Notice that the Comm column values are now filled in as well.
6. Save the workbook.	An example follows the exercise.

Calculating commissions with a range lookup

	A	B		C		D	E	F	
9	**Rep**	**Total Sales**		**Quota**		**% Quota**	**Comm%**	**Comm**	
10	Blackwell	$	3,046	$	4,000	76%	0%	$	-
11	Daniels	$	9,207	$	8,500	108%	4%	$	368
12	Franklin	$	12,539	$	12,000	104%	4%	$	502
13	Hernandez	$	15,600	$	13,000	120%	5%	$	780
14	Lloyd	$	23,392	$	22,000	106%	4%	$	936
15	McCanney	$	20,067	$	22,000	91%	0%	$	-
16	Patterson	$	7,113	$	6,500	109%	4%	$	285
17	Sanchez	$	12,905	$	10,000	129%	6%	$	774
18	Schiller	$	19,240	$	20,000	96%	0%	$	-
19	Westlein	$	16,846	$	18,000	94%	0%	$	-
20	**Total**	$	139,955						

Formula bar: E10 =VLOOKUP(D10,QuotaComm,2,1)

Reference functions

Reference functions provide information about data reference or position, or provide data according to its reference or position. Two very useful functions of this type are MATCH, which gives the relative position of data in an array, and INDEX, which returns a value according to its relative position in an array.

Often, these functions are combined to look up a value based on two values. For example, MATCH finds where a value is in the first column of a table, and another MATCH function finds where a second value is in the first row of the same table. An INDEX function can then look up a value in the table based on those two positions.

Using MATCH

Here is the syntax for the MATCH function:

Exam Objective: MOS Excel Expert 3.2.3

MATCH(lookup_value, lookup_array, [match_type])

1. Select a cell, and begin to enter MATCH.
2. Enter the *lookup_value* argument.
   This is the value that you want to find in a table.
3. Enter the lookup_array argument.
   This is the table in which you want to find the lookup value. Often, it's a single row or column.
4. Enter the match type.
   - 1 will find the largest value less than or equal to the lookup value. To use this type, your values must be sorted in ascending order (or you will get an error).
   - 0 will find the first value that is an exact match. Use 0 when searching for text.
   - -1 will find the smallest value greater than or equal to the lookup value. In this case, the value must be in descending order.
5. Enter the function.

The function returns the relative position in the array of the matched value.

Using INDEX

The INDEX function has the following syntax.

Exam Objective: MOS Excel Expert 3.2.4

INDEX(array, row_num, [column_num])

- *array* is the table (or row or column) in which you want to find a value.
- *row_num* is the relative row position of the value in the table.
- *column_num* is the relative column position of the value in the table.

INDEX returns the value of the cell at the intersection of row_num and column_num within the array. Combined with MATCH functions, INDEX can return values from anywhere in almost any table.

Exercise: Using MATCH and INDEX to return values from a table

`My Lookups` is open.

Note: MOS Excel Expert 3.2.3, 3.2.4

Do This	How & Why
1. Activate the **Index-Match** worksheet.	It contains sales data by rep and product.
2. Observe G10:I11.	This range contains an interface for looking up sales values for a combination of rep and product. In this small a table, such an interface isn't entirely necessary. But you can imagine how useful this could be with a very large amount of data.
3. Could you use VLOOKUP to return the value?	Not by itself. You need to tell VLOOKUP which column number to return from, and you don't know that until a user selects a product.
4. In I11, type =index(.	To being to enter the INDEX function, which will return a value based on its relative position in a table. The first argument is array, which is the table in which you want to look.
5. Select B11:E20, then press **F4**.	To enter an absolute reference to the data in the sales table. So far, the function looks like this: =INDEX($B$11:$E$20
6. Type a comma, then type match(.	You will use a MATCH function as the row_num argument of the INDEX function. This MATCH function will find the relative position of the selected rep in the column rep names.
7. Click G11, then type a comma.	
Continued...	

Do This	How & Why
8. To indicate that the function should look up the value in G11. The next argument is lookup_array.	
9. Select A11:A20, press **F4**, then type a comma.	To enter the argument as an absolute reference, and to move to the next argument, match_type.
10. Type 0, then a closing parenthesis.	To specify that you want to look for an exact (text) match. The function now looks like this:
	=INDEX($B$11:$E$20,MATCH(G11,$A$11:$A$20,0)
11. Type a comma.	Now you need to enter the column_num argument for the INDEX function. You'll use another MATCH function here to return the relative position of the selected product in the row of product headings.
12. Type or enter `match(H11,$B$10:$E$10,0).`	To calculate the relative position of the product in the product column labels. The third argument of the INDEX function should look like this (whew!):
	,MATCH(H11,$B$10:$E$10,0)
13. Type a closing parenthesis, then press **Enter**.	To enter the function.
14. Try selecting other reps and products.	If you've entered the function correctly, it will return the correct value every time. This technique is very powerful and useful.
15. Save and close the workbook.	

Using INDEX and MATCH to look up values

I11 f_x =INDEX($B$11:$E$20,MATCH(G11,$A$11:$A$20,0),MATCH(H11,$B$10:$E$10,0))

	A	B	C	D	E	F	G	H	I
10	Rep	Tucana Roast	Indus Tea	Phoenix Roast	Vela Herbal		Rep	Product	Sales
11	Blackwell	$ 275	$ 680	$ 940	$ 1,150		McCanney	Phoenix Roast	$ 4,646
12	Daniels	$ 1,596	$ 2,914	$ 2,164	$ 2,534				
13	Franklin	$ 3,938	$ 2,624	$ 2,770	$ 3,207				
14	Hernandez	$ 3,325	$ 2,852	$ 3,851	$ 5,572				
15	Lloyd	$ 5,072	$ 6,271	$ 5,891	$ 6,159				
16	McCanney	$ 3,618	$ 4,827	$ 4,646	$ 6,977				
17	Patterson	$ 1,843	$ 1,400	$ 1,586	$ 2,284				
18	Sanchez	$ 4,055	$ 2,262	$ 3,958	$ 2,631				
19	Schiller	$ 4,766	$ 4,756	$ 4,765	$ 4,955				
20	Westlein	$ 4,488	$ 4,164	$ 3,832	$ 4,361				

Assessment: Lookup and reference functions

To use a lookup function, you always have to sort the values in the first column or row of the lookup table. True or false?

- True
- **False**

Which function would you use to return the relative position of data within an array?

- INDEX
- VLOOKUP
- **MATCH**

MATCH can perform range-type lookups on an array sorted in ascending or descending order. True or false?

- **True**
- False

Summary: Logical and lookup functions

You should now know:

- About using functions to make decisions; how to use the IF function to return different values, depending on whether a condition is true; how to nest IF functions, or use AND, OR, or NOT to make more complex decisions; and how to use SUMIF and SUMIFS to calculate sums according to criteria

- About how lookup functions work, how to use VLOOKUP to return a value from a table based on a lookup value, and how to use MATCH and INDEX to find the location of a value in a table and then return that value

Synthesis: Logical and lookup functions

In this synthesis exercise, you'll use and decision-making functions to display values based on conditions, create a more complex condition, use VLOOKUP to find and use values in a table, and combine MATCH with a lookup function to create a more powerful lookup.

1. Open `Logical Lookups Synthesis` from the `Logical and Lookup Functions` data folder. The first worksheet contains an invoices list.

2. In column I, enter a function that will return "High Volume" if the units for the invoices is greater than or equal to 10, or blank text if not. Use "" for blank text.

3. Copy the formula to the entire column.

4. In column J, enter a function that will return "Yes" if BOTH of these conditions are true: the units are greater than or equal to 10 and the region is "Eurozone." If one or neither condition is met, return "No."

5. In L10, use SUMIF to calculate the total sales for orders in which the units are greater than or equal to 10. **Hint**: You'll need to enclose the entire criteria argument in quotation marks.

	F	G	H	I	J	K	L
9	Units	Cost	Total Sale	High Volume?	High & Eurozone?		High Volume Sales Total
10	4	17.10	$ 68.40		No		16913.7
11	14	16.20	$ 226.80	High Volume	No		
12	11	15.30	$ 168.30	High Volume	No		
13	9	16.20	$ 145.80		No		
14	5	16.20	$ 81.00		No		
15	1	17.10	$ 17.10		No		
16	9	16.20	$ 145.80		No		
17	13	16.20	$ 210.60	High Volume	Yes		

6. Save the workbook as `My Logical Lookups Synthesis`.

7. Activate the **Employees** worksheet.

8. In cell I8, enter a function that will build the employee's full name using the employee ID selected in H8. To combine the first and last names, use the CONCATENATE function. Put a single space between the first and last names by using " " as the middle of the argument in the CONCATENATE function.

9. In M8, enter a function that will return the value for the employee ID in K8 and the type of information selected in L8. Use a MATCH function to determine the column number to use in a VLOOKUP function. **Hint**: Depending on how you set up your lookup ranges, there might be one more column in the lookup table for the VLOOKUP function than there is in the lookup array for the MATCH function. What does that mean you will need to do with the col_index_number argument of VLOOKUP?

10. Select different employee IDs and Information types in cells K8 and L8 to test your function.

11. Why are hire dates in M8 formatted as dates, while salaries are formatted as currency?

12. Save and close the workbook.

The completed employee lookups

Emp ID	Full Name		Emp ID	Information	Value
069	Wallace Bronk		039	Salary	$63,000

Chapter 2: Advanced Formulas

You will learn:

- About formula auditing features, and how to find errors in your formulas
- How to control formula options
- How to use array functions to streamline workbooks and save resources

Learning time: 70 minutes

Module A: Auditing and error-trapping

On the Formula tab, Excel offers many auditing features to help you understand and correct errors in the formulas in your worksheet. You can trace precedent and dependent cells, check errors, and evaluate formulas step-by-step. You can then use a decision-making function such as IFERROR to trap and handle errors when they occur.

You will learn how to:

- Trace precedent and dependent cells for a formula

- Find and correct the source of errors in a formula

- Evaluate a formula to understand what it does

- Use IFERROR to display a value if a formula produces an error

Precedents and dependents

Knowing the relationships between the cells, ranges, and formulas on your worksheets is essential to understanding them, and to solving problems.

Formula precedents are the cells or ranges to which a formula refers. Here, the formula in cell F2 refers to the range B2:E2. The precedent arrow shows that relationship clearly.

Dependents, on the other hand, are the formulas that *refer to* a particular cell. Here, the dependent arrows show that cell B2 feeds into the formulas in cells B5 and F2, while F2 then feeds into F5. Thus, B5, F2, and F5 are all *dependent upon* B2.

Tracing precedents and dependents

You can display precedent and dependent arrows to show the relationships among cells, ranges, and formulas. To display these, you use the buttons in the Formula Auditing group on the Formulas tab.

Exam Objective: MOS Excel Expert 3.5.1

```
Trace Precedents    Show Formulas
Trace Dependents    Error Checking
Remove Arrows       Evaluate Formula
          Formula Auditing
```
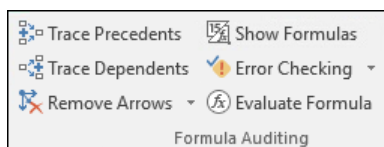
- Click **Trace Precedents** to display arrows showing the direct precedent cells and ranges for the active cell's formula.

- Click **Trace Dependents** to display arrows showing any cells that contain formulas that depend on the active cell.

- Click one of the buttons a second or a third time to show second- and third-level precedents and dependency.

- To remove arrows, click **Remove Arrows**, then click a command in the menu.

Exercise: Showing precedent and dependent relationships in a worksheet

Exam Objective: MOS Excel Expert 3.5.1

Do This	How & Why
1. Open `Auditing`.	From the `Advanced Formulas` data folder. The Reps worksheet shows sales by rep and product. Column G is supposed to calculate percentages of the total sales for each rep, but there are errors.
2. Show formula precedents for G11.	
a) Select G11.	
b) On the Formulas tab, in the Formula Auditing group, click **Trace Precedents**.	The formula in the cell refers to F11 and F21, so precedent arrows point from those cells to G11.

G11		×	✓	f_x	=F11/F21

	E	F	G	H
10	Vela Herbal	Total	% of All	
11	$ 1,150	$ 3,046	2.2%	
12	$ 2,534	$ 9,207	#DIV/0!	
13	$ 3,207	$ 12,539	#DIV/0!	
14	$ 5,572	$ 15,600	#DIV/0!	
15	$ 6,159	$ 23,392	#DIV/0!	
16	$ 6,977	$ 20,067	#DIV/0!	
17	$ 2,284	$ 7,113	#DIV/0!	
18	$ 2,631	$ 12,905	#DIV/0!	
19	$ 4,955	$ 19,240	#DIV/0!	
20	$ 4,361	$ 16,846	#DIV/0!	
21	$ 39,829	$ 139,955		

Continued...

Do This	How & Why
3. Click **Trace Precedents** again.	Be sure that G11 is still active when you click the button. Excel adds the next level of precedent arrows, showing that cells in B11:E11 feed into F11, and the results in column F feed into F21.
4. Click **Remove Arrows**.	The precedent arrows disappear.
5. Select B11, then click **Trace Dependents**.	To show arrows pointing to the cells containing formulas that refer to B11 (B21 and F11).

6. Click **Trace Dependents** again.

To show second-level dependencies. The formulas for grand total (in F21) and % of All for Blackwell ultimately depend on cell B11.

Rep	Tucana Roast	Indus Tea	Phoenix Roast	Vela Herbal	Total	% of All
Blackwell	$ 275	$ 680	$ 940	$ 1,150	$ 3,046	2.2%
Daniels	$ 1,596	$ 2,914	$ 2,164	$ 2,534	$ 9,207	#DIV/0!
Franklin	$ 3,938	$ 2,624	$ 2,770	$ 3,207	$ 12,539	#DIV/0!
Hernandez	$ 3,325	$ 2,852	$ 3,851	$ 5,572	$ 15,600	#DIV/0!
Lloyd	$ 5,072	$ 6,271	$ 5,891	$ 6,159	$ 23,392	#DIV/0!
McCanney	$ 3,618	$ 4,827	$ 4,646	$ 6,977	$ 20,067	#DIV/0!
Patterson	$ 1,843	$ 1,400	$ 1,586	$ 2,284	$ 7,113	#DIV/0!
Sanchez	$ 4,055	$ 2,262	$ 3,958	$ 2,631	$ 12,905	#DIV/0!
Schiller	$ 4,766	$ 4,756	$ 4,765	$ 4,955	$ 19,240	#DIV/0!
Westlein	$ 4,488	$ 4,164	$ 3,832	$ 4,361	$ 16,846	#DIV/0!
Total	$ 32,975	$ 32,749	$ 34,402	$ 39,829	$139,955	

Do This	How & Why
7. Remove the dependent arrows.	Click **Remove Arrows**.

Errors

There are many kinds of errors that can happen in Excel. Sometimes, you see an error value in a cell, such as #VALUE! or #DIV/0!, and those values provide a clue as to what's going on. #DIV/0!, for example, means that the formula is in some way trying to divide by zero, which is mathematically impossible.

But not all error values tell you as clearly what the problem is. And there are also logic errors, those that produce a result—but the *wrong* result. To solve errors that come up in your worksheets, you need to use the tools Excel provides and some good problem-solving skills.

Tracing and correcting errors

If you see an error in a cell, you can trace it and then correct it.

Exam Objective: MOS Excel Expert 3.5.3

1. Select the cell containing the error.
2. If you like, click the Error SmartTag to get information about the error.

3. Click the Error Checking drop-down arrow, then click **Trace Error**.

Excel will add precedent arrows pointing to the cells that are generating the error. It's not perfect or automatic; you will still need to think about the error, but at least you have clues as to where it came from.

4. Correct the error either by changing the original formula, or by changing a precedent cell.

Exercise: Tracing and correcting a formula error

`Auditing` is open.

Exam Objective: MOS Excel Expert 3.5.3

Do This	How & Why
1. Select cell G12.	It displays the error value #DIV/0!. When you select a cell containing an error, an error SmartTag appears.
2. Click the SmartTag.	It tells you that this is a Divide by Zero error and gives you handling options. You'll use a different approach.
3. Press **Esc**.	To close the SmartTag options.
4. Click the Error Checking drop-down arrow, then click **Trace Error**.	It shows that F12 and F22 are precedents for this formula. The error is that the formula divides by zero, and F22 contains nothing, or zero.
Continued...	

Do This	How & Why
5. Why do you think the error happened?	When the formula was copied from G11, the relative reference to F21 was adjusted. You need to lock that reference in place, then copy the formula to all the cells in column G again.
6. Correct the formulas. a) Select G11:G20.	
b) In the formula in G11, change the reference to F21 to be absolute.	Select the reference, and press **F4**. The formula should look like this. `=F11/$F$21`
c) Press **Ctrl**+**Enter**.	To enter the formula throughout the range. Notice that the error-tracing arrows disappear now that you've corrected the errors.
7. Save the workbook as `My Auditing`.	

The corrected percentage formulas

	A	B	C	D	E	F	G
10	Rep	Tucana Roast	Indus Tea	Phoenix Roast	Vela Herbal	Total	% of All
11	Blackwell	$ 275	$ 680	$ 940	$ 1,150	$ 3,046	2.2%
12	Daniels	$ 1,596	$ 2,914	$ 2,164	$ 2,534	$ 9,207	6.6%
13	Franklin	$ 3,938	$ 2,624	$ 2,770	$ 3,207	$ 12,539	9.0%
14	Hernandez	$ 3,325	$ 2,852	$ 3,851	$ 5,572	$ 15,600	11.1%
15	Lloyd	$ 5,072	$ 6,271	$ 5,891	$ 6,159	$ 23,392	16.7%
16	McCanney	$ 3,618	$ 4,827	$ 4,646	$ 6,977	$ 20,067	14.3%
17	Patterson	$ 1,843	$ 1,400	$ 1,586	$ 2,284	$ 7,113	5.1%
18	Sanchez	$ 4,055	$ 2,262	$ 3,958	$ 2,631	$ 12,905	9.2%
19	Schiller	$ 4,766	$ 4,756	$ 4,765	$ 4,955	$ 19,240	13.7%
20	Westlein	$ 4,488	$ 4,164	$ 3,832	$ 4,361	$ 16,846	12.0%
21	Total	$ 32,975	$ 32,749	$ 34,402	$ 39,829	$139,955	

Error-trapping

Error-trapping means what it sounds like: catching an error and doing something else with it. For example, if you know that a function will return an error if an input cell has text in it, you can wrap that function in another function so that if an error occurs, users will see something other than an error value. In this small worksheet, the total is calculated from the sale amount and the discount.

Customer	Sale	Discount	Total
Accounts Now	$ 85.50	10%	$ 76.95
Blastera	$ 275.40	None	#VALUE!
BlazerFire	$ 16.20	20%	$ 12.96

The total for Blastera produces an error because it's trying to calculate on the text "None." But you could "trap" this error, so that when the function results in an error, it simply displays the sale amount.

Customer	Sale	Discount	Total
Accounts Now	$ 85.50	10%	$ 76.95
Blastera	$ 275.40	None	$ 275.40
BlazerFire	$ 16.20	20%	$ 12.96

Evaluating formulas

With complicated formulas and functions, it's sometimes hard to see exactly where an error might be occurring. By using the Evaluate Formula feature, you can step through a calculation to determine exactly when it's going wrong.

Exam Objective: MOS Excel Expert 3.5.4

1. Select the cell containing the formula you want to evaluate.
2. On the Formulas tab, in the Formula Auditing group, click **Evaluate Formula**.
   To display the formula in the Evaluate Formula window.
3. Click **Evaluate** to go through the formula piece by piece, to see where the error might lie.
   When you first see an error, it is usually a clear indicator of exactly which part of the formula is producing the error.
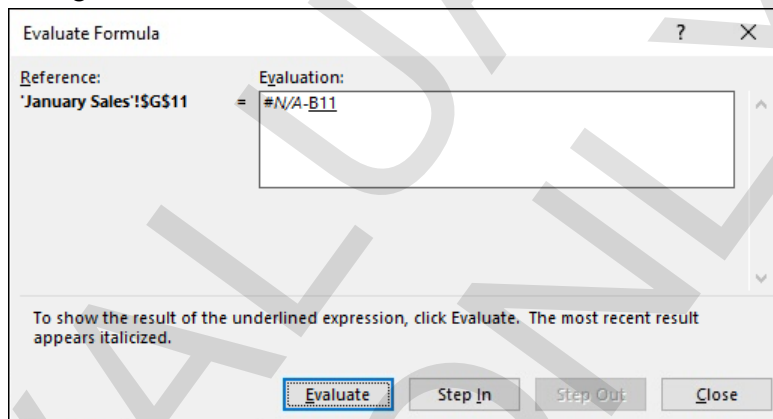
4. Use **Step In** and **Step Out** to evaluate precedent cells' formulas and to return to the main formula.
5. Click **Close** when you're finished.

Using IFERROR

The IFERROR function is one of the best tools you have for trapping errors. Its syntax looks like this.

IFERROR(value, value_if_error)

- Use the formula you're concerned about as the *value argument*.
   If the formula in the value argument does not produce an error, IFERROR returns the result of the value argument.
- For *value_if_error*, use whatever you want to appear when the formula in question results in an error.

Exercise: Evaluating a function and using IFERROR

`My Auditing` is open.

Exam Objective: MOS Excel Expert 3.5.4

Do This	How & Why
1. Activate the **January Sales** worksheet.	This contains a report of January sales by customer.
2. Select and observe cell E11.	This VLOOKUP function returns the customer's discount percentage from a table on the Customer Info sheet.
3. Select and observe F11 and G11.	These formulas calculate what the sales amount would have been before the discount and how much the discount was, respectively.
4. In cell D11, select **Award Sportswear**.	Now, the formulas in F11 and G11 are showing errors.
5. Select F11, then click **Evaluate Formula**.	On the Formulas tab, in the Formula Auditing group. The Evaluate Formula window appears.
6. Click **Evaluate**.	The VLOOKUP function here is looking for the value "Award Sportswear" in the January Sales table.
7. Click **Evaluate** again.	Now, you can see the range "JanuarySales" refers to.
	Evaluation:
	`= (VLOOKUP("Award Sportswear",$A$11:$B$51,2,0)/(1-E11))`
8. Click **Evaluate** again.	To see the error happen. The #N/A! error is happening because although Award Sportwear is a customer, they aren't in the January sales report. The lookup value "Award Sportwear" isn't in the lookup table; it's "not available," or "N/A."
9. Click **Close**.	To close the Evaluate Formula window.
10. In F11, use IFERROR to display "No Sales" when VLOOKUP results in an error.	
a) Select F11.	
b) On the formula bar, click just after the equal sign, and type as shown.	Don't forget the open parenthesis after "IFERROR."
	`=IFERROR(VLOOKUP(D11,JanuarySales,2,0)/(1-E11)`
c) Click at the end of the formula, type a comma, then type **"No Sales")**.	The formula should look like this.
	`=IFERROR(VLOOKUP(D11,JanuarySales,2,0)/(1-E11),"No Sales")`

Do This	How & Why
d) Enter the formula.	The cell should display "No Sales." But the formula in G11 is still producing an error.
11. Use IFERROR to trap the error in cell G11.	The formula will look like this. =IFERROR(F11-B11,"No Sales")
12. Test the formulas by selecting a few different customers.	
13. Save and close the workbook.	

The worksheet with trapped errors

Customer	Sales		Customer	Discount %	Pre-Disc	Disc $
Accounts Now	$ 85.50		Brocadero	10%	No Sales	No Sales
Blastera	$ 275.40					
BlazerFire	$ 16.20					

Assessment: Auditing and error-trapping

You can use arrows to trace only one level of precedence and dependence. True or false?

- True
- **False**

Which of the following are ways to find and fix errors? Choose all that apply.

- **Use the Trace Error command.**
- **Evaluate a formula that produces an error.**
- Double-click a cell containing an error.
- **Trace precedents for the cell.**
- Trace dependents for the cell.

You can use IFERROR to trap logic errors on your worksheets. True or false?

- True
- **False**

Module B: Formula options

You can control many aspects of how formulas behave and appear in Excel. You can control how and when formulas calculate, for example, and you can choose to display formulas rather than their values.

You will learn:

- About calculation and iteration, and how to control both
- How to display formulas

How Excel handles formulas

Most of how Excel handles formulas is invisible. You enter a formula, you see the result. What happens inside the program, you don't normally think about much.

But Excel handles various aspects of formulas in particular ways, and you might want to change those defaults. For example, if a workbook contains a large number of formulas, it might run slowly if it recalculates continuously. Or, you might want to change the degree of Excel's precision.

Changing when a workbook recalculates

By default, Excel recalculates formulas whenever you make a change to a value to which those formulas refer. If a workbook contains many formulas, or input values are changed often, this can make a workbook run slowly. But you can control how and when calculation happens.

Exam Objective: MOS Excel Expert 1.2.3

The Calculation group on the Formula tab
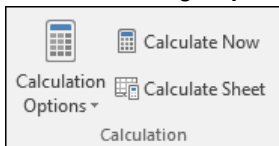
1. On the Formulas tab, in the Calculation group, click **Calculation Options**.
2. Choose the option you want.
   - **Automatic** is the default, and means that Excel will recalculate formulas whenever input values change.
   - **Automatic except for Data Table** is the same, except for a feature called Data Tables, which will recalculate only when you manually recalculate them.
   - **Manual** means that formulas will recalculate only when you tell them to do so.
3. If you want to force calculation, click one of the Calculate buttons.
   - **Calculate Now** calculates all formulas in the workbook.
   - **Calculate Sheet** calculates formulas only for the active worksheet.

Controlling iteration

Iteration means how many times Excel will try to calculate a formula. This can come into play with circular references.

A *circular reference* is a formula that in some way refers to the cell that contains it. In general, you should avoid circular references, but there are situations in which they are necessary and not an error. For example, in the United States, a person's state tax can be deductible on their federal tax return, and vice versa. If you set this up in a workbook, it would contain circular references.

If you do want a circular reference, but don't want Excel to go on calculating it forever, you can limit the number of times it tries to calculate.

1. In Backstage view, click **Options**.

2. Display the Formulas category.

3. Under Calculation Options, click **Enable iterative calculation**.

4. Set the iteration options you want.

   • **Maximum Iterations** determines how many times Excel will try to calculate. Generally, a small number such as 3 or 4 will suffice.

   • **Maximum Change** will cause iteration to stop when the change in value from the last pass is less than the value you specify.

5. Click **OK**.

Displaying formulas

You might sometimes find it useful to display the actual formulas on a worksheet. To do so, on the Formulas tab, click **Show Formulas**. Click the button again to return to displayed values. You can also press **Ctrl**+` (grave accent) as a shortcut to toggle the display of formulas.
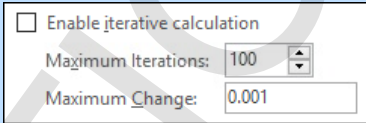
Exam Objective: MOS Excel Core 1.4.8

Exercise: Experimenting with formula options

Exam Objective: MOS Excel Core 1.2.3 and Expert 1.4.8

Do This	How & Why
1. Open `Formula Options`.	From the `Advanced Formulas` data folder. It contains a sales report.
2. On the Formulas tab, in the Calculation group, click **Calculation Options**.	To observe Excel's calculation options. Automatic is the default, meaning that Excel will calculate formulas whenever you change their precedent values. You also can choose manual calculation if you have a large workbook with many calculations that are slowing down performance. ✓ Automatic Automatic <u>E</u>xcept for Data Tables <u>M</u>anual
Continued...	

Do This	How & Why
3. Observe the **Calculate Now** and **Calculate Sheet** buttons.	In the Calculation group. If you use manual calculation, you can use these buttons to calculate either the entire workbook or just the active worksheet.
4. In Backstage view, click **Options**, then click **Formulas**.	
5. Observe the iterative calculation options.	If you need to use circular references in a workbook, you should enable iterative calculation so that Excel does not endlessly try to evaluate a result.
6. Click **Cancel**.	
7. In the Formula Auditing group, click **Show Formulas**.	To toggle between showing values and showing formulas on the worksheet. If you want a quick view of the formulas on your entire worksheet, this can be a very useful option. Notice that when you show formulas, Excel widens all the columns.
8. Press **Ctrl+`**.	The grave accent key is on the upper left of your keyboard. This is a shortcut for toggling the display of formulas. To go back to showing values on the worksheet.
9. Close the workbook.	You don't need to save any changes.

In step 5, the How & Why cell includes:

Enable iterative calculation

Maximum Iterations: 100

Maximum Change: 0.001

In step 7, the How & Why cell includes the table:

Total	% of All
=SUM(B11:E11)	=F11/$F$21
=SUM(B12:E12)	=F12/$F$21
=SUM(B13:E13)	=F13/$F$21
=SUM(B14:E14)	=F14/$F$21

Assessment: Formula options

You can recalculate only a single formula in Excel. True or false?

- True
- **False**

By default, Excel tries to calculate a circular reference indefinitely. True or false?

- **True**
- False

Module C: Arrays
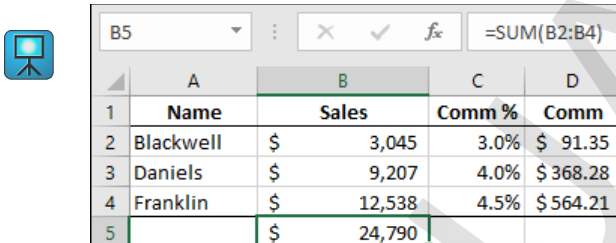
An array is a collection of data in Excel. A range is one kind of array. You can use arrays in formulas and as arguments of functions, and you can create special formulas and functions, called *array formulas* and *array functions*, that enable you to perform many calculations at once.

You will learn:

- About arrays, and the various ways in which Excel uses them

- How to enter an array formula to perform more than one calculation with a single formula

- How to enter an array function

About arrays

While working with functions, you've probably seen the term "array" used to describe the kind of data a function expects as an argument. Broadly speaking, an array is just a collection of things. But in Excel, an *array* is a collection of data organized in a single row, a single column, or a grid of rows and columns. Consider this worksheet:
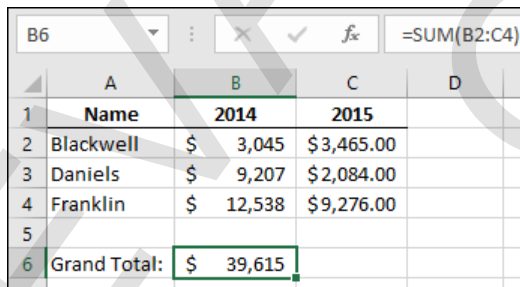
In the SUM function, the argument B2:B4 is a range. But the data it refers to is an array of values that, in Excel, looks like this.

{3045;9207;12538}

The brackets show that this is an array, and the values in the column are separated by semicolons. In this next example, the range is three rows by two columns.

So, the array that the range B2:C4 refers to looks like this:

{3045,3465;9207,2084;12538,9276}

Here, each value in a row is separated by a comma (3045,3465), while rows of data are separated by semicolons. This is the general syntax for an array in Excel. You won't often work directly with arrays, but it's very useful to understand what is going on with them. And you can, if you'd like, enter an array directly in this format in any function that is expecting an array argument.

Array formulas

An *array formula* uses arrays to perform multiple calculations on multiple values in an array. That sounds confusing, but is easier to understand when you see one.

Here, the array formula is the same in all the cells in the range D2:D4. It performs its calculations by multiplying the value in this row (2) in the array B2:B4 by the value in this row (2) in the array C2:C4. Here is the same worksheet, showing the same array formula in cell D3.

When you use an array formula in this way, you have to enter it in the entire range at once, and then you can't edit the individual cells (you can edit only the entire array formula).

Using array formulas has some great advantages.

- Consistent formulas: The formulas in a totals column, for example, are all exactly the same. There is less likelihood of mistakes than when entering many formulas.

- Protection: You cannot easily change an array formula.

- File size: When you use an array formula, Excel saves only one formula, rather than many. Thus, files can be smaller.

Array formulas aren't perfect. Complicated ones can evaluate slowly. Also, if you're sharing your workbooks with other users, they might find array formulas hard to understand.

Entering array formulas

1. Select the entire range in which you want to enter the array formula.

2. Type the formula you want, using arrays as the input values.

   If there are not enough values in an input array compared to the range in which you enter the array formula, you will get an error in some of the cells.

3. Press **Ctrl**+**Shift**+**Enter**.

   Excel will enter the array formula in the entire range.

Editing and deleting array formulas

To change an array formula that has been entered in a range, you have to select the entire range in which it was entered. You can then make changes to it.

- To delete an array formula, select the entire range in which it was entered, then press **Delete**.
- To edit an array formula, select the entire range in which it was entered, edit the formula, then press **Ctrl**+**Shift**+**Enter** to enter the edited formulas as an array formula again.

Exercise: Using an array formula to calculate invoice totals

Do This	How & Why
1. Open `Arrays`.	From the Advanced Formulas data folder. This worksheet contains nearly 1000 rows of invoice data. You could use a regular formula to calculate the Total Sale value for one row, then copy it throughout the range; but instead, you'll use a single array formula.
2. Select H10:H1007.	To enter an array formula in a range, you must first select the range.
3. Type an equal sign, then select F10:F1007.	Click F10, hold down **Shift**+**Ctrl**, and press Down Arrow. You're beginning to build the array formula. Rather than multiplying a single cell by a single cell, this formula will multiply an array (a column in this case) by another array (another column).
4. Type *, then select G10:G1007.	Don't press Enter yet. The formula should look like this. `=F10:F1007*G10:G1007`
5. Press **Ctrl**+**Shift**+**Enter**.	To enter the array formula. Note the brackets enclosing the formula, which indicate that it's an array formula. `{=F10:F1007*G10:G1007}`
6. Examine several cells in column H.	The same array function returns different results for every cell.
7. Attempt to delete H11. a) Select H11, the press **Delete**. b) Click **OK**.	Excel displays a message stating that you cannot change part of an array.
8. Select H10:H1007, then press **Delete**.	To delete the array function. To delete or edit an array function, you must select the entire range that contains it.
9. Click .	To undo the deletion of the array function.
10. Save the workbook as `My Arrays`.	

Array formulas calculating total sales

Array functions

You can use functions in array formulas to create *array functions*. This is especially useful to perform more than one calculation in a single cell.
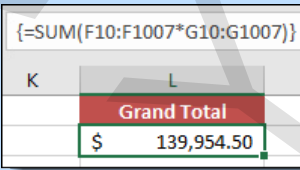
Here, the array SUM function adds up all the commission values by multiplying each Sales amount by each Comm % amount, and then adding them. You enter array functions as you would any other array formula: type the function, and then press **Ctrl**+**Shift**+**Enter**.

Exercise: Using an array function to calculate grand totals

`My Arrays` is open.

Do This	How & Why
1. Select L10.	You'll enter an array formula to calculate the grand total with a single formula.
2. Type **=SUM(**.	You begin an array function just as you would a normal function.
3. Select F10:F1007, then type *****.	
4. Select G10:G1007.	The array function will multiply the values in column F by those in column G, then add them all up. =SUM(F10:F1007*G10:G1007)
5. Press **Ctrl**+**Shift**+**Enter**.	To enter the array function. Excel inserts the brackets enclosing it. The function performs many calculations in a single step. {=SUM(F10:F1007*G10:G1007)}
	K **L** **Grand Total** $ 139,954.50
6. Save the workbook.	

Using TRANSPOSE

You can use the TRANSPOSE array function to transpose the rows and columns of one range to another. The advantage of using the TRANSPOSE function over using paste options to transpose is that the transposition is dynamic; if you change the source range, the transposed range also changes.
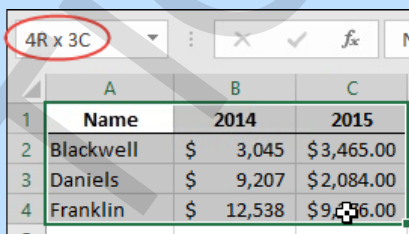
A range transposed with TRANSPOSE

E1		⨯ ✓	fx	{=TRANSPOSE(A1:C4)}			
	A	B	C	D	E	F	G	H
1		2014	2015			Sue	Mike	Jim
2	Sue	200	300		2014	200	400	600
3	Mike	400	500		2015	300	500	700
4	Jim	600	700					
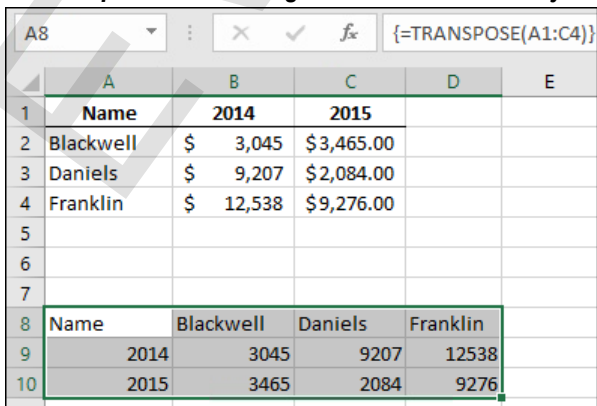
1. Select the range that will contain the transposed range.

   The transposed range must have the opposite dimensions of the source range. That is, if you start with 3 columns and 4 rows as in the preceding example, the range you select must have 4 columns and 3 rows.

2. Enter the TRANSPOSE function.

   a) Type **=TRANSPOSE (**, then select the source range.

   b) Type the closing parenthesis, then press **Ctrl**+**Shift**+**Enter**.

Exercise: Using TRANSPOSE to exchange rows and columns

My Arrays is open.

Do This	How & Why
1. Activate the Transposing worksheet.	It contains very simple sales data for three reps (in rows), over two years (in columns). What if you want the years in rows? You could use paste options to transpose the data, but that data wouldn't change when this did. You'll use the TRANSPOSE array function to create linked, transposed data.
2. Watch the Name box as you select A1:C4.	While you select a range, Excel shows you its dimensions in the name box. This range is 4 rows by 3 columns (4R x 3C).
3. Select A8:D10.	Because the range you want to transpose is 4 rows by 3 columns, the transposed range needs to be 3 rows by 4 columns.
4. Enter the TRANSPOSE array function. a) Type =transpose(.	
b) Select A1:C4.	The range that you want to transpose.
c) Type a closing parenthesis, then press **Ctrl**+**Shift**+**Enter**.	The data is transposed into the new range. TRANSPOSE is an array function, so to change it you would have to select the entire range that contains it.
5. Save and close the workbook.	

The transposed data using the TRANSPOSE array function

Assessment: Arrays

Array formulas must be entered in more than one cell. True or false?

- True
- **False**

Which of the following is the keyboard method for entering an array formula? Choose the one correct answer.

- Ctrl+Enter
- Shift+Enter
- **Ctrl+Shift+Enter**
- Ctrl+Alt+Enter

Summary: Advanced formulas

You should now know:

- About formula auditing features, including how to trace precedents and dependents, how to find errors in your formulas by tracing them and evaluating formulas, and how to trap errors by using IFERROR

- How to control formula options such as calculation and iteration, and how to display formulas

- About arrays, how to use array formulas and functions to perform multiple calculations on multiple ranges using a single formula, and how to exchange rows and columns by using TRANSPOSE

Synthesis: Advanced formulas

In this synthesis exercise, you'll trace precedents and dependents in a café budget and use IFERROR to trap an error. Then, you'll use an array formula to calculate yearly increase, and an array function to calculate profit or loss.

1. Open `Advanced Formulas Synthesis` from the `Advanced Formulas` data folder. This workbook contains a three-year budget for a café.

2. Trace all levels of precedents for cell E16.

3. Remove arrows, then trace all levels of dependents for cell B8.

4. Remove arrows, then find the source of the error in cell E10.

5. Use IFERROR to return blank text ("") if the average functions in E8:E16 return an error.

6. Save the workbook as `My Advanced Formulas Synthesis`.

7. In F8:F16, enter an array formula to calculate average yearly increase for each expense and the total expenses. Hint: subtract the Year1 array from the Year3 array, and divide by 2.

8. In cell B24, enter an array function to calculate the sum of profit or loss for all the years.

9. Save and close the workbook.

The completed budget with array formulas

	A	B	C	D	E	F
7	**Expense**	**Year1**	**Year2**	**Year3**	**Average**	**Yearly Increase**
8	Rent	$ 24,000	$ 24,000	$ 24,000	$ 24,000	$ -
9	Remodeling	$ 12,000	$ 2,000	$ 2,000	$ 5,333	$ (5,000)
10	Legal				$	$ -
11	Equipment	$ 9,000	$ 1,000	$ 1,000	$ 3,667	$ (4,000)
12	Supplies	$ 12,000	$ 13,200	$ 14,500	$ 13,233	$ 1,250
13	Advertising	$ 4,000	$ 1,000	$ 1,200	$ 2,067	$ (1,400)
14	Payroll	$ 60,000	$ 65,000	$ 80,000	$ 68,333	$ 10,000
15	Miscellaneous	$ 10,000	$ 11,000	$ 12,000	$ 11,000	$ 1,000
16	Totals:	$131,000	$117,200	$134,700	$ 127,633	$ 1,850
17						
18						
19		**Year1**	**Year2**	**Year3**		
20	**Revenue**	$ 95,000	$115,000	$145,000		
21						
22						
23						
24	**Total Profit/Loss**	-27900				

Chapter 3: Special functions

You will learn:

- How to use date and time functions
- How to manipulate text with functions
- About some of Excel's statistical and financial functions

Learning time: 75 minutes

Module A: Date and time functions

Excel stores dates as serial numbers, which allows you to perform all sorts of calculations. But understanding how they work and knowing what functions are available to get the date information you want are essential if you want to handle dates well in your workbooks.

You will learn:

- How to manipulate dates to give you information about the current date, year, month, day, or day of the week

- How to calculate days or workdays between two dates, and how to calculate an end date given a start date and a number of workdays

Dates

Excel stores dates as serial numbers, so that it can use them to perform calculations. From Excel's point of view, time began on January 1, 1900, or serial number 1. January 2, 1900, is 2, and so on.

Because of this, you can subtract any two dates to get the number of days between them. But what if you want to know quantities of years, months, or weeks? How would you determine the weekday or month if you know a date? What about calculating workdays, as opposed to calendar days?

Fortunately, Excel has functions that can help you handle all of these situations.

Getting information from a date

You can format dates to show various combinations of day, month, and year in various ways. But it's helpful to know that when you do so, the underlying date is still the serial number for that whole date. If you want to work with day numbers, weekday or month names, or the year as a number, you'll need to use various functions to return those values.

Exam Objective: MOS Excel Expert 3.3.2

- YEAR returns the year, a number from 1900 to 9999.
  The syntax for year is shown here.

    YEAR(serial_number)

- MONTH returns the number of the month (1–12).

    MONTH(serial_number)

- WEEKDAY returns the number of the day of the week.

    WEEKDAY(serial_number,[return_type])

  - The return_type argument allows you to choose the numbering system. In some countries, the standard is to set Sunday as 1; but this approach is not universally accepted, making this argument very useful.

- TEXT converts a number value to text in a format you specify. This is very useful with dates when you want to cover to the text name for a month or a year.

    TEXT(value, format_text)

  - The format_text argument must be enclosed in quotations marks and should correspond to the format code you would use for what you want to see on the Number tab of the **Format Cells** window. In fact, it's often easiest to copy a format code there before trying to construct the TEXT function you want.

Using TODAY and NOW

It's often helpful to have the current date in a cell on a worksheet. You can get the current date by using TODAY, or the current date and time by using NOW.

Exam Objective: MOS Excel Expert 3.3.1

- TODAY returns the current date using the current date format for the cell.
  The function takes no arguments, so its syntax is very simple.
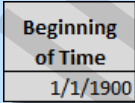
  TODAY()
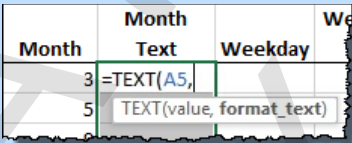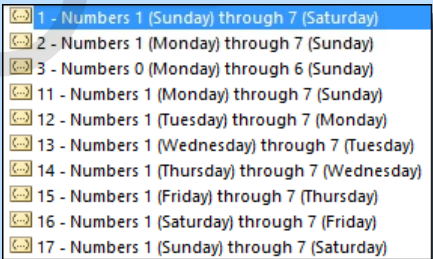
- NOW returns both the date and time. For most applications, you won't need the time, so TODAY will be a better choice. Like TODAY, NOW has no arguments.

Exercise: Working with and converting dates

Exam Objective: MOS Excel Expert 3.3.1, 3.3.2

Do This	How & Why
1. Open Dates.	From the Special Functions data folder. The Dates worksheet has a column of dates, and then several columns in which you will convert those dates using date functions.
2. In A2, enter 1.	The cell is formatted as a date and shows that 1/1/1900 is the first day in Excel. All of the math you can do with dates depends on their being stored as serial numbers. **Beginning of Time** 1/1/1900
3. In A9, enter =TODAY().	The TODAY function shows today's date. This can be very useful for calculations, or for tests for conditional formatting (highlighting a cell, for example, if a due date has passed).
4. In B5:B9, display the year for the dates in column A. a) Select B5:B9. b) Type or enter =YEAR(A5). c) Press **Ctrl**+**Enter**.	 To display the year for the dates in the range.
5. Save the workbook as My Dates. Continued....	

Do This	How & Why
6. In C5:C9, display the number of the month for the dates in Column A. a) Select C5:C9. b) Type or enter =MONTH(A5). c) Press **Ctrl**+**Enter**.	 To display the number of the month for each date. But what if you want the name of the month?
7. Display the full month names in column D. a) Select D5:D9. b) Type =TEXT(A5,.	 To begin to enter the text function, which converts numbers to text. Next, you have to specify the format.
c) Type "mmmm".	To specify that you want to see the full month name; "mmm" would display the short month name, such as "Jun" or "Mar."
d) Type), then press **Ctrl**+**Enter**.	To complete the function and enter it throughout the range. The month names appear in the column.
8. Use WEEKDAY to show the number of the weekday in column E. a) Select E5:E9. b) Type =WEEKDAY(A5,.	 Excel displays a list of return_type argument values that you can use to specify different international systems.
c) Type 1), then press **Ctrl**+**Enter**.	The numbers for the day of the week appear in the column.
9. Use TEXT to display weekday names in column F. a) Select F5:F9.	

Do This	How & Why
b) Enter =**TEXT(A5,"ddd")** throughout the range.	Type the function, then press **Ctrl**+**Enter**. The column shows abbreviations for the weekday days, because that's what the "ddd" format specifies.
c) Change the format to "dddd".	Now, the full weekday names appear.
10. Why do you see different dates in row 9 than those shown in the following figure?	
11. Save the workbook.	

The completes My Dates worksheet.

	A	B	C	D	E	F
4	Date	Year	Month	Month Text	Weekday	Weekday Text
5	3/1/2015	2015	3	March	1	Sunday
6	5/16/2015	2015	5	May	7	Saturday
7	9/30/2015	2015	9	September	4	Wednesday
8	1/2/2016	2016	1	January	7	Saturday
9	1/5/2016	2016	1	January	3	Tuesday

Date calculations

Because dates are stored as serial numbers, you can add and subtract with them, which is useful. But Excel also provides functions for performing more subtle calculations, such as workdays between two dates, the end of a month, or an end date, given a start date and a number of days or workdays.

Using date calculation functions

Excel has many functions you can use to perform calculations on dates. Some of these provide international versions, so that you can, for example, specify different definitions of what a "weekend" is.

Exam Objective: MOS Excel Expert 3.3.2

- EOMONTH returns the serial number of the last day of a month.

  EOMONTH(start_date, months)

  - *start_date* is a date.
  - *months* is the number of months before or after start_date for which you want to return the last day of the month.

- EDATE returns the serial number for a date a certain number of months in the future from a start date.

  EDATE(start_date, months)

- WORKDAY and WORKDAY.INTL return the serial number of a date that is a certain number of workdays in the future.

  WORKDAY(start_date, days, [holidays])

  WORKDAY.INTL(start_date, days, [weekend], [holidays])

  - *holidays* is a reference to a range containing a list of holidays to exclude when calculating.
  - In WORKDAYS.INTL, *weekend* enables you to choose from a list of definitions of "weekend" (Saturday and Sunday, Sunday only, and so on).

- NETWORKDAYS and NETWORKDAYS.INTL calculate the number of workdays between two dates, excluding holidays. NETWORKDAYS.INTL also has the optional weekend argument.

  NETWORKDAYS(start_date, end_date, [holidays])

  NETWORKDAYS.INTL(start_date, end_date, [weekend], [holidays])

Exercise: Performing calculations using date functions

`My Dates` is open.

Exam Objective: MOS Excel Core 3.3.2

Do This	How & Why
1. In column B of the Calculations worksheet, calculate the end of the month for column A.	
a) On the Calculations worksheet, select B5:B9.	
b) Type =EOMONTH(A5,0)	The second argument specifies how many months in the future or past for which you want the end of month date. In this case, 0 gives the current month.
c) Press **Ctrl+Enter**.	To calculate the last day of each month. Note that cell B6 reflects that Excel knows that 2016 is a leap year.
2. In column C, calculate a date that is the number of months later than column A for the number shown in J5.	
a) Select C5:C9.	
b) Type or enter =EDATE(A5,$J$5).	Note the absolute reference to J5. You do not want that reference to change, as the formula is entered throughout column C.
c) Press **Ctrl+Enter**.	To see dates six months in the future.

Do This	How & Why
3. Edit the formulas in column C to calculate dates the number of years shown in K5 in the future.	Change the month argument to multiply the years in K5 by 12. The formula should look like this. `=EDATE(A5,12*$K$5)`
4. In column E, calculate completion dates based the start date, days to add, US weekends, and holidays on the Holidays worksheet.	
a) Go to the range named Holidays2016.	It is on the Holidays worksheet, and contains a list of 2016 holidays. There are only two shown, which isn't a lot of time off! But it will demonstrate the principle.
b) On Calculations, select E5:E9.	
c) Type or enter `=WORKDAY.INTL(A5,D5,.`	To specify that you want to calculate an end date when the start date is in A5 and the number of workdays is in D5. The next argument, weekend, gives you choices of different definitions of a weekend.
d) Type `1,Holidays2016`.	As you start to type the range name for the holidays argument, you can use AutoComplete to finish it. The 1 specifies the US definition of the weekend, Saturday and Sunday.
e) Type a closing parentheses, then press **Ctrl**+**Enter**.	To calculate the completion dates. Notice that although the days to add in row 8 is five, which is a week of workdays in the US, the completion date is eight calendar days after the start date. This is because there is one holiday (Labor Day, 9/5/16) in that time frame.
5. In column G, calculate the workdays between the start dates and end dates.	
a) Select G5:G9.	
b) Type or enter `=NETWORKDAYS(A5,F5,Holidays2016).`	
c) Press **Ctrl**+**Enter**.	To calculate the workdays between the two dates, exclusive of the holidays. Notice that row 6 has one more day than row 5 because of the leap year. If you wanted to specify a different weekend definition, you would use NETWORKDAYS.INTL.
6. Save the workbook.	

The workbook with calculated dates

Start	End of Month	Months Later	Days to Add	Completion Date	End Date	Workdays Between
2/16/2015	2/28/2015	2/16/2017	20	3/16/2015	3/16/2015	21
2/16/2016	2/29/2016	2/16/2018	20	3/15/2016	3/16/2016	22
12/30/2015	12/31/2015	12/30/2017	10	1/14/2016	1/6/2016	5
9/1/2016	9/30/2016	9/1/2018	5	9/9/2016	9/30/2016	21
1/2/2016	1/31/2016	1/2/2018	50	3/11/2016	3/1/2016	42

Time

In Excel, time is the decimal part of a date serial number. So, a decimal between 0 and 1 represents a time in the 24 hours of a day.

Time Decimal	Time
0.041667	1:00:00 AM
0.333333	8:00:00 AM
0.5	12:00:00 PM
0.75	6:00:00 PM
1	12:00:00 AM

Here, you can see how the decimals relate to the formatted times. A third of a day (0.333333) is eight hours, or 8:00:00 AM. Three-quarters of the way through the day (0.75) is 6:00:00 PM.

You can perform calculations by subtracting an end time from a start time, and then multiplying by 24, the number of hours in a day.

Exercise: Performing simple time calculations

`My Dates` is open.

Do This	How & Why
1. Activate the **Time** worksheet.	This is a simple time card, documenting time in and time out, and then calculating hours worked.
2. Select A4:A6, then press **Ctrl+Shift+~**.	The keyboard shortcut for General format. You can see that the times in column A are stored as decimals between 0 and 1.
3. Click ↶ .	To undo the format change.
4. In B4, enter `5:00 PM`.	Excel recognizes a time when you enter it in a standard format like this.
5. Enter the other times as shown.	
		A	B
	|---|---
		Time In	Time Out
	4	8:00 AM	5:00 PM
	5	10:00 AM	12:00 PM
	6	2:00 PM	10:00 PM
6. Calculate the hours worked in column C.	

Do This	How & Why
a) Select C4:C6.	
b) Type = (B4-A4)*24.	To calculate the difference, you subtract the time in from the time out. Then you need to multiply by 24 to get the hours.
c) Press **Ctrl+Enter**.	To enter the formula throughout the range. The hours-worked figures appear. Note that calculating time can be trickier than this when start and end times are on different days.
7. Save and close the workbook.	

The completed time calculations

| C4 | ▼ | ⋮ | × | ✓ | *fx* | =(B4-A4)*24 |

	A	B	C	D
	Time	Time	Hours	
3	In	Out	Worked	
4	8:00 AM	5:00 PM	9	
5	10:00 AM	12:00 PM	2	
6	2:00 PM	10:00 PM	8	

Assessment: Date and time functions

In Excel, dates begin with January 1, 1900. True or false?

- **True**
- False

Which of the following functions would you use to return the name of a month for a date? Choose the one correct answer.

- DATE
- MONTH
- **TEXT**
- EOMONTH

Which function would you use if you want the number of workdays between two dates in a location where the work week has 6 days? Choose the one correct answer.

- NETWORKDAYS
- **NETWORKDAYS.INTL**
- WORKDAY
- WORKDAY.INTL

Times are stored as numbers between 0 and 24. True or false?

- True
- **False**

Module B: Text functions

Text functions allow you combine, trim, and find information within text. Excel has many text functions, and becoming familiar with a few of them is important, particularly if you will be working with data from other sources.

You will learn how to:

- Combine, trim, and alter text values using text functions
- Get information out of a text value by using text functions

Manipulating text

One of the most basic things you'll do with text is combining two text values, or *strings*. The CONCATENATE function does that. And if you work with text that comes from outside of Excel, you might need to use TRIM to remove extraneous spaces within strings.

Using CONCATENATE and TRIM

CONCATENATE combines two or more text values, such as when you want to combine a first name, a space, and a last name. TRIM removes extra spaces from a string.

Exam Objective: MOS Excel Core 4.3.3

- Here is the syntax for CONCATENATE. The output will be a single text string with all the arguments put together.

  CONCATENATE(text1, [text2], ...)

- TRIM removes extra spaces from the text argument, meaning all those except single spaces between words.

  TRIM(text)

Using UPPER, LOWER, and PROPER

You can convert text to all upper-case letters by using the UPPER function, or to all lower-case by using LOWER. Use PROPER to capitalize the first letter of each word in text data.

Exam Objective: MOS Excel Core 4.3.2

- Here is the syntax for UPPER. The output will be the same text string in all upper-case.

  UPPER(text)

- LOWER converts the text to the same string, but in all lower-case letters.

  LOWER(text)
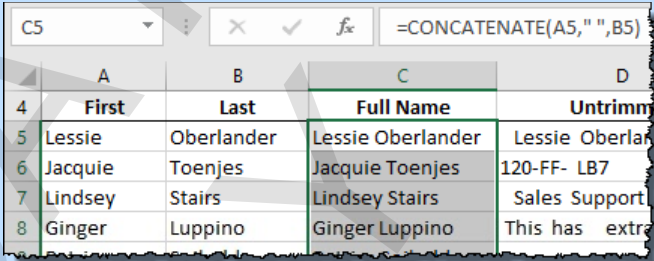
- PROPER capitalizes the first letter of a string, and then also any other letters that follow any character other than a letter. In general, it capitalizes the first letter of each word.

  PROPER(text)

Exercise: Combining and trimming text

Exam Objective: MOS Excel Core 4.3.2, 4.3.3

Do This	How & Why
1. Open `Text`.	From the `Special Functions` data folder. The workbook contains data for experimenting with text functions.
2. In column C, use CONCATENATE to combine the first and last names.	
a) Select C5:C10.	
b) Type or enter `=CONCATENATE(A5," ",B5).`	The word "concatenate" just means combine, and that's what this function does. It combines the first name (in A5) with a single space (" "), and then the last name (B5).
c) Press **Ctrl+Enter**.	To combine the first and last names, with a space between.

			fx	=CONCATENATE(A5," ",B5)
	A	B	C	D
4	**First**	**Last**	**Full Name**	**Untrimm**
5	Lessie	Oberlander	Lessie Oberlander	Lessie Oberla
6	Jacquie	Toenjes	Jacquie Toenjes	120-FF- LB7
7	Lindsey	Stairs	Lindsey Stairs	Sales Support
8	Ginger	Luppino	Ginger Luppino	This has extra

Do This	How & Why
3. Save the workbook as `My Text`.	
4. Observe the data in column D.	These are text values, but there are extra spaces all over: at the beginning, at the end, in the middle.
5. In column E, use TRIM to remove extra spaces from the values in column D.	
a) Select E5:E8.	
b) Type or enter `=TRIM(D5).`	
c) Press **Ctrl+Enter**.	To trim the text.

Untrimmed	Trimmed
Lessie Oberlander	Lessie Oberlander
120-FF- LB7	120-FF- LB7
Sales Support	Sales Support
This has extra space	This has extra space

Do This	How & Why
6. Save the workbook.	

Extracting text

Another common use of text functions is extracting text from other text. For example, you might want just the last three digits of a code, or perhaps a first name when you have a full name. Excel has functions for extracting from the beginning, middle, or end of a string, and for finding a particular character or length of a string. By putting these together, you can extract almost anything you need.

Exam Objective: MOS Excel Core 4.3.1

- The LEFT function extracts any number of characters from the left of a string.

  LEFT(text, [num_chars])

- The RIGHT function does the same, but from the right side of the text.

  RIGHT(text,[num_chars])

- The MID function works slightly differently, because you have to tell it with what character to begin extracting (the start_num argument).

  MID(text, start_num, num_chars)

Getting information about text

Sometimes, you need to figure out other information before you can use LEFT, MID, or RIGHT. You might, for example, need to know how long a string is before you know how many characters to extract. Or, you might need to know where a certain character is.

- Use LEN to get the length of a string.

  LEN(text)

- Use FIND to find a particular character or string within another string. FIND returns the number of the position of find_text within a string. You can also give the position relative to a start_num position in within_text.
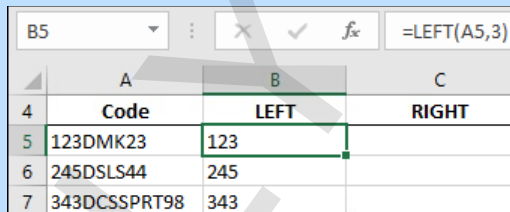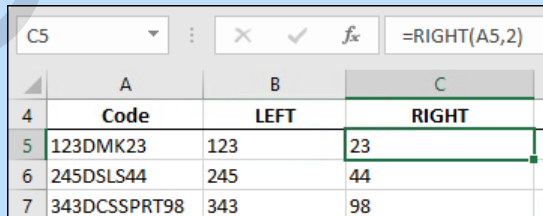
  FIND(find_text, within_text, [start_num])

Exercise: Extracting parts of text strings

`My Text` is open.

Exam Objective: MOS Excel Core 4.3.1

Do This	How & Why
1. Activate the **Extracting** worksheet.	Column A contains employee codes. Each has three digits at the beginning and two at the end. In the middle is the letter "D," followed by a text code that identifies a department. The department codes are not of uniform length.
2. In column B, extract the three-digit prefix from the codes.	
a) Select B5:B7.	
b) Type or enter `=LEFT(A5,3)`.	In plain language, you want the leftmost three characters of the string in cell A5.
c) Press **Ctrl+Enter**.	To enter the function. The three-digit codes appear in column B.

B5			✕	✓	*fx*	=LEFT(A5,3)

	A	B	C
4	**Code**	**LEFT**	**RIGHT**
5	123DMK23	123	
6	245DSLS44	245	
7	343DCSSPRT98	343	

3. In column C, extract the two-digit suffixes from the codes.	
a) Select C5:C7.	
b) Type or enter `=RIGHT(A5,2)`.	
c) Press **Ctrl+Enter**.	To enter the function and return the two-digit codes.

C5			✕	✓	*fx*	=RIGHT(A5,2)

	A	B	C
4	**Code**	**LEFT**	**RIGHT**
5	123DMK23	123	23
6	245DSLS44	245	44
7	343DCSSPRT98	343	98

4. Examine the alphabetic parts of the codes.	They always begin with D, but then the rest could be any number of characters long (these are department codes).
5. To use MID to extract the department, what information will you need about the code strings?	You will need to know where the "D" appears, and you will need to subtract 5 (the number of numeric code numbers) from the total lengths of the strings.
Continued...	

Do This	How & Why
6. In column D, find the position of "D" within the codes.	
a) Select D5:D7.	
b) Type or enter =FIND("D",A5).	This function will find the position of the letter "D" within the string in A5.
c) Press **Ctrl+Enter**.	To enter the functions. The D is in the fourth position in each string.
7. In column E, use LEN to calculate the lengths of the codes.	Select the cells, enter or type =LEN(A5), then press **Ctrl+Enter**.
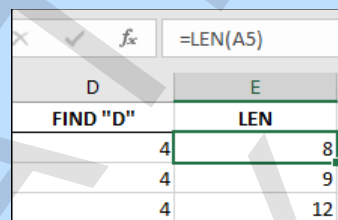
	fx	=LEN(A5)
D		**E**
FIND "D"		LEN
4		8
4		9
4		12

Do This	How & Why
8. In column F, enter a function to return the department codes.	
a) Select F5:F7.	
b) Type or enter =MID(A5,D5,E5-5).	MID is a little more complex. It takes text from the code (A5), starting with the position indicated by D5 (where the "D" is) and gives you the number of characters equal to the length of the code (E5) minus the five characters that are digits.
c) Press **Ctrl+Enter**.	To return the department codes.
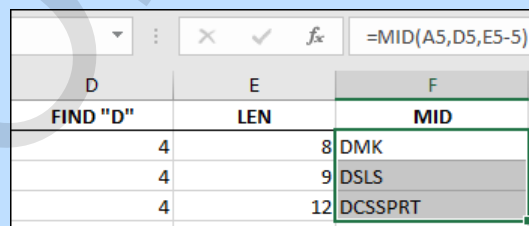
			fx	=MID(A5,D5,E5-5)
D		**E**		**F**
FIND "D"		LEN		MID
4		8		DMK
4		9		DSLS
4		12		DCSSPRT

9. Save and close the workbook.

Assessment: Text functions

What is the result of the function CONCATENATE("My","Name")? Choose the one correct answer.

- my name
- My Name
- myname
- **MyName**

The LEFT, RIGHT, and MID functions all take the same arguments. True or false?

- True
- **False**

The LEN function takes only a single argument, a text string. True or false?

- **True**
- False

Module C: Other functions

Excel has functions for almost any purpose you can imagine. In particular, it has an enormous variety of statistical and financial functions. You can choose a category in the Insert Function window to see the functions in that category, and of course, use the Help system to get information about any function.

You will learn:

- How to view a category of functions

- How to use MIN, MAX, and COUNT

- About financial functions and how to use FV

Statistical functions

Excel has dozens of built-in statistical functions that you can use to analyze data and help you make predictions. AVERAGE, of course, is a simple statistical function. But if you use the Insert Function window, you can choose the Statistical category and see many more.
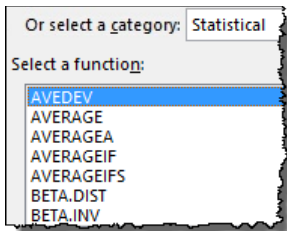
Here are a few of Excel's statistical functions.

- COUNT, COUNTA, and COUNTBLANK which return the count of cells containing numbers, non-blanks, and blanks, respectively.

- FORECAST, which returns a value projected along a linear trend.

- MAX and MIN, which return the maximum and minimum values for a set.

- RANK.AVG, which returns the ranking of a value within a set.

These are just a few of many.

Using MIN and MAX

MIN and MAX help you determine the smallest and largest values in a range.

Exam Objective: MOS Excel Core 4.1.3

- MIN returns the smallest value of a set of numbers.

  MIN(number1, [number2], ...)

- MAX returns the largest value of a set of numbers.

  MAX(number1, [number2], ...)

Exercise: Calculating minimum and maximum sales values

Exam Objective: MOS Excel Core 4.1.3

Do This	How & Why
1. Open `Functions`.	From the `Special Functions` data folder. The first worksheet contains a list of invoices.
2. Select J10, then click f_x .	The Insert Function button is on the formula bar. The Insert Function window appears.
3. In the "Or select a category" list, click **Statistical**.	To display a list of statistical functions. Scroll through to see what is available.
4. Close the window.	
5. In J10, enter `=MIN(H10:H1007)`.	To calculate the minimum invoice's Total Sale amount (15.30).
6. In K10, enter `=MAX(H10:H1007)`.	To calculate the maximum Total Sale amount, which is a bit lower than the average.
7. Save the workbook as `My Functions`.	

Using statistical functions

f_x	=MAX(H10:H1007)

J	K
Minimum Sale	**Maximum Sale**
$ 15.30	$ 273.60

Using COUNT, COUNTA, and COUNTBLANK

COUNT, COUNTA, and COUNTBLANK are useful when you need to know how many of something is in a range. COUNT tells you how many cells contain numbers, while COUNTA counts non-blank cells, regardless of data type. COUNTBLANK tells you how many blank cells are in the arguments.

Exam Objective: MOS Excel Core 4.1.4

- COUNT gives you the total number of cells in the argument ranges that contain numbers. It treats text or logical values as "not a number."

COUNT(value1, [value2], ...)

- COUNTA works slightly differently, counting non-blank cells in a range. Use COUNTA to get a count on a range containing text, for example.

COUNTA(value1, [value2], ...)

- COUNTBLANK counts the empty cells in a range you specify.

COUNTBLANK(range)

Exercise: Counting the number of customers with sales

My Functions is open.

Exam Objective: MOS Excel Core 4.1.4

Do This	How & Why
1. Activate the Counts worksheet.	It contains customer sales data by product. Not all customers bought all products. You'll enter Count functions in G3:J3 to determine how many customers bought each product.
2. Enter the COUNT function in G3:J3.	
a) Select G3:J3.	
b) Type =COUNT(B3:B48).	The COUNT function counts how many cells in the argument's range have numbers in them.
c) Press **Ctrl**+**Enter**.	To enter the function in the range. You can easily see how many customers bought each product.
3. Save the workbook.	

Using COUNT to determine how many customers bought each product

Total Customers			
Indus	**Phoenix**	**Tucana**	**Vela**
36	32	43	45

Financial functions

Excel's financial functions category contains many useful functions for finance calculations, such as interest, depreciation, and future value.

```
Or select a category: Financial
Select a function:
ACCRINT
ACCRINTM
AMORDEGRC
AMORLINC
COUPDAYBS
COUPDAYS
COUPDAYSNC
```
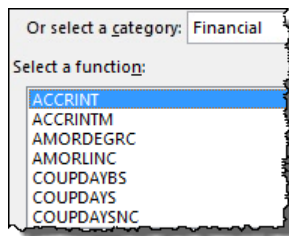
Using FV

The FV function calculates the *future value* of an investment according to the initial value, a rate of return, and a number of periods of return.

`FV(rate,nper,pmt,[pv],[type])`

Exam Objective: MOS Excel Expert 3.4.5

- *rate* is the interest rate per period (period can be defined however you want).
- *nper* is the number of periods over which the investment will compound.
- *pmt* is the payment made each period, if this is a loan being calculated. This argument is required, but if you're calculating the value of an investment, it will be blank (you put a comma after a comma to indicate a blank argument).
- *pv* is the present value, or the initial amount invested or loaned. If this is an investment, you're paying money, so the value should be negative.
- *type* specifies options for payment type ("1," if payments are at the beginning of periods; "0," or omitted at the end of periods).

Other financial functions

Excel has dozens of financial functions. Here are a few that you might find useful. You can look up their syntax by finding them in the Insert Function window, then clicking **Help on this function**.

- PMT calculates the payment on a loan or investment given the rate, number of periods, and present value.
- PV calculates the present value on a loan or investment given the rate, number of periods, and payment amount.
- IRR calculates the internal rate of return for a series of cash flows.
- NPER calculates the number of periods for an investment given constant payments and interest rate.
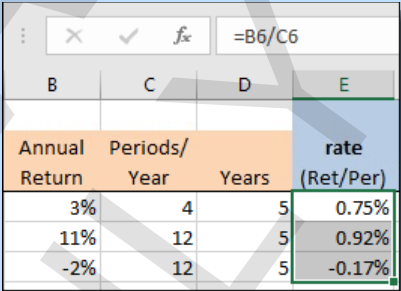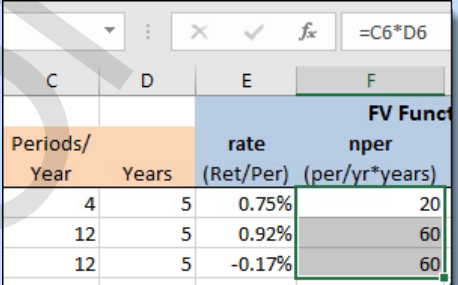
These are just a few of many.

Exercise: Comparing future value of investments

`My Functions` is open.

Exam Objective: MOS Excel Expert 3.4.5

Do This	How & Why
1. Activate the **Financial** worksheet.	This worksheet is for comparing three investment scenarios. Row 6 will be a fairly low-risk investment with a near-guaranteed 3% rate of return over 5 years. The load will compound quarterly (four times per year). Rows 7 and 8 are two possible outcomes for a riskier investment that compounds monthly, or 12 times per year. Best case would be an 11% yearly return, but it could lose 2% per year.
2. Observe columns E:I.	The FV function calculates future value according to the five arguments shown in bold. You'll calculate three of those arguments using the information in the other columns, then use FV in column J.
3. In column E, calculate the return rate per period by dividing the annual return by the periods per year.	The formula will look like this.

| **4.** In column F, calculate the number of periods by multiplying periods/year by years. | The formula looks like this. |

| **5.** Observe column G. | If you were calculating the future value of a loan, you would enter a payment amount as the pmt argument of the FV function. For an investment, you would leave this argument blank. |

Do This	How & Why
6. In column H, calculate the present value, pv, by taking the negative of the investment amount.	An investment is a negative present value, because you pay that money in. The formula looks like this.

Do This	How & Why
7. Observe column I.	The FV function also lets you specify a payment type for a loan: 1, for the beginning of a period; 0, or omitted, for the end. You'll leave this out.
8. In column J, calculate the future values for the loans.	
a) Select J6:J8.	
b) Type or enter =FV(E6,.	The rate is the value in column E (annual return divided by periods per year).
c) Type or enter F6,.	To enter the number of periods (nper) argument. The formula looks like this so far.

Do This	How & Why
d) Type another comma.	To leave the pmt argument blank, because this is not a loan.
e) Type or enter H6).	To enter the present value argument, which is the negative of the investment amount.
f) Press **Ctrl**+**Enter**.	To enter the formula in the range.

Do This	How & Why
9. Which investment do you think is the best choice?	It really depends on how likely you think the second investment is to achieve the higher return rate.
10. Save and close the workbook.	

Assessment: Other functions

Which function should you use to determine how many cells in a range contain names rather than blank cells?

- COUNT
- **COUNTA**
- COUNTBLANK

The FV function can be used to calculate future values of both investments and loans. True or false?

- **True**
- False

Summary: Special functions

You should now know:

- About how Excel handles dates and times, how to use functions to get information about dates, and how to calculate dates and times

- How to use functions to combine, trim, or get information from text values

- About some of Excel's statistical functions, how to use the MIN, MAX, and COUNT functions, about Excel's financial functions, and how to calculate future value of an investment or loan using FV

Synthesis: Special functions

In this synthesis exercise, you'll do three things. First, you'll create a product development schedule based upon the amounts of work (in workdays) between milestones of the project. Next, you'll extract the numeric and alphabetic parts of a code that contains both. And finally, you'll use the PMT function to calculate loan payments.

1. Open Special Functions Synthesis from the `Special Functions` data folder. The first worksheet contains a structure for a product development schedule.

2. Use WORKDAY.INTL to complete the schedule. Use the Holidays2016 range as the holidays argument, and use any weekend definition you want. The function you create in C4 should be able to be copied to E4, G4, and I4.

Product Development Schedule

Start Date	Days to Beta	Beta	Days to Review	Review Complete	Days to Final Draft	Final Draft	Days QA/ Final Fix	Complete
1/1/2016	20	2/1/2016	3	2/4/2016	5	2/11/2016	3	2/17/2016

3. Save the workbook as `My Special Functions Synthesis`.

4. On the codes worksheet, in column B, construct a function that will extract the numeric portion of the code. There are different quantities of numbers in the codes. The first letter of the alphabetic portion will always be "A," so if you find its position, you will want one fewer character from the left.

5. In column C, construct a function to extract the alphabetic portions of the codes. To get the number of characters for this part, first subtract 1 from the position of the "A," then subtract that difference from the length of the code.

Code	Numeric	Alphabetic
1243AHOBH	1243	AHOBH
25AHOWMNNH	25	AHOWMNNH
459824APZHELH	459824	APZHELH

6. Save the workbook.

7. On the Payments worksheet, in cell B6, enter the PMT function to calculate the payment on the loan using the information in cells B3, B4, and B5. PMT is similar to FV in terms of its arguments. Hints: (1) Use the Insert Function feature to guide you through the arguments. (2) The rate argument is per period, so remember to divide the yearly interest rate by the number of months in a year. (3) The pv argument should be negative for a loan. 4) You won't need the fv or type arguments.

Continued...

8. Try changing the loan amount, interest rate, or months to see how it affects the payment.

Loan Amount	$	120,000
Interest Rate		6%
Months		360
PMT	$	719

9. Save and close the workbook.

Chapter 4: Importing and Exporting

You will learn:

- About the Power Pivot Data Model, and how to import and work with data from other sources in Excel

- How to export Excel workbooks to various formats, and why you would use them

Learning time: 40 minutes

Module A: The Power Pivot Data Model

Excel is often used as a processing program for data that comes from elsewhere: database files, web queries, and information exported from other programs as text files. Excel has many tools for importing or working directly with external data sources. One of the most powerful features is the Power Pivot add-in. When you use data (from Excel or elsewhere) in Power Pivot, you create a Data Model within Excel that gives you enormous data-processing power.

You will learn:

- How to import data from a text file into Excel

- How to get external data from a Microsoft Access database

- How to use Power Pivot to access features of the Power Pivot Data Model, including cube functions

Text files

Excel can open any text file, and it has tools for spreading text data through columns. Some text files are structured so that their data can be more easily read by other programs. These include comma-separated variable files (CSV) and tab-delimited files.

The Data tab has great tools for getting all sorts of external data into Excel, and this is often the best way to get data from text files. When you use these tools, Excel guides you through a process of converting the data into columns and controlling formats.

Importing text files

There are many types of text files you might want to import into Excel, including txt, csv, and more.

Exam Objective: MOS Excel Core 1.1.2
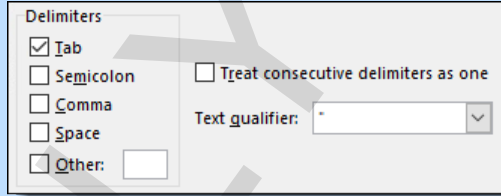
1. In Backstage view, click **Open**, then browse files.

    To display the Open window.

2. In the file type list, click **Text Files**.

3. Navigate to and open the file you want.

    Excel displays the first of the Text Import Wizard windows.

4. Select the type of data arrangement for the text file.

    • *Delimited* means that there is some character that distinguishes one piece of data from another. In a csv file, that would be a comma; in a tab-delimited file, it would be a tab.

    • *Fixed width* means that each type of data is in a column with spaces in between.

5. Select any other options you want, then click **Next**.

    The next step of the Wizard depends on what kind of file you chose before.

6. Specify the data delimiters or set widths for the columns (depending on the type of file), then click **Next**.

7. Specify formatting options for any columns.

    a) Click a column heading.

    b) Click an option under Column data format.

    c) Repeat for all columns you want to format.

8. Click **Finish**.

The data appears in the worksheet. If it doesn't turn out as you intended, try again with a different delimiter, or column widths.

Exercise: Importing a text file with values delimited by commas

Exam Objective: MOS Excel Core 1.1.2

Do This	How & Why
1. In Backstage view, click **Open**.	
2. Click **Computer**, then click **Browse**.	To display the Open window.
3. In the File Type list, click **Text Files**.	You will open a text file that has values delimited by commas.
4. Open `Invoices`.	The txt file in the chapter's data folder. The first of the Text Import Wizard windows appears. Here, you specify the type of file that you're opening.
5. Observe the options under "Choose the file type that best describes your data."	**Delimited** is selected. A comma-separated variable file, like this one, has its data values delimited by—not surprisingly—commas; so, it's the right choice. But some text files are arranged in columns; for those, Fixed width is a better option.
6. Click **Next**.	To move to the next step of the wizard, which lets you specify the specific delimiter character.
7. Clear **Tab**, then check **Comma**.	This is a comma-separate file. Notice that the preview of the data now shows it nicely separated into columns.
8. Click **Next**.	To move to the third wizard step, which gives you formatting options for each column.
9. Click **Date**.	To specify a date format for the data in the first column. You can format any column by clicking its heading and then clicking a format option.
10. Click **Finish**.	To finish the steps of the wizard and import the data starting in cell A1 of the current worksheet.
11. Widen column A, then select a cell in it.	Notice that the format is Date.
12. Save the workbook as an Excel Workbook called `My Imported Data`, then close it.	You will need to click **Excel Workbook** in the Save as type list.

External connections

In addition to simply opening data into Excel from other formats, you can create *external connections* to other data. When you use an external connection, the data in Excel are actually linked to the source of the data, so that you can refresh the Excel data to reflect any changes to the source data.

Excel supports external connections to all sorts of files and formats.

Access   Microsoft Office's database program.

Web   Data from web sources.

Text   Data from text files.

Other Sources   There are many possibilities, such as SQL Server, the Windows Azure Marketplace, XMLS, OLEDB, and ODBC.
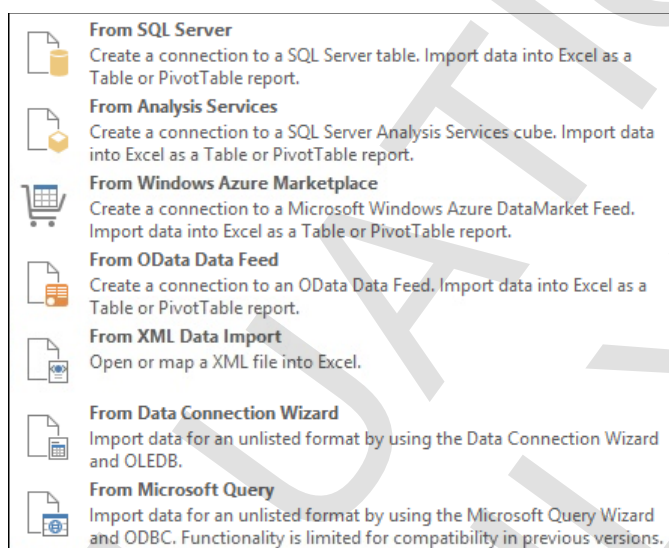
Queries   You can use the options in the Get and Transform Data group on the Data tab to create queries for bring data into Excel from various sources from text files to databases or other types of data feeds.

Getting external data

When you use options in the Get External Data group, you create a connection, a bridge, between Excel and data in another source.

Exam Objective: MOS Excel Expert 3.4.1
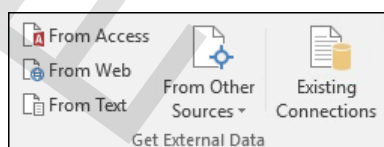
The exact procedure for establishing an external connection is different for each type, but the general process is similar.

1. In the Get External Data group, click a **From** button.
   If you clicked **From Other Sources**, then click an option.

2. Either navigate to and open the file containing the data (for Access, Text, Web, and XML sources), or use the Data Connection Wizard.

3. Define the specific data to which you want to connect.

4. Define how you want the data to appear in Excel.

The data will appear in Excel and be linked to the source data, meaning you will see changes to the source data reflected in Excel.

Using Get & Transform Data options

The new Get & Transform Data options on the Data tab allow you create queries that both retrieve data and change it on the fly. Those queries can be saved for later use, or shared with others. After you load a query, the data comes into Excel as part of the Power Pivot data model, which gives you all sort of tools and power.
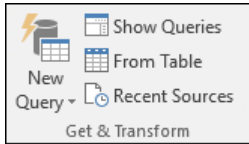
Exam Objective: MOS Excel Expert 3.4.1

Some of how Get & Transform works varies depending on the data source you're connecting to, but most of what you do happens in the Query Editor window using a consistent interface. In general, it works as follows.

1. Click **New Query**, then click a type of source, then click a data source.

The New Query button is in the Get & Transform group on the Data tab. the types of sources are shown here.
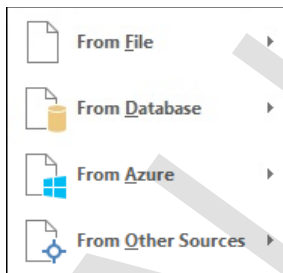
2. Navigate to and select your data source, then click **Import**.

The Navigator appears. Here you can select tables, and either load or edit them before loading.

3. To edit a source, select it on the left and then click **Edit**.

To display the Query Editor window. At this point, you still haven't loaded the data into Excel. Rather, the Query Editor allows you manipulate how the data will come into Excel when you do load it. You can choose or remove data (by row or column), change data types, and even combine the query with other queries to import more complex data sets.
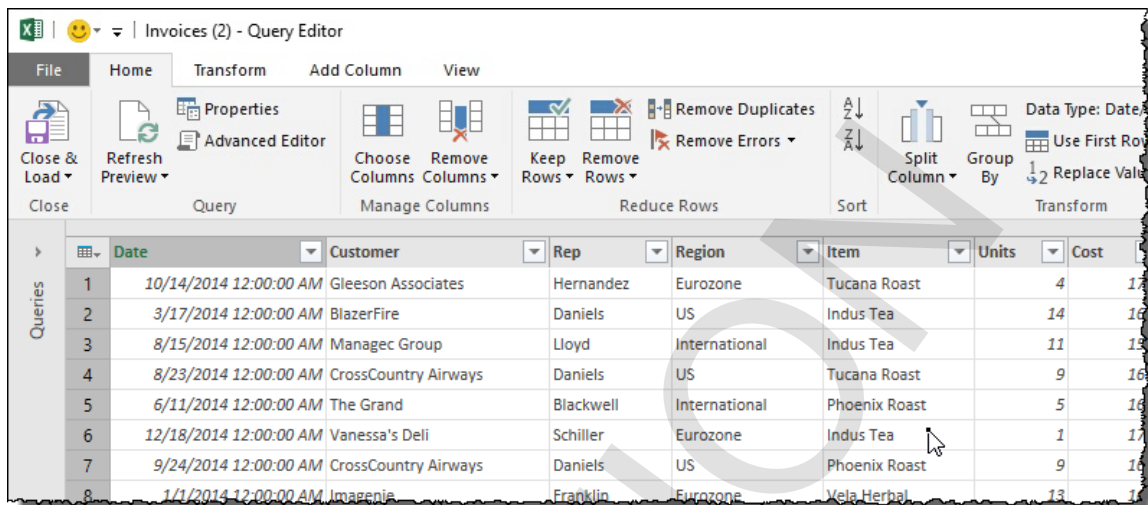
4. When you have the query the way you want it, click **Close & Load**.

The data load into Excel, where you can work with it in many ways.

Exercise: Connecting to a Microsoft Access data source

If you want to see how changing data in a linked source affects that data in Excel, you will need to have Microsoft Access installed on your computer.

Exam Objective: MOS Excel Expert 3.4.1

Do This	How & Why
1. Create a new, blank workbook.	You will use a query to get information from an Access Database, so you can use its information in Excel.
2. On the Data tab, click **New Query > From Database > From Microsoft Access Database**.	In the Get & Transform group. The Import Data window (another version of the Open window) appears.
3. Navigate to and select `Invoices.`, then click **Import**.	The Access database file in the current chapter's data folder. Excel takes a few seconds to establish the connection, and then the Navigator window appears. If the database had many tables, you would see them all on the left, but this Access file has only one table.
4. Select the Invoices table, then click **Edit**.	The Edit button is in the lower-right of the window. To open a view of the table in the Query Editor window. Here, you can manipulate the data in various ways before you open it. Those steps then become part of the query, so if you run it in the future, all of this will happen automatically.

Do This	How & Why
5. Observe the Manage Columns and Reduce Rows groups.	On the Home tab of the ribbon. Here, you can manage which rows and columns will be included in the imported data.
6. Change the Date column to show only dates (not times).	
a) Select the column.	Click its heading.
b) In the Transform group, click the Data Type dropdown, then click **Date**.	To change the format of the column.
c) Observe the Applied Steps area.	In the Query Settings pane, on the right of the window. Each step you take while getting and transforming data is reflected here, and then saved with the query. When you run the query in the future, it will perform all the steps again automatically.
7. Click the Close & Load button.	Don't click the dropdown arrow, but if you do, click **Close & Load** in the menu.
	To import the data as a table. Now, all of Excel's table and data tools are available to you.
8. Save the workbook as `My Access Invoices`.	In the current chapter's data folder.
9. Observe the External Table Data group on the Table Tools Design tab.	When you have a table linked to external data, you have options here for managing the data. Note the Unlink button. You can disconnect the link to the Access file, if you want. If you do have linked data, you want to be sure that you can access it from where you are.
Continued...	
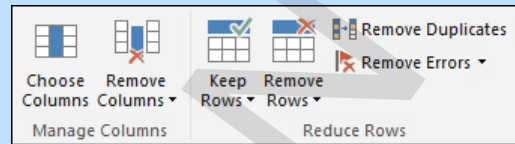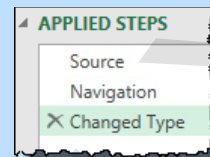
Do This	How & Why
⚠ **CAUTION:** This step can be performed only if you have Microsoft Access installed on your computer. If you don't, you can skip this step.	
10. Demonstrate the link between the Access file and the Excel table.	
a) Close My Access Invoices.	
b) Open `Invoices.mdb` in Microsoft Access.	It is in the same folder. If you get a security warning, click the button to enable the content.
c) Open the **Invoices** table, and then change the customer for the first invoice to `Red Rock Mountain Tours`.	Simply select the customer text, type over it, and press **Enter**.
d) Close Microsoft Access.	
e) Re-open `My Access Invoices`.	If you get a security warning stating that "Data connections have been disabled," click **Enable Content**.
f) On the Data tab, click **Refresh All**.	The customer for the first invoice changes to reflect the change in the Access file.
11. Save the workbook, and then close it.	

Power Pivot

Power Pivot is an Excel add-in that gives you access to an enormous range of data processing features. If you are familiar with PivotTables, then the basics of working with PowerPivot will be familiar to you. It is, essentially, a tool for rearranging and summarizing data.

But that sells it far short. The most powerful feature of Power Pivot is that when you use it, you gain access to the Power Pivot Data Model, which is a way of connecting related tables of data and mining them for information. You can do this with Excel alone, but Power Pivot has a number of advantages.

- Compression allows for workbooks using large data sets to be more than 10 times smaller than normal Excel Workbooks, improving performance and portability.

- Handle data sets with 100 million rows.

- Create relationships among tables using an intuitive, visual interface.

- Write powerful formulas in the Data Analysis Expressions (DAX) expression language.

- Access data in the model by using CUBE functions.

Power Pivot is a powerful set of tools and a very deep subject, and this is only the tip of the iceberg.

Loading Power Pivot

Not all versions of Microsoft Office (and Excel) come with the Power Pivot Add-In. If your version includes it, you will need to load it either directly or indirectly.

- In the Options window, click **Add-Ins**; in the Manage list, select **COM Add-ins** and click **Go**; check **Microsoft Power Pivot for Excel**; and click **OK**.

    The Power Pivot tab will be added to the ribbon.
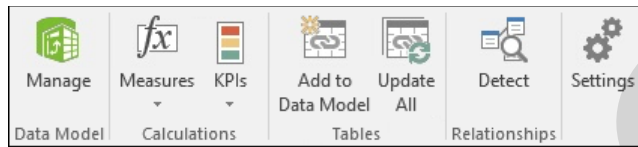
- On the Data tab, click **Manage Data Model**.

    If Power Pivot is not loaded, Excel will display a message prompting you to enable the Data Analysis add-ins to use the feature. Click **Enable**. The Power Pivot tab will be added to the ribbon, and the Power Pivot window will open.

Using Power Pivot

Power Pivot can help you perform all sorts of powerful analyses on large data sets, both within and external to Excel. It is especially useful for large data sets, and for managing related tables. Power Pivot is a complex feature, but here are some of the basics of how to use it.

Exam Objective: MOS Excel Expert 3.4.4

1. On the Power Pivot tab, click **Manage**.

    To open the Power Pivot window. Power Pivot is a separate, data analysis environment, but it has many tools that will be familiar to you from working with Excel, such as PivotTables, filtering, sorting, and formatting.

2. Use the Get External Data group to create a connection to a data source.

    You can also use the Power Pivot tab to add Excel tables to the Data Model and then work with them.

3. Use the Power Pivot ribbon to analyze and format your data as you wish.

Exercise: Adding Excel tables to the Data Model

To perform this task you need to have a version of Excel that includes the PowerPivot add-in. If you do not have it, your instructor can demonstrate or skip the activity.

This exercise is meant to be an introduction to several concepts within the world of the Power Pivot Data Model. It covers a lot of ground, including loading Power Pivot, relating tables, creating a calculated filed, and creating a PivotTable from related tables.

Do This	How & Why
1. Open `Power Pivot`.	From the current chapter's data folder. The worksheet contains four lists of data: one for invoices, one for customers, one for items, and one for reps.
2. Observe the data in the lists.	These lists have related in formation in them. The Invoices 2014 list has a column for customer, while the Customer list has a single row for each customer, and then tells who that customer's rep is and what the customer's discount is. Items is a list of the items and their prices, and Item is a column in the Invoices list. Finally, the Reps table shows which region each rep belongs to. In order to get the most out of these tables, you will relate them. But first, you have to make them into Excel tables.
3. Make all four lists into tables, naming them as follows. • `fInvoices`. The "f" at the beginning of the name distinguishes this as a "fact" table, or a "transaction" table. • `dCustomers`. The "d" mars this as a "dimension" table, or what an Excel user might think of as a lookup table. • `dItems`. • `dReps`.	Select a cell within each table, click **Table** (on the Insert tab), then click **OK**. On the Table Tools Design tab, click in the Table Name box to name the table.
4. Open Power Pivot.	On the Data tab, click **Manage Data Model**. If the Power Pivot add-in is already loaded, Power Pivot will open directly. If not, you'll get a message prompting you to enable Data Analysis add-ins. Click **Enable**. The Power Pivot window has its own ribbon and interface. No data are showing because you have to first import data, get data, or add it to the data model.
5. Add the four tables to the Data Model.	For each table, switch to Excel, select a cell within the table, and then click Add to Data Model (on the Power Pivot tab, which is added to the Excel ribbon when you load the Power Pivot add-in). Each time you add a table, you will see a tab added in the Power Pivot window. Some things about working with tables in Power Pivot will be familiar, others will be different.

Do This	How & Why
6. Return to Excel and save the workbook as `My Power Pivot.`	

My Power Pivot with tables and the Power Pivot ribbon tab

Creating relationships

One great use of Power Pivot is to relate tables, and then use them together in PivotTables and other kinds of data collection and reporting.

Exam Objective: MOS Excel Expert 3.4.4

1. Add at least two tables to the Data Model.

   a) For Excel data, first convert all tables to table format, and name them.

   You should have a consistent, memorable naming schedule. One common distinction among table types is to divide them in the *fact tables* (what you might think of as a transaction table, such as invoices) and *dimension tables* (or lookup tables, such as a list of product information). You can add "f" or "d" to the beginning of table names to distinguish their types.

   b) For each table, select any cell within it, and then click **Add to Data Model**.

   On the Power Pivot tab.

   The Power Pivot window opens, and a tab is created for the table you added to the data model. Switch back to Excel to add any other tables to the Data model.

2. In Power Pivot, on the Home tab, click **Diagram View**.

   In the View group.

   Power Pivot shows you each of the tables in a diagram. You can click and drag to create relationships.

3. Drag to create a relationship between two tables.

   Again, this is a big subject. You'll want to drag from a dimension table to a fact table, in general.

   You will see lines between the tables in the diagram indicating the direction and type of relationship. The relationship shown here is a one-to-many relationship between a dimension table of items (dItems) and a fact table of invoices (fInvoices).
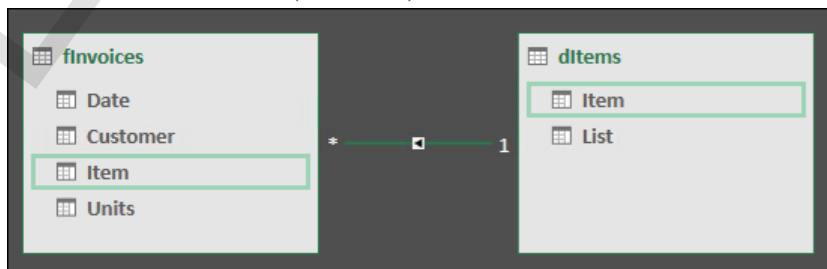
After you've created relationships, you will be able to use related tables together in PivotTables. You'll also be able to add calculated columns that refer to data in different related tables.

Working with related tables

After relating tables in Power Pivot, the world of the Data Model opens up to you. You can use references to fields in related tables in calculated fields in the tables, and you can use the related tables together to create pivot tables.

Exam Objective: MOS Excel Expert 3.4.4

Calculated fields in tables in Power Pivot refer to fields in the tables, rather that cell references. Functions that access the data model for these kinds of calculations are called *Data Access Expression (DAX) functions*. In a calculated column or field, you access fields in a related table by using the RELATED function.

When you create a PivotTable from within Power Pivot, you will see the familiar Excel PivotTable interface. But if you have related tables in your Data Model, you will see all of those tables in the Field Chooser. Simply create the PivotTable using the skills you already have.

Exercise: Working with related tables in Power Pivot

My Pivot Tables is open, and the previous activity has been completed. If your instructor demonstrated the previous activity, he or she should continue with this one.

In this exercise, you'll relate the four tables you added to the Data Model, create a calculated field in one of the tables that uses related table fields, and then create a PivotTable based on multiple, related tables.

Do This	How & Why
1. In Power Pivot, click **Diagram View**.	In the View group on the Home tab. This is a visual representation of the tables in the data model. You can create relationships among the tables here by dragging.
2. Drag the Customer field from the dCustomers table to the fInvoices table.	To create a one-to-many relationship between the Customers table and the Invoices table.
Continued...	
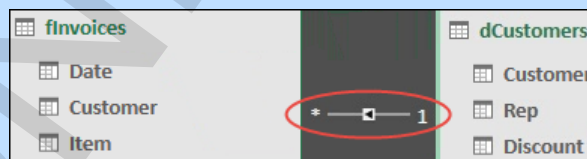
Do This	How & Why

3. Create the following relationships.

- One-to-many from dItems to fInvoices, using the Item field.

- One to many from dReps to dCustomers, using the Rep field.

Drag the relevant field from the one table to the other. You can drag the tables around diagram view to lay them out as you like. In the end, your relationships should look something like this. Notice that when you point to a relationship line, the related fields are highlighted (as the Rep fields are in this figure).
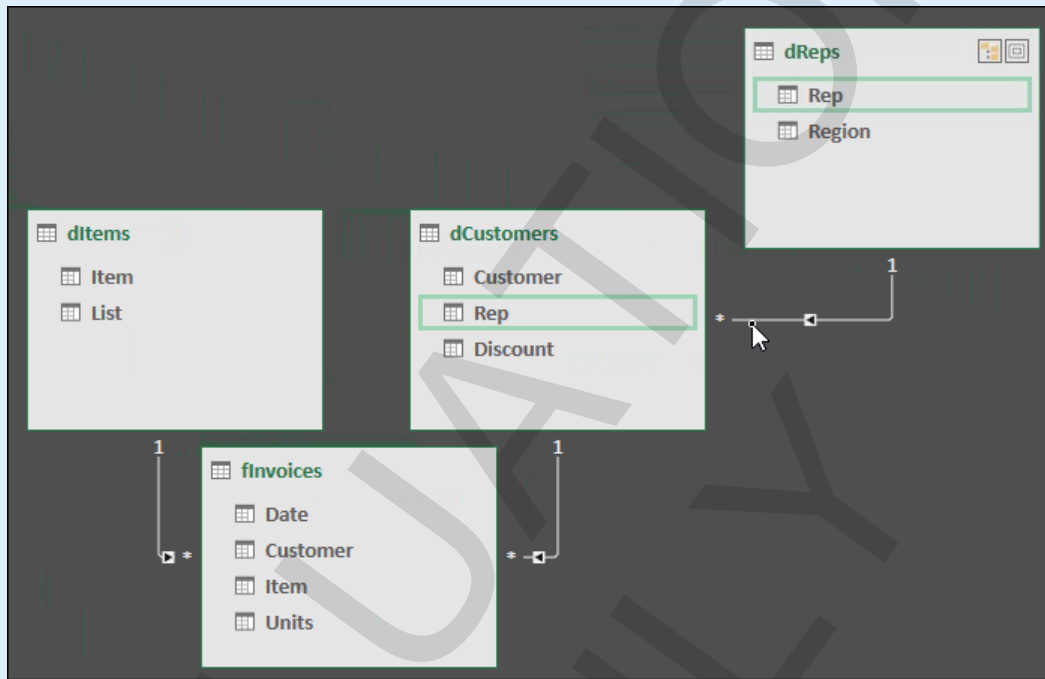
4. Return to Data view.

Click the Data View button.

5. Add a calculated revenue column to the fInvoices table.

To do this, you will multiply the Units in the fInvoices table by the List price in the dItems table by the Discount in the dCustomers table. To create this formula, you will use the RELATED function.

a) In the fInvoices table, click **Add Column**, then type =.

b) Start to type fInvoices, then select **fInvoices[Units]** from the AutoComplete list and press **Tab**.

To being to build this formula.

c) Type *RELATED(, select dItems[List], press tab, and type).

The RELATED function is a Data Access Expression (DAX) function that allows you to perform calculations using data from a related table. Such functions have row context, meaning they know to operate on data from the current row. So far, the formula should look like this.

=fInvoices[Units]*RELATED(dItems[List])

Do This	How & Why
d) Type or enter *(1-RELATED(dCustomers[Discount])), then press Enter.	You have to subtract the Discount percentage from 1, then multiply. To create the calculated field.
6. Rename the column Revenue, and format it as currency.	
a) Right-click the column heading, click **Rename Column**, type Revenue, and press **Enter**.	
b) Click $.	In the Formatting group. The column should look like this. Revenue $61.56 $181.44 $124.74 $107.73
7. Save the workbook.	
8. Create a Pivot Table showing the sum of revenue by region and product.	Creating a PivotTable works exactly the same in Power Pivot as in Excel.
a) In Power Pivot, click **PivotTable**.	A window appears asking if you want this in an existing or new worksheet (the default).
b) Click **OK**.	A new worksheet appears with the familiar PivotTable interface. Notice that all four tables appear in the PivotTable Fields pane. Choose fields to add to report: Search ▷ ⊞ dCustomers ▷ ⊞ dItems ▷ ⊞ dReps ▷ ⊞ fInvoices
c) Create the PivotTable as shown.	You will need to get the fields from at least two of the tables. ▼ FILTERS ‖‖ COLUMNS Item ☰ ROWS Σ VALUES Region Sum of Reven... The PivotTable appears on the worksheet, summrizing data pulled from three tables.
9. Update the workbook.	An example follows the exercise.

The PivotTable created from related tables

Sum of Revenue	Column Labels				
Row Labels	Indus Tea	Phoenix Roast	Tucana Roast	Vela Herbal	Grand Total
Eurozone	$9,794.52	$13,787.12	$13,372.20	$11,502.40	$48,456.24
International	$9,779.94	$10,800.80	$7,806.15	$10,152.80	$38,539.69
US	$6,480.00	$6,679.20	$6,417.63	$6,048.00	$25,624.83
Grand Total	$26,054.46	$31,267.12	$27,595.98	$27,703.20	$112,620.76

Cube functions

Cube functions allow you to access data within the Power Pivot Data Model (which is sometimes called a "cube") directly, without needing a Pivot Table. In order to use cube functions, you need to first have a connection to a data source. Then, they work, in general, by taking arguments that specify first the connection name, then what you want to get from the data, which can be aggregated values, ranked members of the set, of value of a property of a particular member.

Cube functions can be complicated to enter directly, but there is a great shortcut to understanding their syntax. You can created a PivotTable, and then convert all of its cells to cube functions. You can then copy and modify those functions to get the data you want elsewhere in your spreadsheets.

Here are a few examples of cube functions that are available to you.

- CUBEMEMBER returns a member from the data set, or cube.
- CUBEVALUE returns a particular aggregated value from the cube.
- CUBESETCOUNT returns the number of items in the cube.

Working with cube functions

To learn about cube functions, you can convert a PivotTable to functions and examine their syntax.

Exam Objective: MOS Excel Expert 3.4.4

1. Create a PivotTable and select a cell within it.
2. On the PivotTable Tools Analyze tab, click **OLAP Tools > Convert to Formulas**.

    All of the row and column headings and data values in the PivotTable are converted to formulas that use cube functions.

You can then copy and paste the formulas elsewhere and modify them to get the information you want. This method won't give you examples of all the possible cube functions, but it will give you a good start on understanding how they work.

Exercise: Converting a PivotTable to cube functions and examining cube function structure

`My Power Pivot` is open and the previous two exercises must be complete in order to perform this one. If your infrastructure demonstrated the previous ones, she or he might want to demonstrate this one as well.

Do This	How & Why

1. Convert the PivotTable to formulas.
 	
   a) Select a cell within the PivotTable.
 	
   b) On the PivotTable Tools Analyze tab, click **OLAP Tools > Convert to Formulas**.
	The PivotTable, as such, disappears along with the PivotTable tools.

2. Select and observe C4.

The Indus Tea column label. The CUBEMEMBER function is returning the member "Indus Tea" from the data set, or cube. The first argument, "ThisWorkbookDataModel", identifies the connection. The second, "[dItems].[Items].&[Indus Tea]", tells the function to return the Indus Tea value from the Items column in the dItems table. Knowing that you need to use the ampersand ("&") to concatenate the name of the column with the value would have been hard to guess.

```
=CUBEMEMBER("ThisWorkbookDataModel","[dItems].[Item].&[Indus Tea]")
```

3. Select and observe C5.

The first value under Indus Tea. The CUBEVALUE function returning the value for Sum of Revenue (the reference to $B$3) for Indus Team (the CUBEMEMBER function return in C$4) and Eurozone (the CUBEMEMBER function return in $B5). The function works because those cells refer to other cube function that return the data in its proper form. To enter a CUBEVALUE function directly, you would need to follow syntax more like what you saw in the CUBEMEMBER function in the previous step.

```
=CUBEVALUE("ThisWorkbookDataModel",$B$3,$B5,C$4)
```

⚠ **CAUTION:** This next step is tricky. The CUBEVALUE function will work only if you get it just right.
 	
4. In cell B13, enter a CUBEVALUE function to return the Tucana Roast sales for the sales rep named "Lloyd."
 	
   a) Select B13, begin to type `CUBEVALUE`, select the function from the AutoComplete list, and press **Tab**.
 	
   b) Begin to type `"This`, then select ThisWorkbookDataModel from the AutoComplete list.
 	
       Continued...
	
	To being to enter the cube function. The first argument you need is the connection name.

Do This	How & Why

c) Type and use AutoComplete to enter `","[Measures].[Sum of Revenue]"`.

So far, the function should look like this. "Measures" are calculations based on tables that are stored with the cube (such as the sum of the revenue values in the Invoices table).

=CUBEVALUE("ThisWorkbookDataModel","[Measures].[Sum of Revenue]"

d) Complete function by entering `,"[dReps].[Rep].[Lloyd]","[dItems].[Item].[Tucana Roast]")`.

The rest of the function looks like this. You can use AutoComplete to enter the table and column names, but will need to type the values ("Lloyd" and "Tucana Roast").

,"[dReps].[Rep].[Lloyd]","[dItems].[Item].[Tucana Roast]")

e) Press **Enter**.

If you got the function right, you will see the correct value, $4,516.11, in B13.

5. Save and close the workbook.

The PivotTable converted to formulas and a cube function

=CUBEVALUE("ThisWorkbookDataModel","[Measures].[Sum of Revenue]","[dReps].[Rep].[Lloyd]","[dItems].[Item].[Tucana Roast]")

	B	C	D	E	F	G	H	I	J	K
	Sum of Revenue	Column Labels								
	Row Labels	Indus Tea	Phoenix Roast	Tucana Roast	Vela Herbal	Grand Total				
	Eurozone	$9,794.52	$13,787.12	$13,372.20	$11,502.40	$48,456.24				
	International	$9,779.94	$10,800.80	$7,806.15	$10,152.80	$38,539.69				
	US	$6,480.00	$6,679.20	$6,417.63	$6,048.00	$25,624.83				
	Grand Total	$26,054.46	$31,267.12	$27,595.98	$27,703.20	$112,620.76				
	$4,516.11									

Assessment: The Power Pivot Data Model

The only text files that can open in Excel are ones that are structured with delimiters. True or false?

- True
- **False**

Which of the following are data sources for which you can create external connections? Choose all that apply.

- Microsoft Word
- **SQL server databases**
- **Microsoft queries**
- **Text files**
- **Microsoft Access**

All data you import into Excel is linked to its source. True or false?

- True
- **False**

Which of the following are advantages of using Power Pivot over using Excel along? Choose all that apply.

- **Can improve speed of calculation when using very large data sets.**
- Power Pivot can work with external data while Excel cannot.
- **Power Pivot can handle larger data sets than Excel can.**
- **Power Pivot provides access to the Data Model through cube functions.**

Module B: Exporting data

You can export data from Excel in a variety of formats that you can then use in many other programs.

You will learn:

- About using Save As to export data in various formats
- How to save worksheets as text files

Export formats

You can save Excel files in many different formats when you want to share your Excel data with users who don't have Excel, or when you want to use your data in a different program.

XML Data	Preserves structure and data. Can be read by many programs.
Web page	Creates a file that can be viewed in web browsers.
Text	Text files are the most portable, in the sense that most programs can read them. Excel can save a worksheet as a tab-delimited file, a comma-separated filed (CSV), or a structured text file (space delimited). You will lose some structural and formatting features when you save data as a text file.
PDF or XPS	Portable, viewable formats that do not allow interactions but can be viewed by anyone with a viewer.
Open Document Spreadsheet	An open-source document format.

Saving worksheets as text files

1. In Backstage view, click **Export**, then click **Change File Type**.

2. Click one of the file types shown, then click **Save As**.

   If you don't see the file type you want, click **Save as Another File Type**, then click **Save As**. You can select many formats from the File Type list in the Save window.

3. Enter a name for the file, and click **Save**.

   You might see a warning regarding some of the features of the workbook (such as multiple worksheets). Be sure you understand what you can and cannot save. It is best to save single worksheets, and just the data and basic structure.

Exercise: Saving a workbook as a CSV file

Do This	How & Why
1. Open `Invoices`.	The Excel file of that name in the `Importing and Exporting` data folder. The workbook contains a single worksheet with a list of invoices.
2. In Backstage view, click **Export**, then click **Change File Type**.	
3. Under Other File Types, click **CSV (Comma delimited)**, then click Save As.	There are many formats from which to choose. Comma-separated variable files are easily read by many other programs.
4. Name the file `My CSV Invoices`, then click **Save**.	When you attempt to save a workbook in a format that does not support some of its features, Excel warns you. Text files do not preserve formatting, so if the formatting is essential to you, you should consider another approach.
5. Click **Yes**.	To save the file in CSV format.
6. Close the workbook.	Excel prompts you to save changes, because the version you are seeing contains features (formatting) that were not saved in the CSV file.
7. Click **Don't Save**.	

Assessment: Exporting data

Which format should you save to if you want the greatest number of users to be able to see your formatted worksheet, regardless of the programs they have on their computer? Select the one best answer.

- Open Document Spreadsheet
- CSV (Comma delimited)
- **PDF**
- XML Data

Which format should you save to if you want users of database programs to be able to easily import the data and structure of your worksheet? Select the one best answer.

- XPS
- **CSV (Comma delimited)**
- XML

Summary: Importing and exporting

You should now know:

- About working with data from other sources in Excel, including how to import, structure, and format text files; how to get and manage external data connections; how to load the Power Pivot Excel add-in; and how to use Power Pivot and cube functions to access the Power Pivot Data Model

- How to export Excel workbooks to various formats, especially text formats such as CSV, and why you would use different export formats

Synthesis: Importing and exporting

You will need an active Internet connection and a web browser to complete this activity.

1. In a web browser, go to https:/finance.yahoo.com.

2. Look up a stock quote for a company in which you are interested. You can look up multiple companies by separating their symbols with a comma. Here are some sample companies.

   - Apple (AAPL)

   - Cisco Systems (CSCO)

   - Google (GOOGL)

   - British Petroleum (BP)

3. When the page with quotes appears, copy its URL from the address bar of your browser.

4. Close the web browser.

5. In Excel, open a new, blank workbook.

6. On the data tab, click **New Query > From Other Sources > From Web**.

   A window appears asking you for the URL of the web page you want to use.

7. Paste the copied URL into the box and click **OK**.

   After a few moments, the navigator window gives you choices of tables on the page.

8. Load the first table (with the stock quotes for your selected companies).

9. Wait a few moments, then click Refresh All. You should see some of the figures change because this is a live connection.

10. Save the workbook as `My Stocks`, then close it.

11. Close the Browser.

Microsoft stock quote information from Yahoo Finance

Symbol	Time & Price	Time & Price2	Chg & % Chg	Chg & % Chg2	Day's Low & High	Day's Low & High2
AAPL	11:06am EST	97.915	1.465	+1.52%	97.31	99.11
CSCO	11:05am EST	25.1936	-0.2164	-0.85%	25.14	25.63

Chapter 5: Analysis

You will learn:

- How to perform what-if analysis using scenarios and Goal Seek
- About the Analysis Toolpak and the tools it provides for analyzing data

Learning time: 50 minutes

Module A: What-if analysis

What-if analysis is just what it sounds like: you ask "what if" about your data by looking at the results of different assumptions. Any time you have a formula and change an input value, you're performing what-if analysis of a sort. But Excel provides very powerful tools for doing this, including the ability to save scenarios, a watch window for observing cell changes, and a feature called Goal Seek.

You will learn how to:

- Use a watch window to observe formula values change as you change precedent cells

- Use scenarios to observe how different sets of input values affect worksheet outcomes

- Use Goal Seek to find input values according to a desired formula outcome

Watch windows

A *watch window* allows you to watch a particular cell's formula and result as you make changes on a worksheet.
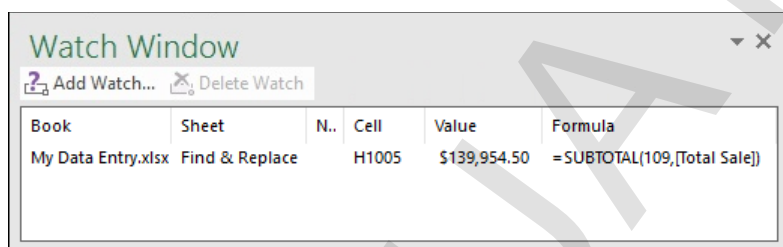
What's nice about a watch window is that it stays right in front of you, no matter where you scroll or what worksheet you make active. So, if you have formulas that are important to you as you make change, but can't see them from where you are, a watch window can solve that problem.

Adding a watch window

To use a watch window, you have to show it, then add cells to it.

Exam Objective: MOS Excel Expert 3.5.2

1. On the Formulas tab, click Watch Window.

   The Watch Window appears, but it doesn't show any cells to watch yet.

2. Select the cell you want to watch, then, in the watch window, click **Add Watch**.

   The Add Watch window appears, with the cell you selected showing. You don't have to select a cell before clicking the Add Watch button, but it's a bit of a shortcut.

3. Click **Add**.
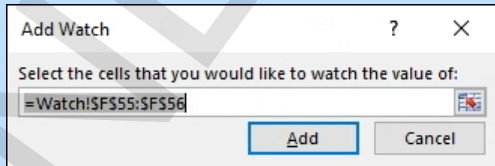
   The cell now appears as a row in the watch window. You can observe its value, formula, and other information.

You can move the watch window around to a convenient location, and it will stay in view as you move around the worksheet, or even to other worksheets and workbooks. You can also dock the watch window below the ribbon by dragging it to the top of the screen.

Exercise: Using a watch window to observe formula changes

Exam Objective: MOS Excel Expert 3.5.2

Do This	How & Why
1. Open `What If`.	From the current chapter's data folder. This workbook contains three worksheets. The first has sales projections for a number of Java Tucana customers, along with their discount percentages, and the discounted sales projections.
2. Observe the totals.	At the bottom of the data. There are totals formulas for the total discounted sales, and the total discounts.
3. Select F55:F56.	You'll add a watch window so that you can observe how making changes to various discount percentages affects the totals.
4. On the Formulas tab, click **Watch Window**.	The watch window appears, but there are no cells currently being watched.
5. Add watches for the selected cells.	
a) Click Add Watch.	To display the Add Watch window. The range you selected is highlighted.
b) Click **Add**.	The two cells appear as rows in the watch window. You can see their references, values, and formulas at a glance.
6. Dock the watch window.	Drag it to the top of the window, below the ribbon. This moves it out of the way of the worksheet.
7. Observe the values of cells F55 and F56.	They are $152,140 and $30,760, respectively.
8. Change the discount percentage for Award Sportswear to 40%.	Change the value in cell D10. You can see the new values for the watched cells in the watch window.
9. Close the watch window.	Click its close box.
10. Save the workbook as **My What If**.	

Using a docked watch window

Scenarios

Scenarios are different assumptions about the same data and formulas. In this example, the two budgets have the same line items, but one assumes no health club, cell phone, or car, while the other has all three.

Exam Objective: MOS Excel Expert 3.4.3

Scenario 1			Scenario 2	
No health club, cable, or car			*With health club, cable, and car*	
Bill	**Amt**		**Bill**	**Amt**
Mortgage	$ 1,200		Mortgage	$ 1,200
Health Club	$ -		Health Club	$ 99
Cable	$ -		Cable	$ 75
Car	$ -		Car	$ 320
Cell Phone	$ 240		Cell Phone	$ 240
Uitlities	$ 200		Uitlities	$ 200
Monthly	**$ 1,640**		**Monthly**	**$ 2,134**

You could, of course, copy and paste the budget all over a worksheet and use different values in each copy. But Excel's scenarios feature offers a much better approach. By using scenarios, you can perform what-if analysis by comparing results for many different sets of input data.

Saving scenarios

One of the best tools for what-if analysis is the Scenario Manager.

1. Select the input values that you want to save as a scenario.

For example, in a projected budget, you might want to highlight the expense or revenue figures.

2. On the Data tab, click **What-If Analysis > Scenario Manager**.

At first, there are no scenarios in the Scenario Manager.

3. Click **Add**, then enter a name for the scenario.

Note that a reference to the range you selected appears in the "Changing cells" box.

4. Click **OK**.

   To display the Scenario Values window. Here, you can set the values included for the scenario, or accept the values that were in the cells when you added the scenario.

5. Set values if you want, then click **OK**.

   To return to the Scenario Manager.

6. Add more scenarios.

   You can do this entirely from the Scenario Manager window by clicking **Add**, naming a new scenario, and entering values. But you can also return to the worksheet, change the values there, and then add a new scenario with those values.

7. When you're finished with the Scenario Manager, click **Close**.

Switching between scenarios

After you've created scenarios for a worksheet, you can switch between them in a couple of ways.

- Use the Scenario Manager.

   a) Display the Scenario Manager.

   b) Select the scenario you want, then click **Show**.

- Add the Scenario command to the Quick Access Toolbar.

   a) Click the Customize Quick Access Toolbar arrow, then click **More Commands**.

   b) In the All Commands category, click **Scenario**, then click **Add**.

   c) Close the Excel Options window.

   You'll notice that the command has no icon. But it does allow you to select scenarios from a drop-down list, which is very convenient if you work with scenarios often.

Exercise: Performing what-if analysis on a budget

My What If is open.

Exam Objective: MOS Excel Expert 3.4.3

Do This	How & Why
1. Activate the Budget worksheet.	It contains the actual budget figures for the first year of a café, with projections for the second year.
2. Observe the Year2 figures.	The figures assume revenue growth, with associated increases for payroll, supplies, and miscellaneous expenses, but decreases in upfront investment items like remodeling and equipment. The result is a modest profit. You'll create a scenario for these figures, which are your best guess for what will actually happen: a realistic scenario.
3. Create a scenario for the expense lines and the revenue line, called `Realistic`.	
a) Select both the expense lines and the revenue line, as shown.	Select C8:C15, then, while holding **Ctrl**, click C17. Because total expenses is calculated by a formula, you wouldn't include it in the selection for a scenario. Select only values for a scenario.

	B	C
7	**Year1**	**Year2-Proj**
8	24000	24000
9	12456	2000
10	5200	500
11	9873	1000
12	12395	13200
13	4096	1000
14	62348	65000
15	9120	11000
16	$139,488	$117,700
17	$ 95,000	$120,000
18	$ (44,488)	$ 2,300

b) On the Data tab, click **What-If Analysis > Scenario Manager**.	To display the Scenario Manager window.
c) Click **Add**, then name the scenario `Realistic`.	Note that the range you selected shows in the "Changing cells" box. You could also add a comment.

Scenario name:
Realistic

Changing cells:
C8:C15,C17

d) Click **OK**, then click **OK** again.	To accept the scenario values and return to the Scenario Manager.
e) Click **Close**.	

Do This	How & Why
4. Create a pessimistic scenario. a) Change the values as shown.	The cells you need to change are highlighted here. In the pessimistic scenario, there is less revenue and a bit more supplies expense, so we cut back on the remodeling and equipment lines. The result is a slightly unprofitable scenario.

	A	B	C
7	**Expense**	**Year1**	**Year2-Proj**
8	Rent	24000	24000
9	Remodeling	12456	1000
10	Legal	5200	500
11	Equipment	9873	500
12	Supplies	12395	14000
13	Advertising	4096	1000
14	Payroll	62348	65000
15	Miscellaneous	9120	11000
16	**Total Expenses**	$139,488	$117,000
17	**Revenue**	$ 95,000	$110,000
18	**Profit**	$ (44,488)	$ (7,000)

Do This	How & Why
b) Display the Scenario Manager, and click **Add**.	Notice that the same range appears in the "Changing cells" box, as before. After you've created one scenario on a worksheet, you don't have to select the ranges to create a second.
c) Name the scenario **Pessimistic**.	
5. Create and name an optimistic scenario according to the figures shown.	Again, the cells you need to change are highlighted. In this scenario, there is more revenue. We restore some of the expense lines and actually increase payroll. The result is much more profit than in the "realistic" scenario.
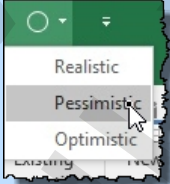
	A	B	C
7	**Expense**	**Year1**	**Year2-Proj**
8	Rent	24000	24000
9	Remodeling	12456	2000
10	Legal	5200	1000
11	Equipment	9873	1000
12	Supplies	12395	14000
13	Advertising	4096	2000
14	Payroll	62348	70000
15	Miscellaneous	9120	11000
16	**Total Expenses**	$139,488	$125,000
17	**Revenue**	$ 95,000	$145,000
18	**Profit**	$ (44,488)	$ 20,000

Do This	How & Why
6. Save the workbook.	
7. Use Scenario Manager to switch among the scenarios.	Display the Scenario Manager, click the scenario you want, then click **Show**. Close the Scenario Manager when you're done.
Continued...	

Do This	How & Why
8. Add the Scenario command to the Quick Access toolbar.	
a) Click the Customize Quick Access toolbar arrow, then click **More Commands**.	
b) In the All Commands category, select **Scenario**, then click **Add**.	To quickly find the command, click in the list, then press **S** to move to commands that begin with that letter.
c) Click **OK**.	To close the Excel Options window. The button appears on the toolbar, although it does not have an icon. It does have a drop-down arrow, and you can use this to quickly select a scenario for the active worksheet.
9. Use the Scenario list to switch among scenarios.	If you will be using scenarios often, it's a good idea to add this button somewhere.

Seeking a goal

Another type of what-if analysis involves knowing an outcome—for example, an amount of profit you want to have—and trying to figure out the input or inputs you need to reach it. In this type of analysis, you'll know the value you want from a formula, but not the referred-to values. Excel provides a couple of utilities for seeking input values.

- *Goal Seek*, which allows you to look for a single value based on a desired outcome.

- *Solver*, which is an add-in utility that lets you solve more complicated scenarios for more than one variable, and with constraints. To use Solver, you need to add its add-in to the program. You can manage Add-ins in the Excel Options window. Click **Add-ins**, then select **Excel Add-ins** from the Manage list, and finally click **Go**.

Using Goal Seek

To use Goal Seek, you have to set a cell with a formula to a particular value, and tell Goal Seek which cell you want to change.

Exam Objective: MOS Excel Expert 3.4.3

1. On the Data tab, click **What-If Analysis > Goal Seek**.

    To display the Goal Seek window.

2. Set the cells and values.

The *Set cell* contains the formula for which you want to set a value. Here, for example, we're solving for a particular profit percentage.

To value is the value to which you want to set the cell. In this case, 4%.

By changing cell is the cell for which you want to solve a value. This example would find a revenue figure (C9) that would result in 4% profit.

3. Click **OK**.
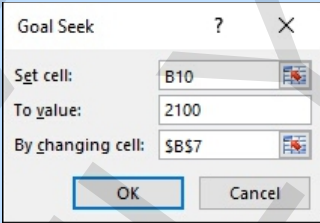
    A Goal Seek Status window tells you the results. Click **OK** to close it.

Exercise: Using Goal Seek to find an input based on a total budget outcome

My What If is open.

Exam Objective: MOS Excel Expert 3.4.4

Do This	How & Why
1. Activate the Goal Seek worksheet.	It contains a simple personal budget. There is currently no car payment. You'll use Goal Seek to figure out how much of a car payment can be afforded based on a fixed total budget amount.
2. Select cell B10.	You will set the total budget to a particular value in the Goal Seek utility.
3. On the Data tab, click **What-If Analysis > Goal Seek**.	To display the Goal Seek window.
4. In the To value box, type 2100.	To specify 2100 as a total budget amount.
5. Click in the "by changing cell" box, then click B7.	The window should look like this. You're saying that you want to know how much of a car payment you can afford if your total monthly budget is $2,100.
	Goal Seek ? ✕ Set cell: B10 To value: 2100 By changing cell: $B$7 OK Cancel
6. Click **OK**.	The Goal Seek Status window tells you that a solution has been found.
7. Click **OK** to close the window.	The car payment can be as much as $290 to stay within the monthly budget.
8. Save and close the workbook.	

The completed personal budget with car payment goal

	A	B
1	**Personal Budget**	
2		
3	**Bill**	**Amt**
4	Mortgage	$ 1,200
5	Health Club	$ 95
6	Cable	$ 75
7	Car	$ 290
8	Cell Phone	$ 240
9	Uitlities	$ 200
10	**Monthly**	**$ 2,100**

Assessment: What-if analysis

Which of the following is NOT cell information that is shown for a watched cell in the watch window? Choose the one correct answer.

- Workbook
- Sheet
- **Number format**
- Value
- Formula

To add a scenario, you must select the input range before opening the Scenario Manager. True or false?

- True
- **False**

Which method would you use to figure out how many of an item to order if you have a budget and know the unit cost? Choose the best answer.

- Scenarios
- **Goal Seek**
- Solver

Module B: The Analysis Toolpak

The Analysis Toolpak is an Excel add-in that provides tools that walk you through various kinds of statistical analysis. By default, the Toolpak is not enabled, so you need to do that before you can use its tools.

You will learn:

- About the Analysis Toolpak, and how to load its add-in

- How to use the Analysis Toolpak to calculate the Pearson correlation coefficient for two-variable data

- How to use the Analysis Toolpak to create a histogram for a set of data

About the Analysis Toolpak

You can use Excel's statistical functions to perform all sorts of statistical analyses on various kinds of data. But if you do a lot of statistical analysis, you should examine the tools Excel provides through the *Analysis Toolpak add-in*.

The Analysis Toolpak provides tools to walk you through various types of statistical analysis. Instead of using functions, you select a type of analysis, specify input ranges for parts of the calculation, and Excel does the rest. Here are some of the types of analysis you can perform using the Analysis Toolpak. This list is by no means complete.

Correlation	The correlation coefficient, a value between -1 and +1, shows how closely two variables are related.
Covariance	Helps to determine the relationship between two sets of data.
Exponential Smoothing	Smooths out peaks and valleys in data so you can recognize trends.
Histogram	A chart that shows how many data points fall into each of several categories, called *bins*, that you define.
Moving Average	Smooths out irregularities by calculating the average at each point in a data set.
Regression	A process for estimating the relationship between a dependent variable and one or more independent variables.

Loading the Analysis Toolpak add-in

To access the Analysis Toolpak, you need to load its add-in.

1. In Backstage view, click **Options**.

2. Click **Add-ins**.

3. In the Manage list, click **Excel Add-ins**, then click **Go**.

    To display the Add-Ins window.
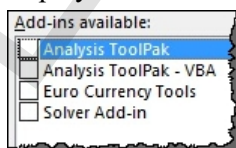
4. Check **Analysis ToolPak**, then click **OK**.

After loading the Toolpak add-in, there will be an Analysis group on the Data tab, with a Data Analysis button.
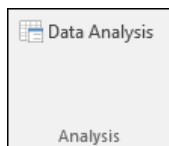
Analyzing correlation

The Correlation tool in the Analysis Toolpak calculates the Pearson correlation coefficient for two sets of data. A correlation coefficient of -1 means the data are strongly correlated in a negative way (as one set goes up, the other goes down); 0 means the data are not correlated at all, and +1 means the data are strongly positively correlated.

1. Set up two sets of data in contiguous columns on a worksheet.

   Having labels at the top will make the correlation output easier to read.

2. On the Data tab, in the Analysis group, click **Data Analysis**.

   To display the Data Analysis window. Here, you have access to many tools for various kinds of data analysis.
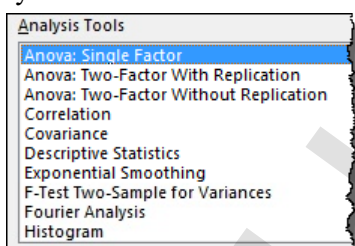
3. Click **Correlation**, then click **OK**.

   To display the Correlation window. Each data analysis tool has a window like this. You enter input ranges, select options, and then Excel takes care of calculating, structuring, and formatting the result.
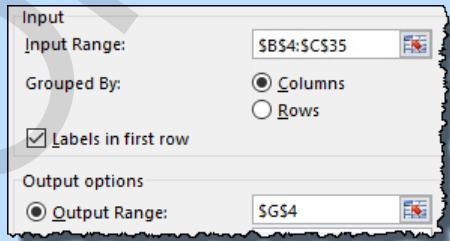
4. Specify the input and options you want.

   • *Input Range* should contain the data sets you want to analyze. Including row labels will make the output more readable.

   • Click **Labels in first row** if you included labels in the input range.

   • In the "Output options" area, select where you want the output to appear.

5. Click **OK**.

   Excel creates a small table that shows the correlation coefficients between all pairs of data sets in the input range.

Exercise: Analyzing correlation between temperature and lemonade sales

Do This	How & Why
1. Open `Analysis Tools`.	From the `Analysis` data folder. This worksheet contains daily sales for lemonade in July, 2015. It also contains high-temperature data. You'd like to know how closely sales correlate to temperature, so you can know whether it's reasonable to predict how much lemonade you need, depending on weather forecasts.
2. Load the Analysis Toolpak add-in.	
a) In Backstage view, click **Options**, then click **Add-Ins**.	
b) In the Manage list, click **Excel Add-ins**, then click **Go**.	To display the Add-Ins window. Note that if you have the Developer tab showing on the ribbon, you can simply click the **Add-Ins** button to display this window.
c) Click **Analysis Toolpak**, then click **OK**.	
3. On the Data tab, click **Data Analysis**.	To display the Data Analysis window, which gives you access to all of the Analysis Toolpak features.
4. Click **Correlation**, then click **OK**.	To display the Correlation window.
5. Click in the "Input range" box, then select B4:C35.	The input range contains the row of labels, as well as all the temperature and sales data.
6. Click **Labels in first row**.	
7. Under "Output options," click **Output Range**, click in the Output Range box, and then select G4.	To specify that you want the correlation information on this same worksheet. The window should look like this.

8. Click **OK**.	Excel calculates correlation coefficients between each possible pairing of the sets, then places them in a formatted table, using the row labels. The correlation coefficient between temperature and sales is .96887, which is a very high positive correlation. This means that as temperatures go up, you would expect sales to go up.

	Temp (C°)	Sales
Temp (C°)	1	
Sales	0.96887	1

Do This	How & Why
9. Save the workbook as `My Analysis Tools`.	

Creating histograms

A histogram is a very common chart that shows the frequency of data values in a set by grouping them into bands called *bins*. For example, you could use a histogram to show how many invoices fall into bins for every $25 range.
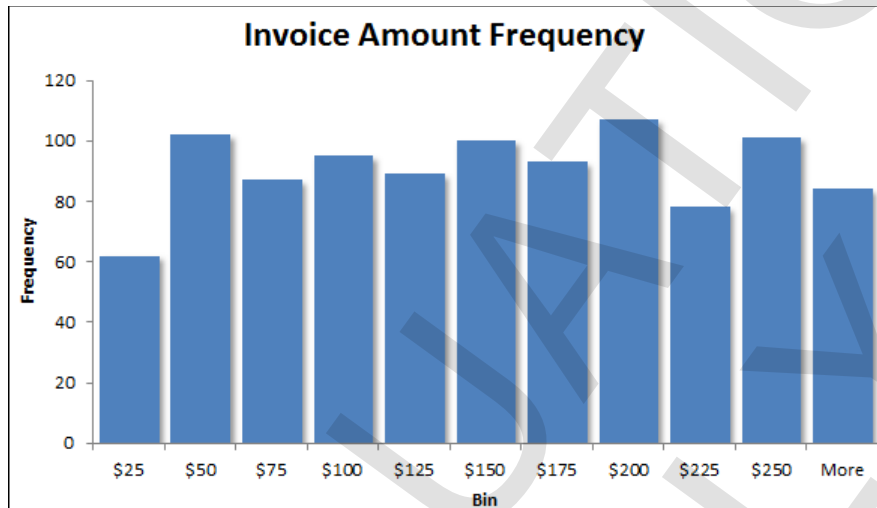
1. Set up your data and your bins.

- The input should be a single set of data. In the preceding example, this would be a column of invoice amounts.

- The bins should be the bands you want to use for the histogram bins. In this example, the bins range looks like this.

2. Click **Data Analysis**, then select **Histogram** and click **OK**.

To display the Histogram window.

3. Specify the input and bin ranges for the histogram.
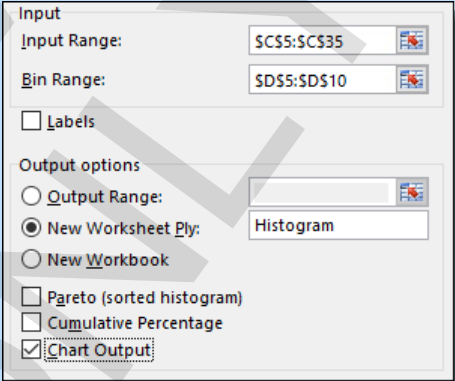
4. Specify where you want the output.

5. To create a chart, click **Chart Output**.

6. Click **OK**.

Exercise: Creating a histogram for lemonade sales

`My Analysis Tools` is open, and the Analysis Toolpak add-in has been loaded.

Do This	How & Why
1. Observe the Bins column.	Bins are groups for values in a histogram. When you create a bin range, be sure to sort the values in ascending order.
2. Click **Data Analysis**, click **Histogram**, and click **OK**.	To display the Histogram window. It works much like the Correlation window, but the specific information it asks for relates to creating histograms.
3. Click in the Input Range box, then select C5:C35.	To specify the daily sales amounts as the input range.
4. Click in the Bin Range box, then select D5:D10.	To specify the bins for the histogram. Note that the values in the bins range are sorted in ascending order.
5. Under "Output options," click **New Worksheet Ply**, then type `Histogram` in the box.	
6. Click **Chart Output**.	To specify that you want a histogram chart, not just a table. The window should look like this.
7. Click **OK**.	To create the histogram on a new worksheet. It clearly shows that daily sales most often are in the bin between $150 and $200. You could now format the chart as you would any other Excel chart.
8. Save and close the workbook.	

The daily lemonade sales histogram

Bin	Frequency
$ 100	1
$ 150	5
$ 200	12
$ 250	3
$ 300	6
$ 350	4
More	0
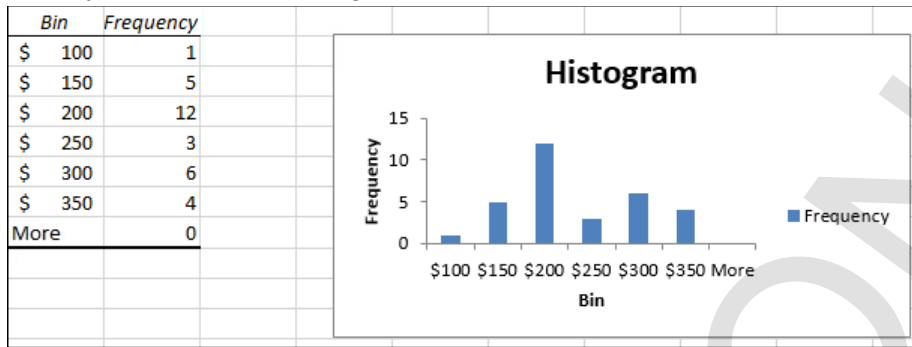
Assessment: The Analysis Toolpak

The Analysis Toolpak is available immediately with a default installation of Excel. True or false?

- True
- **False**

Which of the following statements is most accurate in regards to correlation? Choose the single best answer.

- Highly correlated data shows a cause-and-effect relationship.
- **A correlation coefficient of close to +1 shows a high correlation.**
- You can show correlation only for two data sets.

The Histogram tool will set up your bins for you. True or false?

- True
- **False**

Summary: Analysis

You should now know:

- How to perform what-if analysis by using a watch window, how to use scenarios to save and switch between different sets of input values for a worksheet, and how to use Goal Seek to solve for an input value when you have a desired outcome for a formula

- About the Analysis Toolpak and the tools it provides for analyzing data, how to calculate correlation for two sets of data, and how to create a histogram to show frequency of grouped data values, or bins

Synthesis: Analysis

In this synthesis exercise, you'll start by looking at a sales representative compensation-plan calculator. You'll create scenarios for high and low commission rates, then you'll use Goal Seek to figure out how much sales a rep would need to have to make the high commission plan a better deal.

1. Open `Analysis Synthesis` from the `Analysis` data folder. The first worksheet contains a sales compensation-plan calculator that takes into consideration salary, commission rate, and sales.

2. Create two scenarios for the comp plan by changing cells B4 and B5 (the salary and the commission rate). Call the original values scenario `High Comm`. Then, change the values as shown, and call this scenario `High Salary`.

Salary	$ 45,000
Comm%	4%
Sales	$ 200,000
Total Comp.	$ 53,000

3. Which is the better scenario for the sales rep if she has $200,000 in sales?

4. Which is better if she sells $300,000 in sales?

5. Using the High Comm scenario, use Goal Seek to determine how much sales the rep must achieve to make $75,000 in total compensation.

6. Save the workbook as `My Analysis Synthesis`.

7. Activate the **FEGuide** worksheet, which contains US Government fuel economy data for 2016 model cars.

8. Create a correlation matrix that shows correlation coefficients between sets of engine displacement (Eng Displ), number of cylinder (# Cyl), number of gears (# Gears), and combined mileage. Use the labels in the first row, and place the matrix on the same worksheet, beginning in cell L1.

9. Is the correlation strong between mileage and engine displacement? Is it negative or positive?

10. Which data sets have the strongest positive correlation?

11. Create mileage bins in column J starting at 15, and stepping by 5, up to 40. (Hint: enter the first two values, then use the Auto Fill handle.)

12. On a new sheet, create a histogram showing the distribution of the combined mileage figures by bin for all models.

13. What is the most frequently occurring mileage bin?

14. Save the workbook as `My Analysis Synthesis` and close the workbook.

Chapter 6: Macros and Forms

You will learn:

- About using macros to automate repetitive tasks, and how to record and run a macro
- Various ways to run a macro
- About using forms to gather input from users, and how to create simple forms

Learning time: 80 minutes

Module A: Recording macros

Macros are small programs that can perform tasks for you in Excel, such as formatting or entering data. Fortunately, you don't need to be a programmer to create macros. Excel has a macro recorder that you can use to capture your actions so that you can play them back in the future as a macro.

You will learn:

- What a macro is and how to run one

- How to record a simple formatting macro

- How to examine and edit code in a recorded macro

- How to control the recording of relative and absolute references in a macro

About macros

Macros are short programs that you use to perform repetitive tasks in Excel, such as formatting a report, entering formulas in a column, or rearranging data. Excel macros are written in a language called *Visual Basic for Applications (VBA)*, and if you're familiar with programming, you might want to write code for macros yourself.

But Excel provides a great macro recording feature that can translate your actions into VBA code, so that you can play them back in the future without your having to know much or anything about programming.

To work with macros, you should display the Developer tab on the ribbon, so you can use the Code-group commands.
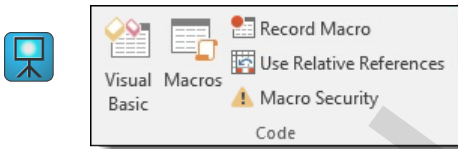
Security concerns

Macros can contain harmful programs called *viruses* that can hurt your computer or your data. For this reason, you should pay careful attention to where your macros come from. You should also use Excel's Trust Center to be sure that you always know when a workbook contains macros.

Exam Objective: MOS Excel Expert 1.1.5

The first line of security is in file types. Ordinary Excel 2016 workbooks have an .xlsx extension, and templates have a .xltx extension. Neither of these file types can contain macros. You can save macros only in macro-enabled workbooks (.xlsm) and macro-enabled templates. This way, you know which files might contain macros before you open them. The .xls and .xlt files used by Excel 2003 and earlier can also contain macros. You shouldn't usually save your own workbooks in these formats without good reason, but you should be aware they might pose a security risk when you receive them from someone else.

Even if you don't have file extensions displayed on your computer, you can still see the file types in the Open window or Windows Explorer. Macro-enabled workbooks and templates also have a different icon with an exclamation point.

The second line of defense is Excel's security settings. By default, even when the file type allows macros in a document, Excel will run them only if the workbook is in a trusted folder, or if you've designated it as a trusted workbook. Otherwise, Excel displays a security warning.
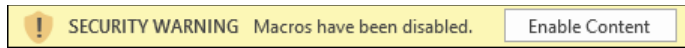
Managing macro security

You should always keep security in mind when opening macro-enabled files. By default, when you open a macro-enabled workbook that isn't a trusted file or in a trusted location, you'll see a security warning, and no macros will run. You can bypass this to enable macros in the workbook, but you should never enable macros on a workbook you don't trust.

- To trust an individual workbook, click **Enable Content** in the Security Warning bar.

- For some files, such as those on network drives, you'll need to click **Yes** in a separate Security Warning window.

- To make broader changes, click **Macro Security** in the Developer Tab's Code group to open the Trust Center window.

  - Click **Macro Settings** to change basic security settings for running macros. A good general setting is **Disable all macros with notification**.
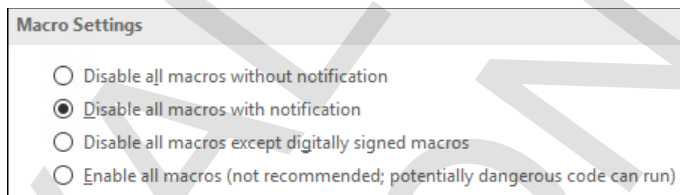
  - Click **Trusted Locations** to add or change trusted folders for opening files. This can be dangerous, so don't add them lightly.

  - Click **Trusted Documents** to change how Excel handles trusted documents, or to clear the current list of trusted documents.

Running a macro

There are many ways to run a macro.

- Click an assigned macro button on the ribbon or Quick Access toolbar.

- Press an assigned keyboard shortcut.

- Click **Macros** in the Code group to open the Macros window. Then select the macro you want, and click **Run**.

Exercise: Running a sales report macro

Exam Objective: MOS Excel Expert 1.1.5

Do This	How & Why
1. Open `Macros Demo`.	From the `Macros and Forms` data folder. You will probably see a security warning, telling you that macros are disabled in the workbook.
2. Click **Enable Content**.	To enable the macros the workbook contains. You should enable content only when you know and trust the source of a workbook. The workbook contains three worksheets, each with a sales report for a different month. The January report has commission calculations, formatted headings, and values formatted as currency.
3. If necessary, display the Developer tab on the ribbon.	Right-click the ribbon, click **Customize the Ribbon**, click **Developer**, then click **OK**.
4. On the Developer tab, in the Code group, click **Macro Security**.	To display the Macro Security settings in the Trust Center. Here, you can control how restrictive you want to be about enabling macros. The default option, **Disable all macros with notification**, is generally a good idea.
5. Click **Cancel**.	
6. Activate the **February** worksheet.	This worksheet represents how the data comes in to the workbook every month. Columns don't quite fit, there is no formatting, and there is no commissions column.
7. Run the MonthlySalesReport macro.	
a) On the Developer tab, click **Macros**.	To display the Macros window. The only macro in this workbook, MonthlySalesReport, is selected.
b) Click **Run**.	To run the macro, which adds the commissions column, then formats the headings, fits the columns, and formats the numbers. You can see how using macros for things you do on a regular basis could save a great deal of time.
8. Activate the **March** tab.	Notice that the monthly reports don't all have the same number of rows. March has nine, while February had seven, and January had eleven. How can the macro know how many rows to operate on?
9. Run the MonthlySalesReport macro on the March worksheet.	The macro works on this data as well.
10. Save the workbook as **My Macros Demo**, then close it.	

Sales data formatted by a macro

Row Labels	Indus Tea	Phoenix Roast	Tucana Roast	Vela Herbal	Total	Comm.
Blackwell		$259.20		$259.20	$ 518.40	$20.74
Daniels	$226.80	$129.60	$205.20		$ 561.60	$22.46
Hernandez	$275.40	$226.80	$673.20	$671.40	$1,846.80	$73.87
Lloyd	$ 48.60	$797.40	$306.00	$592.20	$1,744.20	$69.77
McCanney	$ 51.30		$792.00	$763.20	$1,606.50	$64.26
Sanchez	$149.40	$297.00	$503.10		$ 949.50	$37.98
Schiller	$102.60	$153.90	$595.80	$513.00	$1,365.30	$54.61
Westlein	$291.60	$541.80	$980.10	$145.80	$1,959.30	$78.37

The macro recorder

Although macros are, technically, programs written in VBA, you don't need to be a programmer to create a macro. The macro recorder lets you record what you're doing—selecting cells, entering formulas, formatting ranges—translated into VBA code. There are certainly things VBA programs can do that you can't record, but so many useful macros can be completely recorded without ever having to look at the code.

To get the most out of recording macros, you'll want to think through what you want to accomplish, and practice the actions a few times before you turn on the recorder.

1. The *macro name*, which may include letters, numbers, and underscores, but not spaces.

2. The *shortcut key*, which you can use to quickly run the macro. All shortcut keys include holding down **Ctrl**, and you can add other command keys (**Shift** or **Alt**, for example), by holding them down while you press a letter.

3. *Store macro in* gives you three choices for where to store the recorded macro code.

   - **This Workbook**, which means the macro will be stored with the current workbook.

   - **Personal Macro Workbook**, a hidden workbook that is always open when you a working with Excel. If you want to use the macro all the time when you work in Excel, this is the place to store it.

   - **New Workbook**, which will store the macro in a new workbook, which you will then need to save.

4. You can add a *description*, so you will know what the macro's purpose is later.

Recording a macro

When you've practiced the actions you want to include in your macro, you're ready to use the recorder.

Exam Objective: MOS Excel Expert 2.3.4

1. If you want the macro's actions to begin with the active cell, whatever that is, then select the cell where you'll begin before turning on the recorder.

   If you want the macro to select a particular cell or range as its first action, it doesn't matter where the active cell is when you start recording.

2. On the Developer tab, in the Code group, click **Record Macro**.

   The Record Macro window appears, allowing you to enter information about the macro.

3. Click **OK** to begin recording.

4. Record the actions you want to include in the macro.

5. Click **Stop Recording** (in the Code group) when you are finished.

   You can instead click the Stop Recording button on Status bar.

Saving a macro-enabled workbook

When a workbook contains macros, you have to save it in Macro-Enabled Workbook format. If you attempt to save a workbook with a macro in the regular workbook format, you'll see a warning stating that the macros will not be saved if you continue.

1. Display the Save As window.

2. In the "Save as type" list, click **Excel Macro-Enabled Workbook**.

3. Enter a name, and select a location for the workbook.

4. Click **Save**.

Exercise: Recording a macro to format headings

Exam Objective: MOS Excel Expert 2.3.4

Do This	How & Why
1. Open `Macros Data`.	From the `Macros and Forms` data folder. This workbook contains six worksheets, each with a month's sales data. You'll record a macro to format the headings to be bold, centered, underlined, and wrapped.
2. On the January worksheet, format the headings.	Try to perform the actions with as little extra movement of the mouse as possible, and without moving the active cell except when necessary. The idea here is to practice the actions before recording the macro.
a) Select A1:F1.	
b) Click **B** .	

Do This	How & Why
c) Click ⊡ .	
d) Click ▤ .	
e) Click **Wrap Text**.	The headings should look like this. It is important to perform actions efficiently when you record a macro, so practice first.

	A	B	C	D	E	F
1	Rep	Indus Tea	Phoenix Roast	Tucana Roast	Vela Herbal	Total
2	Blackwell			48.6		48.6
3	Daniels	451.8	275.4	34.2	470.7	1232.1

Do This	How & Why
3. Activate the **February** worksheet.	You'll record the macro while formatting the headings here.
4. On the Developer tab, click **Record Macro**.	To display the Record Macro window.
5. Enter the information as shown.	Macro names cannot have spaces. Also, to get the shift in the keyboard shortcut, click in the box then hold down **Shift** while pressing **F**.

Macro name:
FormatHeadings

Shortcut key:
Ctrl+Shift+ F

Store macro in:
This Workbook

Description:
Format monthly report headings

Do This	How & Why
6. Click **OK**.	To begin recording. Notice that the Record Macro button in the Code group is now the Stop Recording button. Click it when you're finished recording.
7. Record the actions from Step 2 to format the headings.	
8. Click **Stop Recording**.	
9. Attempt to save the workbook as My Macros Data.	Excel displays a warning, because you cannot save macros in a standard-format workbook.
10. Click **No**.	To return to the Save As window.
11. In the "Save as type" list, click **Excel Macro-Enabled Workbook**, then click **Save**.	
Continued...	

Do This	How & Why
12. Use the macro to format the headings on the March worksheet. a) Activate the **March** worksheet. b) Press **Ctrl**+**Shift**+**F**.	To run the macro by using its shortcut key.
13. Save the workbook.	

VBA code

Excel macros are actually programs written in Visual Basic for Application (VBA). VBA code lives in *modules*, and a particular macro is stored in a *procedure* within a module.

```
Sub FormatHeadings()                    1
'
' FormatHeadings Macro
' Format monthly report headings        2
'
' Keyboard Shortcut: Ctrl+Shift+F
'
    Range("A1:F1").Select
    Selection.Font.Bold = True
    With Selection.Borders(xlEdgeBottom)
        .LineStyle = xlContinuous
        .Weight = xlThin                 3
    End With
    With Selection
        .HorizontalAlignment = xlCenter
        .WrapText = True
    End With
End Sub
```
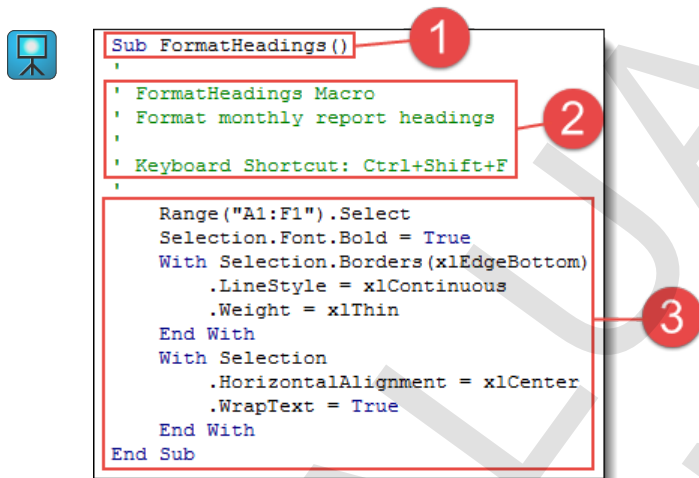
1. The *procedure name* identifies the procedure (or macro name).

2. *Comments* explain what the code does but have no effect when the program runs.

3. *Statements* are the individual programming commands in the procedure (or macro).

Editing macros

You can create macros by recording or by writing code directly. Either way, if you want to edit or step into a macro, you will do that in the Visual Basic for Applications program, which is an editing environment for VBA code.
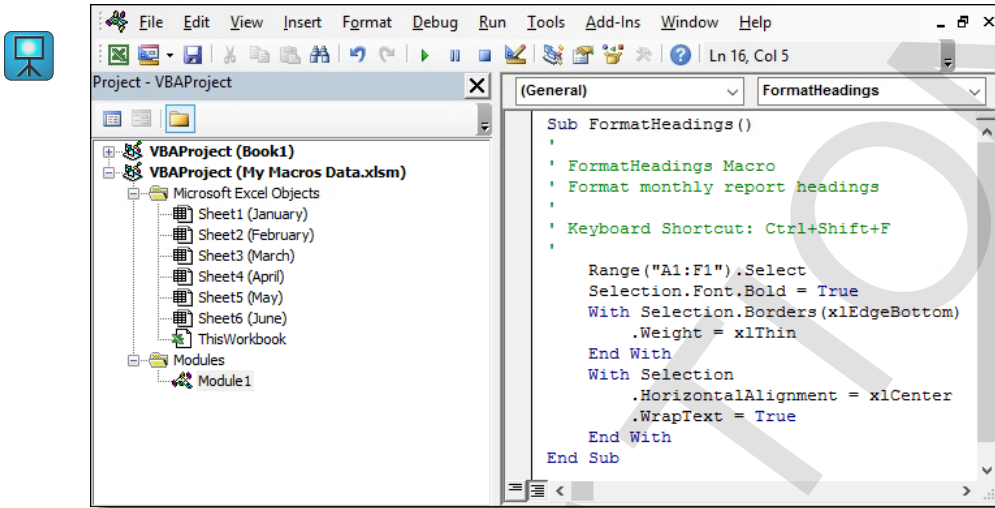
Programming in VBA is beyond the scope of this course, but if you find yourself using macros regularly, you might want to learn the basics. With just a little coding knowledge, you can make simple tweaks without having to re-record the macro.

1. In the Macros window, select the macro you want to edit, then click **Edit**.

   The Microsoft Visual Basic for Applications program opens, with the module containing the macro in a window on the right.
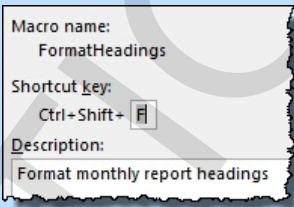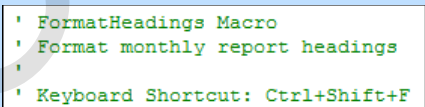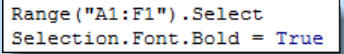
2. Edit the macro.

   - Each macro within a module begins with a `Sub` statement containing the macro name, and ends with an `End Sub` statement.

   - Each statement in the macro represents a step. The purpose of some is easy to guess just by reading them, but others will mean little to VBA novices.

   - VBA code is very syntax-sensitive. Improper spacing, case, or punctuation can change a macro's functions or cause it not to work at all.

3. Click **File > Close and Return to Microsoft Excel**, or press **Alt**+**Q**.

Exercise: Viewing and modifying a macro's code

`My Macros Data` is open.

Do This	How & Why
1. Display the Macros window.	From here, you can run, step into (run the macro a line at a time), edit, delete, or change options for a macro.
2. Click **FormatHeadings**, then click **Options**.	To display the Macro Options window. If all you want to do is change a macro's shortcut key or description, this is the place to do it. Macro name: FormatHeadings Shortcut key: Ctrl+Shift+ F Description: Format monthly report headings
3. Click **Cancel**, then click **Edit**.	To open the macro's module in the Microsoft Visual Basic for Applications editor. The Project pane in the upper left has a tree for each open workbook. You might be surprised that there are more open than you were aware of. Excel add-ins work through hidden workbooks, and those are what you're seeing.
4. Observe the macro's VBA code.	You will likely see slightly different code than what others see, and more code than is probably necessary to accomplish the macro's goals. This is one of the drawbacks of the recorder: Excel is likely to record more code than you need.
5. Observe the first line of the module.	The `Sub` statement defines the name of the macro. `Sub FormatHeadings()`

6. Observe the next three lines.	These are comments, which are preceded by an apostrophe and have no effect on the code itself. They appear in green in the editor.
	
	 `' FormatHeadings Macro`
	 `' Format monthly report headings`
	 `'`
	 `' Keyboard Shortcut: Ctrl+Shift+F`

7. Observe the first two statements of the procedure.	The first selects a range, while the second sets the bold property for the selection to "TRUE." That is, these statements select the headings and make them bold.
	
	 `Range("A1:F1").Select`
	 `Selection.Font.Bold = True`

8. Add a statement to also make the headings italic.	

Do This	How & Why
| a) Copy the bold statement, then paste another copy of it right after it. Like this. | ```
Range("A1:F1").Select
Selection.Font.Bold = True
Selection.Font.Bold = True
With Selection.Borders(xlEdgeBo
      .LineStyle = xlContinuous
``` |
| b) Edit the second bold statement as shown. | ```
Range("A1:F1").Select
Selection.Font.Bold = True
Selection.Font.Italic = True
With Selection.Borders(xlEdgeBottom)
``` |
9. Click **File > Close and Return to Microsoft Excel**.	
10. Activate the **April** worksheet and run the macro.	Press **Ctrl**+**Shift**+**F**. Now, the macro applies italics as well.
11. Save the workbook.	

References in recorded macros

Just as you need to pay attention to absolute and relative references in formulas, the same is true when recording macros. By default, Excel will record references absolutely, meaning that if you select cell A1, the macro will select A1, no matter where the active cell is when it runs that statement.

But often, you want a macro to move over one cell, or select to the beginning of a row, or to the end of a column, and so on. There are a number of techniques for recording references and range selection the way you want.

Controlling references in macros

There are many techniques you can use to control how references are recorded in a macro, and to have a macro select the ranges you want it to. You should carefully plan how you record your macros before you record, particularly in regard to selecting ranges.

- To omit cell selection from the macro, select a starting cell before recording.
- To record selections relative to the active cell, click **Use Relative References** in the Developer tab's Code group.
- To revert to recording absolute selections, click **Use Relative References** again. You can toggle this feature as often as necessary during recording.
- Use **Ctrl**, **Shift**, and the arrow keys to move to the end of a region or **Ctrl** plus the arrow keys to select to the end of a region (**Ctrl**+**Shift** plus arrow keys). Excel records these movements as a `Range` statement that uses an `End` method to do what you intend.
- Use **Ctrl**+**Shift**+* to select the current region around the active cell (the area enclosed by blank rows and columns). Very useful!

Exercise: Recording a macro to enter a formula in a variable range

`My Macros Data` is open.

> ⚠️ **CAUTION:** This exercise is tricky. Follow the instructions closely to make it work right. Your instructor might choose to demonstrate it instead.

Do This	How & Why
1. Compare the data on the May and June worksheets.	There are four reps in the May data, but nine in June. Every month, this can be different. We'd like to record a macro that will calculate a 4% commission for each rep in column G, but we'll need to use some tricks to get the macro to do what we want when used on data of different lengths.
2. On the May worksheet, select any cell.	The first thing we want the macro to do is enter a label in cell G1. Because we always want that specific cell, we'll select it as part of the macro, not before recording.
3. Begin to record a macro using the following name, shortcut key, and description.	Click **Record Macro**, enter the information, then click **OK**.
4. Select G1, type `Comm`, then press **Ctrl**+**Enter**.	To enter the label without moving the active cell.
5. Select G2, type or enter `=F2*4%`, then press **Ctrl**+**Enter**.	To enter the commission formula in cell G2 specifically. Now, how will we copy that formula down but *not* into extra cells? The trick is to use relative references, the totals column, and the select end features.
6. Press **Ctrl**+**C**.	To copy the formula.
7. With G2 active, click **Use Relative References**, then press the left arrow key.	To move to the cell one cell to the left, which is how the recorder will record the action.
8. Press **Ctrl** and the down arrow key.	To move to the end of the Totals column.
9. Press the right arrow key.	To move back to the Comm column. These last three steps were all to find the last cell in the Comm column without knowing exactly where it was.
10. Press **Ctrl**+**Shift** and the up arrow key.	To select upward to the end of the region (the blank cells in the Comm column).
11. Press **Ctrl**+**V**.	To paste the formula.

Do This	How & Why
12. Click **Stop Recording**.	To finish recording the macro. Will it work on different data?
13. Activate the **June** worksheet and run the macro.	Press **Ctrl**+**Shift**+**C**. If you recorded the macro properly, it inserts the formula all the way down for all the reps. You could record both this and formatting as a single macro that would take care of the whole report, but of course, you'd want to practice a few times before trying!
14. Save and close the workbook.	

Assessment: Recording macros

A colleague sends you a workbook, and when you open it, you get a macro security warning. What should you do? Choose the one best answer.

- Go ahead and enable the content.
- Close the workbook immediately.
- **Open the workbook without enabling the content, and ask the colleague about macro content in the workbook.**

Which of the following are ways to run a macro? Choose all correct answers.

- **The Macros window**
- **Shortcut keys**
- **Buttons or objects**

You must always select the cell where you want to begin before recording a macro. True or false?

- True
- **False**

You can toggle between recording relative and absolute references while recording a macro. True or false?

- **True**
- False

Which of the following are locations where you can store VBA code? Choose all correct answers.

- **The current workbook**
- The global workbook
- **The personal macro workbook**
- **A new workbook**

Module B: Running macros

There are many ways to run macros. You can assign them to buttons on the ribbon or the Quick Access toolbar, or to any objects on your worksheets. You can also create them to run under certain conditions, such as when you open or close a workbook.

You will learn how to:

- Assign macros to buttons on the ribbon or the Quick Access toolbar

- Assign macros to objects on a worksheet

- Name macros so that they run when a workbook is opened or closed

Assigning macros

You can assign macros to run when you click commands, objects, and buttons. The process is slightly different for assigning them to toolbars than it is for objects.

Assigning macros to the ribbon or the Quick Access toolbar

To assign a macro to run when you click a command on either the Quick Access toolbar or the ribbon, you use the Excel Options window, just as you would to perform other customizations of those interface elements.

1. Display the Excel Options window for the element to which you want to add the macro.
   - Click the Customize Quick Access Toolbar arrow, then click **More Commands**.
   - Right click the ribbon, then click **Customize the Ribbon**.
2. In the "Choose commands from" list, click **Macros**.

   You will see a list of all the macros in open workbooks. If you have add-ins loaded, there may be quite a few.
3. Click the macro you want, then click **Add**.

   If you're adding the macro to a ribbon location, select that location on the right before you click **Add**.

4. If you like, click **Modify** to change the display name (screen tip) for the macro and its icon. Click **OK** to return to the Excel Options window.
5. Click **OK**.

Assigning macros to buttons and objects

Command buttons are *ActiveX controls* that can be used to run VBA code. You can assign macros to these by entering their names in the VBA code for the buttons. You can also assign macros to run when you click graphical objects on your worksheets.

- To assign a macro to a command button, you need to access the VBA editor and know the name of the macro.

    a) On the Developer tab, in the Controls group, Click **Insert**, then, under ActiveX Controls, click the command button.

    b) Click or drag on the worksheet to create the button. Drag its handles to change its size.

    c) Right-click the button, then click **Properties** to display its properties pane. To change the text that appears on the button, change the Caption property. Close the properties pane when you're done.

    d) Double-click the button to display its VBA code.

    e) In the button's Click procedure, type the name of the macro you want to run.

    f) Close the VBA editor.

- Assigning a macro to a graphical object is simpler, but you can assign only macros in this way, not other VBA code.

    a) Create the object.

    b) Right-click the object, then click **Assign Macro**.

    c) Click the macro you want, then click **OK**.

Exercise: Assigning macros to commands, buttons, and objects

Do This	How & Why
1. Open `Running Macros`.	From the `Macros and Forms` data folder. Click **Enable Macros** in the security warning. This workbook contains six worksheets with monthly sales data. It also has a macro that formats headings in the reports. You'll experiment with assigning that macro to various elements from which it can be run.
2. Add the FormatHeadings macro to the Quick Access toolbar.	
a) Click the Customize Quick Access Toolbar arrow, then click **More Commands**.	To display the Excel Options window.
b) In the "Choose command from" list, click **Macros**.	If you have any add-ins loaded (and you probably do), you'll see quite a long list of macros.
c) Click **FormatHeadings**, then click **Add**.	
d) Click **Modify**.	To display the Modify Button window, where you could change the display name (screen tip) for the macro, or add an icon for it.
e) Click **Cancel**.	To return to the Excel Options window.
f) Click **OK**.	
3. On the Quick Access toolbar, click the FormatHeadings button.	
	The macro runs and formats the headings on the January worksheet.
4. Activate the **February** worksheet.	This worksheet has a rectangle object with the text "Format Headings" on it. You'll assign the macro to this object.
5. Assign the macro to the rectangle.	
a) Right-click the rectangle, then click **Assign Macro**.	
b) Click **FormatHeadings**, then click **OK**.	
6. Deselect the rectangle, then click it.	Another way to run a macro.
7. Save the workbook as My Running Macros.	

Do This	How & Why
8. Activate the **March** worksheet.	Here, you'll create a command button to run two macros, FormatHeadings, and CommCalc, which creates a commissions column on the report.
9. Insert a command button called `Sales Report`.	
a) On the Developer tab, in the Controls group, click **Insert**, then, under ActiveX Controls, click .	This is the Command Button ActiveX control.
b) Click in the worksheet to create the button.	
c) Right-click the button, then click **Properties**.	To display the properties for the button. There are many.
d) Change the Caption property to `Sales Report`, then close the Properties window.	It should look like this. Sales Report
10. Assign both macros to the command button.	
a) Double-click the command button.	To display the CommandButton1_Click() procedure for the button in the VBA window. This procedure will run whenever the button is clicked. You can add macros to it simply by typing their names as statements.

b) Add the macro names as statements in the procedure.	It should look like this.
	
	 `Private Sub CommandButton1_Click()`
	 `CommCalc`
	 `FormatHeadings`
	 `End Sub`

c) Click **File > Close and Return to Microsoft Excel**.	
11. On the Developer tab, notice that Design Mode is selected.	A button will not work while you are in Design mode, because Excel assumes you want to work on the button, not to use it.
12. Click **Design Mode**, then click the button.	The report is formatted and has a commissions column.
13. Save and then close the workbook.	

Running macros when a workbook is opened or closed

You can cause a macro to run automatically when a workbook opens simply by naming it Auto_Open. Similarly, you can cause a macro to run when a workbook closes by calling it Auto_Close. AutoOpen macros can be particularly useful, for example, if you want to refresh data from an external source whenever you open a workbook.

If you want certain macros to run every time you open Excel, you can put them in a procedure named Auto_Open that you store in your Personal Macro Workbook.

Copying macros between workbooks

Use the Visual Basic Editor to copy a macro module from one workbook to another.

Exam Objective: MOS Excel Expert 1.1.2

1. Open both the workbook containing the macro you want to copy, and the workbook to which you want to copy the macro.

2. On the Developer tab, click **Visual Basic** to display the Visual Basic Editor.

3. In the Project pane, expand the trees for the workbooks so you can see the module you want to copy.

4. Drag the module you want from the source workbook to the destination workbook.

Exercise: Refreshing data automatically when a workbook opens

If you want to demonstrate that the automatic refreshing of data works, you'll need to have Microsoft Access installed.

Do This	How & Why
1. Create a new, blank workbook.	
2. Save the workbook as My AutoOpen.	Save it in the current chapter's data folder.

Do This	How & Why
3. Create a data connection to the Invoices access database.	
a) On the Data tab, click **From Access**.	
b) Navigate to the current chapter's data folder, select Invoices, click **Open**, then click **OK**.	The table appears in the worksheet.
4. Update the workbook.	To be sure you always have the latest data when you open the workbook, you'll record a macro that will refresh the data, naming it so that it runs when you open the workbook.
5. On the Developer tab, click **Record Macro**.	
6. Name the macro `Auto_Open`, describe it as `Refresh data`, and click **OK**.	
7. On the Data tab, click **Refresh All**, then stop recording.	
8. Save the workbook as a Macro-Enabled Workbook called `My AutoOpen`.	
9. Demonstrate that the Auto_Open macro is refreshing the data.	
a) Note the total sales value for the first invoice.	It is "68.4."
b) Close the workbook.	
c) Open the `Invoices` database in Microsoft Access.	It is in the current chapter's data folder. Click **Enable Content** if the security warning appears.
d) In the Invoices table, change the first sales total value to `70`, then close Microsoft Access.	
e) Open the My AutoOpen macro-enabled workbook.	Be sure you're opening the macro-enabled one, and not the regular workbook. You will probably receive a security warning. If so, click **Enable Content**. Notice that the data has been refreshed (because the first sales value is now 70).
10. Save and close the workbook.	

Assessment: Running macros

You have created a macro to format the headings in a weekly report after you import the data. The macro is stored in a template that you use to create each weekly report. Which method of running the macro is best? Choose the one best answer.

- An Auto_Open macro

- A button on the Quick Access toolbar

- A command button

- **A graphic object on the worksheet**

You have to use the VBA editor to assign a macro to a command button ActiveX control. True or false?

- **True**

- False

Which approach would you use to run a macro every time you open Excel? Choose the best answer.

- Store it in the normal template.

- Name it Auto_Open, and store it in a workbook called AutoRun.

- **Name it Auto_Open, and store it in your Personal Macro Workbook.**

- Add it to the Quick Access toolbar.

Module C: Forms

You can create userforms, or simply "forms," when you want to get various kinds of input from users, and use VBA to take actions according to that input.

You will learn:

- About userforms and how they work

- About the various kinds of controls you can add to a userform

- How to set properties of userforms and controls

- How to edit VBA code associated with controls on a userform

About forms

Forms are a deep subject, and to be able to use them effectively, you will need to be very comfortable with VBA. However, all of this is beyond the scope of this course. But at its most basic level, a form is a window that you create in the VBA editor. After you create it, you set the form's properties, add controls to it, and then use VBA procedures to enable interaction of various kinds.

A sample form

A form is made up of static elements, such as text or graphics, and *controls*, which enable interaction through VBA. On the sample form, the text is static, while the buttons are controls.

Creating forms

You create a form by inserting a UserForm in the VBA editor.

1. Start the VBA editor (click **Visual Basic** on the Developer tab).

2. Click **Insert > UserForm** to display a blank form in the editor.

3. Set properties for the form in the Properties pane in the lower left.

    - If the Properties pane is not showing, press **F4** or click **View > Properties Window**.

    - The *(Name)* property is the name by which you will refer to the form in any VBA code. It cannot have spaces and should follow a clear, logical naming convention.

    - The *Caption* property is what will appear in the form's title bar.

    - There are many properties for a form. To view them in categories, click the **Categorized** tab in the Properties pane.

4. Add text, graphics, and controls to the form by using the toolbox, which you can display by clicking **View > Toolbox**.

5. Add code for the form and controls to control their behavior.

   Obviously, this is much more complicated than one simple sentence can convey.

6. To view what the form will look like when run, click the **Run** button.

Inserting controls

Controls give users the ability to interact with your forms. You add a control by clicking it in the Toolbox, then clicking or dragging on the form. Coding the interaction among the user, the form, the controls, and workbooks is a bit trickier, but with practice, you can learn to do so.

Exam Objective: MOS Excel Expert 2.3.5

Here are some of the more common form controls.

Form Controls

Button	Tool	Purpose
	Select Objects	Selecting controls and other objects on the form.
	Label	For static text on the form, usually used to describe controls or give instructions.
	TextBox	A box in which users can type.
	ComboBox	A box to display a value connected to a drop-down list.
	ListBox	Displays a list of values from which a user can choose.
	CheckBox	A box user checks by clicking.
	OptionButton	An option users can click. Usually, only one option button in a group can be selected at a time.
	CommandButton	A button that can be clicked to accomplish a VBA task.

1. Click the control you want.

2. Click or drag on the form to create the control.
   You can resize the control by selecting it, then dragging its handles.

3. Set any properties you want for the control.
   If you don't see the properties pane, press **F4**. Be sure to set the name property, so that you know how to refer to the control in code. Names can't have spaces, and should follow clear, logical conventions. Many programmers, for example, will start a control name with a three-letter prefix that identifies its type. For example, a Cancel button might be called "btnCancel," or a text box for entering department names would be "txtDepartment." The text that shows on a button is the caption property.

4. Create any VBA code you need for the controls.
   Again, VBA coding is beyond the scope of this course. One way to get the code you want is to record actions similar to those you want your controls to take, then copy and modify that recorded code. There is also an enormous amount of VBA code available online. You can usually find code that is close to what you want by searching for it on the Internet, copying it into a sub procedure, and then modifying it slightly to fit your needs.

5. Test the form by clicking the **Run** button.

Again, this is only a brief introduction to forms and controls. You will need to provide ways for a user to run a form (such as a button to display the form), and then ways for it to be hidden (perhaps a Cancel button). Trial and error will teach you a great deal.

Exercise: Creating a form to get user input

In this exercise, you will create a very simple form, place two controls on it, and then program a button to move text from a text box to a cell on the active worksheet. It's not the most realistic demonstration of a form, but it shows several of the basics you'll need if you go on to program more useful forms.

Exam Objective: MOS Excel Expert 2.3.5

Do This	How & Why
1. Open a new, blank workbook.	
2. Open the VBA editor.	On the Developer tab, click **Visual Basic**.
3. Click **Insert > UserForm**.	To open a new UserForm, or j"form," in the VBA editor. Basically, this is a blank window you can build on. You might also see the Toolbox next to the form, but don't worry for now; we'll make sure it's showing soon.
4. Observe the Project window.	In the upper left. It shows that a new userform has been added to the VBA project for this workbook.
5. In the form's Properties window, set the (Name) and Caption properties as shown.	Press F4 if you don't see the Properties window in the lower left. The name is important, because it is how you refer to the form. You should name forms and controls in a consistent, logical way, and without spaces. The caption property is what will show on the form's title bar.

Do This	How & Why
6. Display the Toolbox.	Click the form. If the Toolbox doesn't appear, click **View > Toolbox**. The Toolbox contains all the tools you need to build your form.
7. Click **abl** , then drag on the form.	To create a TextBox control that looks something like this (but don't worry about getting it exactly right).
8. Be sure the TextBox is selected, then change its (Name) property to `txtMyName`.	When you run the form, you'll enter your name in this text box. In a real form, you'd probably add some text above the TextBox control to explain what you want the user to do.
9. Click ⌐ , then click below the text box on the form.	To create a command button on the form. You could resize the button by dragging its handles.
10. Set the (Name) and Caption properties for the command button as shown.	Again, use consistent conventions for your control names. Here, we're using "cmd" as the prefix for the name. The button will enter the text in the text box on the active worksheet, so the name makes sense. The caption property is the text that will show on the button.
Continued...	

Do This	How & Why
11. Program the button's Click procedure to enter the text in the text box in cell A1 of the active worksheet.	
a) Double-click the command button.	To display the code window, with the button's click sub procedure (it's called cmdEnterText_Click()).
b) Enter the line of code shown in the procedure.	The first part of the line says you want to set the value of the range A1 on the active sheet (A1), then the rest says you want to set that as equal to the Text property of the txtMyName control. That property will contain whatever text is entered in the text box.

```
Private Sub cmdEnterText_Click()
    ActiveSheet.Range("A1").Value = txtMyName.Text
End Sub
```

Do This	How & Why
12. Close the code window, then click ▶ .	The Run button is on the main VBA editor toolbar. The form runs in Excel, with the worksheet behind it. You can drag or resize it as you would any window.

Do This	How & Why
13. Enter your name in the text box, then click **Enter Text**.	If you got your code right, the text will appear in cell A1 of the active sheet. This is only the introduction to a large subject. How would a user display the form? You would need a button or some other way to show or hide it. How would you make the form disappear when you're finished with it? You'd probably want a Cancel button that hides the form.
14. Click the form's close box.	To return to the VBA editor.
15. Close the VBA editor.	
16. Save the workbook as a macro-enabled workbook called `My Forms`, then close it.	

Assessment: Forms

Which of the following is the most accurate statement about creating forms? Choose the one best answer.

- You can create forms in Excel without using VBA.

- You create forms in the VBA editor, but don't need to use VBA code.

- **You create forms in the VBA editor and use VBA code to control how they work.**

You change the name of a form by clicking its title bar and typing. True or false?

- True

- **False**

Which property of a TextBox control would you access in your VBA code to obtain the text a user typed into the text box? Choose the one correct answer.

- Value

- (Name)

- **Text**

- Caption

Summary: Macros and forms

You should now know:

- About using macros to automate repetitive tasks, how to record and run a macro, and how to control the recording of relative and absolute references

- How to assign a macro to the Quick Access toolbar, to a graphical object, or to a command button, and how to cause a macro to run automatically when a workbook is opened or closed

- About using forms to gather input from users, how to create simple forms and change their properties, how to add controls and change their properties, and how to use VBA procedures to control the interaction between users, controls, and Excel

Synthesis: Macros and forms

In this activity, you'll record a macro to format sales numbers as currency and to best-fit columns. Then, you'll record a macro to select the current region, then select cell A1, and modify that macro with code to perform a sort. Finally, you'll create a button to run both macros.

1. Open `Macros Synthesis` from the current chapter's data folder. This workbook contains worksheets of partially formatted monthly sales reports. You'll create macros to format and sort similar data.

2. Record a macro called ReportFormat that will do all of the following:

   - Select cell A1 absolutely.

   - Select the current region relatively.

   - Apply currency formatting to two decimal places.

   - Select columns A:F absolutely.

   - Best-fit the selected columns.

   - Select cell A1 absolutely.

3. Test the macro on the February worksheet, then save the workbook as a macro-enabled workbook called My Macros Synthesis.

4. Activate January again, and then record a macro called ReportSort that will select cell A1 absolutely, select the current region relatively, and then re-select A1 absolutely. You will insert code into this macro to create a sorting procedure.

5. Edit the ReportSort macro, and add the code to perform a sort based on the Total column (column 6), as shown here. Hint: The inserted code is in a text file called SortCode.txt, if you want to copy and paste it.

```
    Range("A1").Select
    Selection.CurrentRegion.Select
    With ActiveSheet.Sort
        .SortFields.Clear
        .SortFields.Add Key:=Selection.Columns(6), Order:=xlDescending
        .SetRange Selection
        .Apply
    End With
    Range("A1").Select
End Sub
```

6. Test the ReportSort macro on the January and February data.

7. Save the workbook.

8. On the April worksheet, add a command button that will run both the ReportFormat and ReportSort macros. The caption on the button should read "Format Report."

9. Test the command button.

10. Save and close the workbook

Internationalization and Accessibility

You will learn:

- About internationalization and accessibility concerns for you workbooks

Preparing workbooks for internationalization and accessibility

Office 2016 provides settings and options that help make workbooks universally accessible to users and viewers. One of these methods is by breaking the language barrier: Microsoft applications accommodate an ever-expanding catalog of world languages for viewing and working with workbooks, and for using various international symbols and formats. Likewise, there are features that make documents more accessible to those with limited dexterity and movement, vision, and so on, as well as other disabilities.

Many of these considerations involve the application of techniques you've already learned, such as using a clear and easily read or scanned workbook structure and layout, or customizing the ribbon or Quick Access toolbar. But there are a few additional features and settings worth exploring.

You will learn:

- About language considerations for workbook internationalization, including how to change regions, display international formats, and use international symbols.

- About accessibility considerations for workbooks, including how to modify workbooks to be more accessible and to set +Body and +Heading fonts for a theme.

Internationalization

There are many aspects to internationalization of your workbooks. Windows has built-in language features that you can use, for example, after you've installed various foreign language components. Two useful features for internationalization are international data formats and international symbols.

Displaying international data formats

There are many techniques you can use to control the internationalization of data formats in Excel. Excel has many built-in international currency formats, and also has *Locale* settings for date formats. You can also change *Region* settings in Windows that will control how some date formats appear in Excel. Pay attention to whether you are changing formats that are affected by Windows Region settings; not all are.

Exam Objective: MOS Excel Expert 2.4.1,2.4.2

- Change Windows Region settings.

    a) Display the Windows Settings App.

      Here, you control all sorts of things about how Windows behaves.

    b) Click **Time & Language**.

      To display the Time & Language settings window.

    c) Click **Region & language**.

      Here, you can control the country or region in which you are working, or languages that are installed on your computer.

d) From the Country or region, list, select a region.

The region you select will mean that some apps will provide local content. And one of those Apps is Excel, which in some cases will customize date formats to match your region.

e) Close the Time & Language window.

- Select a localized date format.

a) Display the Number tab of the Format Cells window.

b) Click the Date category.

There are two things worth noting here.

Any date format with an asterisk, as the two circled here have, will change to reflect the region selected in the Windows Region & Language settings. It's a good idea to stick with these region-sensitive formats unless you have a good reason not to (for example, you want Americans to see UK formats for some reason).

You can pick a locale and then select a local format for that locale. But most of those are not region-dependent. So, if you select a UK format in this way, any user anywhere in the world will see that UK format, regardless of their region settings.

     c) Select the format you want and close the window.

- Select an international currency format.

     a) Display the Number tab of the Format Cells window.

     b) Click the **Currency** category.

     c) Select an international currency symbol from the Symbol list.

        Note that these symbols are not region-dependent. So all users will see these symbols. This makes sense because in general, you will be meaning to use a specific currency with the data in your workbooks.

     d) Select a format and close the window.

Using international symbols

You can also insert international characters as specific text along with your data.

1. Select the cell in which you want to enter an international symbol, and begin to type the data.

2. On the Insert tab, click **Symbol**.

     To display the **Symbol** window, where you can choose from an enormous variety of symbols.

3. Select a Font and a Subset.

     The subsets are particularly useful for finding international symbols.

     Here, you see the Latin-1 Supplement subset of the normal text font. It contains many of the common symbols you need for many European languages.

4. Select the symbol you want and click **Insert**.

5. Close the Symbol window.

Accessibility

In today's multimedia world, it's become commonplace to be able to view a workbook on a computer, as a printout, as a webpage, on a smartphone, and so on. You need to make sure users can access your information regardless of the medium. It's also important to make documents more accessible to those with limited dexterity and movement, vision, and so on, as well as other disabilities. Fortunately, Office 2016 applications provide users, particularly those with disabilities, easier access to their many features.

Many of the techniques you've already learned, such as using a clear and easily read and scanned workbook structure and layout, or customizing the ribbon or Quick Access toolbar, can help greatly in allowing users better access to documents and tools. But there are a few additional features and settings worth exploring.

Using accessibility tools

You should use Excel's Accessibility Checker to locate potential Accessibility issues. It will not fix them for you, but it will provide information about where the issues are, why they are issues, and what you can do to fix them and improve the workbook's accessibility.

Exam Objective: MOS Excel Core 1.5.7

1. In Backstage view, click **Info**, then click **Check for Issues > Check Accessibility**.

To display the Accessibility Checker on the right side of your workbook.

2. For each issue located, try to solve the problem.

a) Select the issue to get information about it.

For example, here you can see that the checker found an image that does not have alternative text. The checker tells you the procedure for fixing the problem.

Inspection Results

ERRORS

▲ Missing Alt Text
   Picture 1 (Sheet1)

WARNINGS

▲ Default Sheet Names
   Sheet1

Additional Information   ˅
1) Right-click on the object, and click
Format.
2) Click Layout & Properties.
3) Expand the Alt Text section.
4) Type a description of the object
into the Title and Description boxes.

b) Fix any issues the checker finds.

As you fix the issues, they will disappear from the Accessibility Checker. When there are none remaining, you'll see this message.

✓ No accessibility issues found.
People with disabilities should not
have difficulty reading this
workbook.

3. Close the Accessibility Checker.

Summary: Internationalization and accessibility

You should now know:

- About language considerations for workbook internationalization, including how to change regions, how to identify date formats that are region-sensitive, how to set locale for date formats, how to display international currencies, and how to insert international symbols

- About accessibility considerations for workbooks, including how to use the Accessibility Checker to find and fix potential accessibility issues in workbooks, and how to specify +Body and +Heading fonts for a new theme

Alphabetical Index