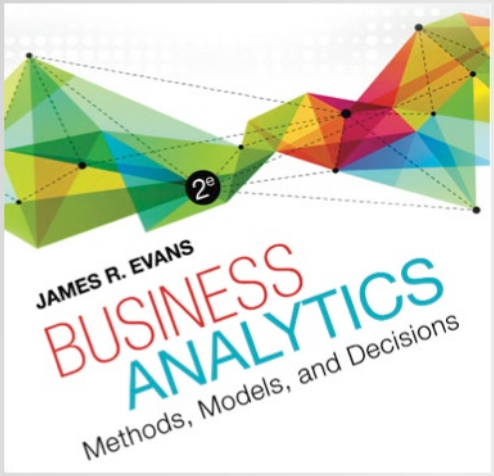



# Chapter 2 Analytics on Spreadsheets

# Why Spreadsheets?

- Many commercial software packages can be used for Business Analytics.
- Spreadsheet software, such as Microsoft Excel, is widely available and used across all areas of business.
- Spreadsheets provide a flexible modeling environment for manipulating data and developing and solving models.

# Windows vs. Mac

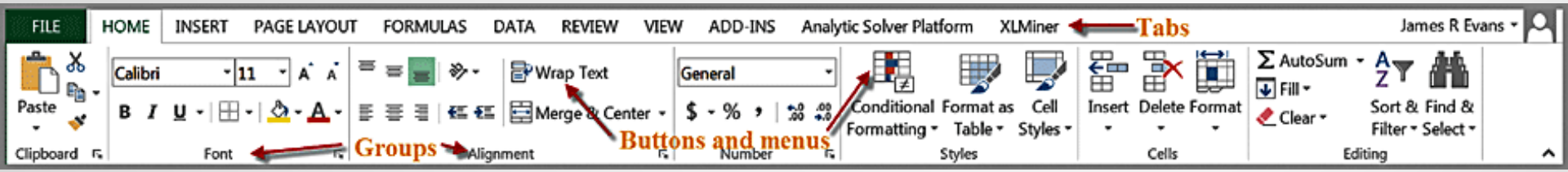
- Mac versions of Excel do not have the full functionality that Windows versions have – particularly statistical features which are important to this book.
- The Excel add-in that we use in later chapters, *Analytic Solver Platform*, only runs on Windows. Thus, if you use a Mac, you should either run Bootcamp with Windows or use a third-party software product such as *Parallels* or *VMWare*.

# Basic Excel Skills You Should Know

- Opening, saving, and printing files
- Using workbooks and worksheets
- Moving around a spreadsheet
- Selecting cells and ranges
- Inserting/deleting rows and columns
- Entering and editing text, data, and formulas
- Formatting data (number, currency, decimal)
- Working with text strings
- Formatting data and text
- Modifying the appearance of a spreadsheet

# Excel Ribbon

- Tabs - Home, Insert, Page Layout, Formulas, …
- Groups - Font, Alignment, Number, Styles, …
- Buttons and Menus
  - Buttons appear as small icons.
  - Menus of additional choices are indicated by small triangles.



# Excel Formulas

- Common mathematical operators are used.
- For example:

$a - bP^5 + \frac{c}{d}$ would be entered into Excel as:

=a− (b*P^5) + (c/d)

# Relative and Absolute References

- Cell references can be **relative** or **absolute**. Using a dollar sign before a row and/or column label creates an absolute reference.
  - Relative references: A2, C5, D10
  - Absolute references: $A$2, $C5, D$10
- Using a $ sign before a row label (for example, B$4) keeps the reference fixed to row 4 but allows the column reference to change if the formula is copied to another cell.
- Using a $ sign before a column label (for example, $B4) keeps the reference to column B fixed but allows the row reference to change.
- Using a $ sign before both the row and column labels (for example, $B$4) keeps the reference to cell B4 fixed no matter where the formula is copied.

# Example 2.1 Implementing Price-Demand Models in Excel

Two models for predicting demand as a function of price

Linear

$D = a - bP$

Formula in cell B8:

=$B$4-$B$5*$A8

Nonlinear

$D = cP^{-d}$

Formula in cell E8:

=$E$4*D8^-$E$5

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | **Demand Prediction Models** | | | | |
| 2 | | | | | |
| 3 | **Linear Model** | | | **Nonlinear Model** | |
| 4 | **a** | 20,000 | | **c** | 20,000 |
| 5 | **b** | 10 | | **d** | 0.0111382 |
| 6 | | | | | |
| 7 | **Price** | **Demand** | | **Price** | **Demand** |
| 8 | $80.00 | $19,200 | | $70.00 | $19,075.63 |
| 9 | $90.00 | $19,100 | | $80.00 | $19,047.28 |
| 10 | $100.00 | $19,000 | | $90.00 | $19,022.31 |
| 11 | $110.00 | $18,900 | | $100.00 | $19,000.00 |
| 12 | $120.00 | $18,800 | | $110.00 | $18,979.84 |
| 13 | | | | $120.00 | $18,961.45 |
| 14 | | | | $130.00 | $18,944.56 |

Note how the absolute addresses are used so that as these formulas are copied down, the demand is computed correctly.

# Copying Formulas

Formulas in cells can be copied in many ways.

- Use the *Copy* button in the *Home* tab, then use the Paste button
- Use *Ctrl-C*, then *Ctrl-V*
- Drag the bottom right corner of a cell (the fill handle) across a row or column

# Basic Excel Functions

- =MIN(*range*)
- =MAX(*range*)
- =SUM(*range*)
- =AVERAGE(*range*)
- =COUNT(*range*)
- =COUNTIF(*range,criteria*)
  - Excel has other useful COUNT-type functions: COUNTA counts the number of nonblank cells in a range, and COUNTBLANK counts the number of blank cells in a range. In addition, COUNTIFS(*range1, criterion1, range2, criterion2,… range_n, criterion_n*) finds the number of cells within multiple ranges that meet specific criteria for each range.

# Example 2.2 Using Basic Excel Functions

| | A | B | C | D | E | F | G | H | I | J |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | **Purchase Orders** | | | | | | | | | |
| 2 | | | | | | | | | | |
| 3 | **Supplier** | **Order No.** | **Item No.** | **Item Description** | **Item Cost** | **Quantity** | **Cost per order** | **A/P Terms (Months)** | **Order Date** | **Arrival Date** |
| 4 | Hulkey Fasteners | Aug11001 | 1122 | Airframe fasteners | $ 4.25 | 19,500 | $ 82,875.00 | 30 | 08/05/11 | 08/13/11 |
| 5 | Alum Sheeting | Aug11002 | 1243 | Airframe fasteners | $ 4.25 | 10,000 | $ 42,500.00 | 30 | 08/08/11 | 08/14/11 |
| 6 | Fast-Tie Aerospace | Aug11003 | 5462 | Shielded Cable/ft. | $ 1.05 | 23,000 | $ 24,150.00 | 30 | 08/10/11 | 08/15/11 |
| 7 | Fast-Tie Aerospace | Aug11004 | 5462 | Shielded Cable/ft. | $ 1.05 | 21,500 | $ 22,575.00 | 30 | 08/15/11 | 08/22/11 |
| 8 | Steelpin Inc. | Aug11005 | 5319 | Shielded Cable/ft. | $ 1.10 | 17,500 | $ 19,250.00 | 30 | 08/20/11 | 08/31/11 |
| 9 | Fast-Tie Aerospace | Aug11006 | 5462 | Shielded Cable/ft. | $ 1.05 | 22,500 | $ 23,625.00 | 30 | 08/20/11 | 08/26/11 |
| 10 | Steelpin Inc. | Aug11007 | 4312 | Bolt-nut package | $ 3.75 | 4,250 | $ 15,937.50 | 30 | 08/25/11 | 09/01/11 |
| 11 | Durrable Products | Aug11008 | 7258 | Pressure Gauge | $ 90.00 | 100 | $ 9,000.00 | 45 | 08/25/11 | 08/28/11 |
| 12 | Fast-Tie Aerospace | Aug11009 | 6321 | O-Ring | $ 2.45 | 1,300 | $ 3,185.00 | 30 | 08/25/11 | 09/04/11 |
| 96 | Steelpin Inc. | Nov11009 | 5677 | Side Panel | $ 195.00 | 110 | $ 21,450.00 | 30 | 11/05/11 | 11/17/11 |
| 97 | Manley Valve | Nov11010 | 9955 | Door Decal | $ 0.55 | 125 | $ 68.75 | 30 | 11/05/11 | 11/10/11 |
| 98 | | | | | | | | | | |
| 99 | Minimum Quantity | 90 | | =MIN(F4:F97) | | | | | | |
| 100 | Maximum Quantity | 25,000 | | =MAX(F4:F97) | | | | | | |
| 101 | Total Order Costs | $ 2,471,760.00 | | =SUM(G4:G97) | | | | | | |
| 102 | Average Number of A/P Months | 30.63829787 | | =AVERAGE(H4:H97) | | | | | | |
| 103 | Number of Purchase Orders | 94 | | =COUNT(B4:B97) | | | | | | |
| 104 | Number of O-ring Orders | 12 | | =COUNTIF(D4:D97,"=O-Ring") | | | | | | |
| 105 | Number of A/P Terms < 30 | 17 | | =COUNTIF(H4:H97,"<30") | | | | | | |
| 106 | Number of O-ring Orders Spacetime | 3 | | =COUNTIFS(D4:D97,"O-Ring",A4:A97,"Spacetime Technologies") | | | | | | |

# Other IF-Type Functions

- SUMIF, AVERAGEIF, SUMIFS, and AVERAGEIFS can be used to embed IF logic within mathematical functions.
- For instance, the syntax of SUMIF is
  - SUMIF(*range, criterion, [sum range]*). "Sum range" is an optional argument that allows you to add cells in a different range.
- Example: In the *Purchase Orders* database, to find the total cost of all airframe fasteners, use

  =SUMIF(D4:D97,"Airframe fasteners", G4:G97)

# Functions for Specific Applications

- Net Present Value (or discounted cash flow) measures the worth of a stream of cash flows, taking into account the time value of money.
- Excel function: =NPV(*rate,value1,value2,…*)
  - *F* is the cash flow ($)
  - *Rate (i)* is the discount rate
  - *value1, value2*,…are equally-spaced payments or income values
  - *t* is a time period

$$\text{NPV} = \sum_{t=0}^{n} \frac{F_t}{(1+i)^t}$$

# Example 2.3 Using the NPV Function

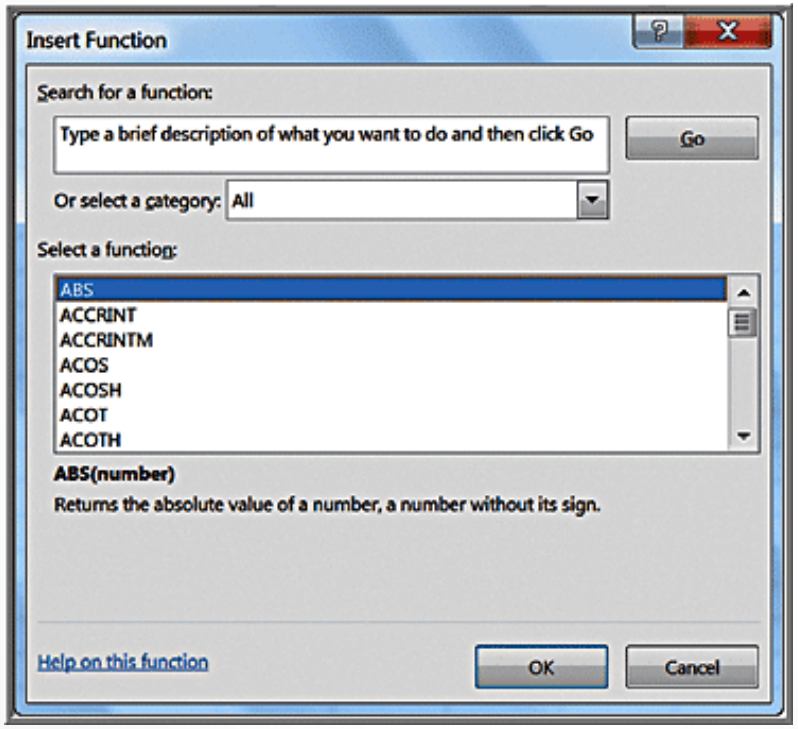
Cell B8:

=NPV(B6, C4:H4) – B5

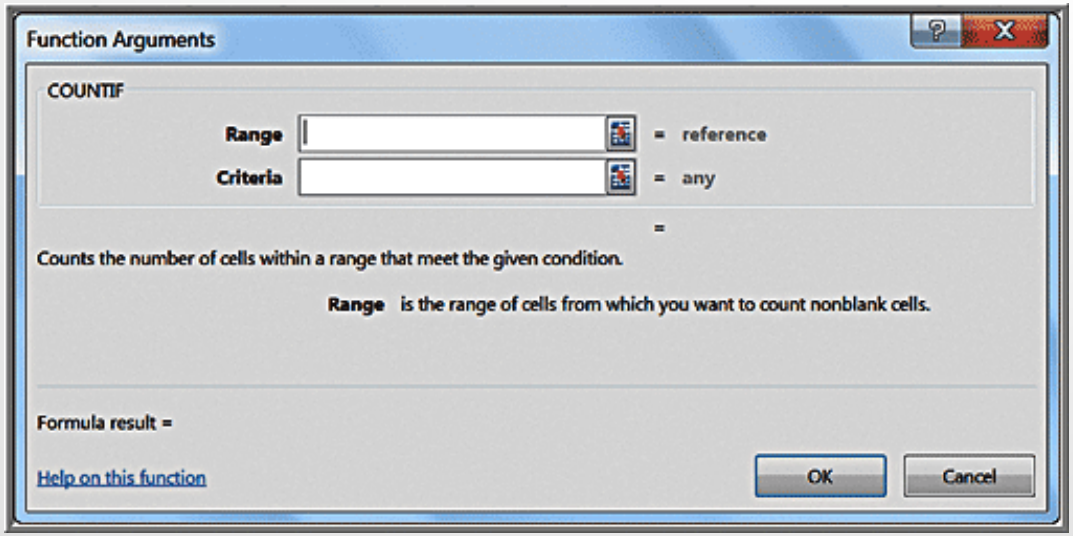
| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | **Net Present Value** | | | | | | | |
| 2 | | | | | | | | |
| 3 | | **Month** | **January** | **February** | **March** | **April** | **May** | **June** |
| 4 | | **Sales Revenue Forecast** | $2,500 | $4,000 | $5,000 | $8,000 | $10,000 | $12,500 |
| 5 | **Fixed Cost** | $25,000.00 | | | | | | |
| 6 | **Discount Rate** | 3% | | | | | | |
| 7 | | | | | | | | |
| 8 | **NPV** | $11,975.81 | | | | | | |

# Insert Function

- Click the Insert function button $f_x$.
- You may type in a description or search.




Example for COUNTIF function



# Logical Functions

- =IF(*condition, value if true, value if false*) – a returns one value if the condition is true and another if the condition is false,
- =AND(*condition1, condition2, …*) – returns TRUE if all conditions are true and FALSE if not,
- =OR(*condition1, condition2, …*) – returns TRUE if any condition is true and FALSE if not.

# IF Function

- =IF(*condition, value if true, value if false*)
- Conditions may include the following:

| | |
|---|---|
| = equal | <> not equal to |
| > greater than | >= greater than or equal to |
| < less than | <= less than or equal to |

- You may nest up to 7 IF functions, replacing the *value if false* with another IF function
- Example:

=IF(A8 =2,(IF(B3 =5,”YES”,“ ”)),15)

# Example 2.4 Using the IF Function

- Suppose that orders with quantities of at least 10,000 units are classified as Large.
  - Cell K4: =IF(F4>=10000, “Large”, “Small”)
- Suppose that large orders with a total cost of at least $25,000 are considered critical.
  - Cell L4: =IF(AND(K4=“Large”, G4>=25000),“Critical”,“”)

| | A | B | C | D | E | F | G | H | I | J | K | L |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | **Purchase Orders** | | | | | | | | | | | |
| 2 | | | | | | | | | | | | |
| 3 | **Supplier** | **Order No.** | **Item No.** | **Item Description** | **Item Cost** | **Quantity** | **Cost per order** | **A/P Terms (Months)** | **Order Date** | **Arrival Date** | **Order Size** | **Type** |
| 4 | Hulkey Fasteners | Aug11001 | 1122 | Airframe fasteners | $ 4.25 | 19,500 | $ 82,875.00 | 30 | 08/05/11 | 08/13/11 | Large | Critical |
| 5 | Alum Sheeting | Aug11002 | 1243 | Airframe fasteners | $ 4.25 | 10,000 | $ 42,500.00 | 30 | 08/08/11 | 08/14/11 | Large | Critical |
| 6 | Fast-Tie Aerospace | Aug11003 | 5462 | Shielded Cable/ft. | $ 1.05 | 23,000 | $ 24,150.00 | 30 | 08/10/11 | 08/15/11 | Large | |
| 7 | Fast-Tie Aerospace | Aug11004 | 5462 | Shielded Cable/ft. | $ 1.05 | 21,500 | $ 22,575.00 | 30 | 08/15/11 | 08/22/11 | Large | |
| 8 | Steelpin Inc. | Aug11005 | 5319 | Shielded Cable/ft. | $ 1.10 | 17,500 | $ 19,250.00 | 30 | 08/20/11 | 08/31/11 | Large | |
| 9 | Fast-Tie Aerospace | Aug11006 | 5462 | Shielded Cable/ft. | $ 1.05 | 22,500 | $ 23,625.00 | 30 | 08/20/11 | 08/26/11 | Large | |
| 10 | Steelpin Inc. | Aug11007 | 4312 | Bolt-nut package | $ 3.75 | 4,250 | $ 15,937.50 | 30 | 08/25/11 | 09/01/11 | Small | |
| 11 | Durrable Products | Aug11008 | 7258 | Pressure Gauge | $ 90.00 | 100 | $ 9,000.00 | 45 | 08/25/11 | 08/28/11 | Small | |
| 12 | Fast-Tie Aerospace | Aug11009 | 6321 | O-Ring | $ 2.45 | 1,300 | $ 3,185.00 | 30 | 08/25/11 | 09/04/11 | Small | |
| 13 | Fast-Tie Aerospace | Aug11010 | 5462 | Shielded Cable/ft. | $ 1.05 | 22,500 | $ 23,625.00 | 30 | 08/25/11 | 09/02/11 | Large | |
| 14 | Steelpin Inc. | Aug11011 | 5319 | Shielded Cable/ft. | $ 1.10 | 18,100 | $ 19,910.00 | 30 | 08/25/11 | 09/05/11 | Large | |
| 15 | Hulkey Fasteners | Aug11012 | 3166 | Electrical Connector | $ 1.25 | 5,600 | $ 7,000.00 | 30 | 08/25/11 | 08/29/11 | Small | |

# Lookup Functions for Database Queries

- These functions are useful for finding specific data in a spreadsheet.
- =VLOOKUP(*lookup_value, table_array, col_index_num, [range lookup]*) - looks up a value in the leftmost column of a table and returns a value in the same row from a column you specify
- =HLOOKUP(*lookup_value, table_array, row_index_num, [range lookup]*) - looks up a value in the top row of a table and returns a value in the same column from a row you specify.
- =INDEX(*array, row_num, col_num*) - returns a value or reference of the cell at the intersection of a particular row and column in a given range.
- =MATCH(*lookup_value, lookup_array, match_type*) - returns the relative position of an item in an array that matches a specified value in a specified order

# Important Notes on Lookup Functions

- In the VLOOKUP and HLOOKUP functions, *range lookup* is optional. If this is omitted or set as *True*, then the first column of the table must be sorted in ascending numerical order.
- If an exact match for the *lookup_value* is found in the first column, then Excel will return the value the *col_index_num* of that row. If an exact match is not found, Excel will choose the row with the largest value in the first column that is less than the l*ookup_value*.
- If range lookup is *False*, then Excel seeks an exact match in the first column of the table range. If no exact match is found, Excel will return #N/A (not available).
- We recommend that you specify the range lookup to avoid errors.

# Example 2.5 Using the VLOOKUP Function

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | Sales Transactions: July 14 | | | | | | | |
| 2 | | | | | | | | |
| 3 | Cust ID | Region | Payment | Transaction Code | Source | Amount | Product | Time Of Day |
| 4 | 10001 | East | Paypal | 93816545 | Web | $20.19 | DVD | 22:19 |
| 5 | 10002 | West | Credit | 74083490 | Web | $17.85 | DVD | 13:27 |
| 6 | 10003 | North | Credit | 64942368 | Web | $23.98 | DVD | 14:27 |
| 7 | 10004 | West | Paypal | 70560957 | Email | $23.51 | Book | 15:38 |
| 8 | 10005 | South | Credit | 35208817 | Web | $15.33 | Book | 15:21 |
| 9 | 10006 | West | Paypal | 20978903 | Email | $17.30 | DVD | 13:11 |
| 10 | 10007 | East | Credit | 80103311 | Web | $177.72 | Book | 21:59 |
| 11 | 10008 | West | Credit | 14132683 | Web | $21.76 | Book | 4:04 |
| 12 | 10009 | West | Paypal | 40128225 | Web | $15.92 | DVD | 19:35 |
| 13 | 10010 | South | Paypal | 49073721 | Web | $23.39 | DVD | 13:26 |

=VLOOKUP(10007, $A$4:$H$475,3) returns the payment type Credit.

=VLOOKUP(10007, $A$4:$H$475,4) returns the transaction code 80103311

# INDEX Function

- =INDEX(*array, row_num, col_num*)
- The INDEX function works as a lookup procedure by returning the value in a particular row and column of an array. For example, in the Sales Transactions database,
- INDEX($A$4:$H$475, 7, 4) would retrieve the transaction code 80103311, which is in the 7th row and 4th column.

# MATCH Function

- =MATCH(*lookup_value, lookup_array, match_type*)
- In the MATCH function, *lookup_value* is value that you want to match in *lookup_array*, which is the range of cells being searched. The *match_type* is either -1, 0, or 1. The default is 1.
- If *match_type* = 1, then the function finds the largest value that is less than or equal to *lookup_value*.
  - The values in the *lookup_array* must be placed in ascending order.
- If *match_type* = 0, MATCH finds the first value that is exactly equal to *lookup_value*.
  - The values in the *lookup_array* can be in any order.
- If *match_type* = -1, then the function finds the smallest value that is greater than or equal to *lookup_value*.
  - The values in the *lookup_array* must be placed in descending order.

# Example 2.6 Using INDEX and MATCH Functions for Database Queries

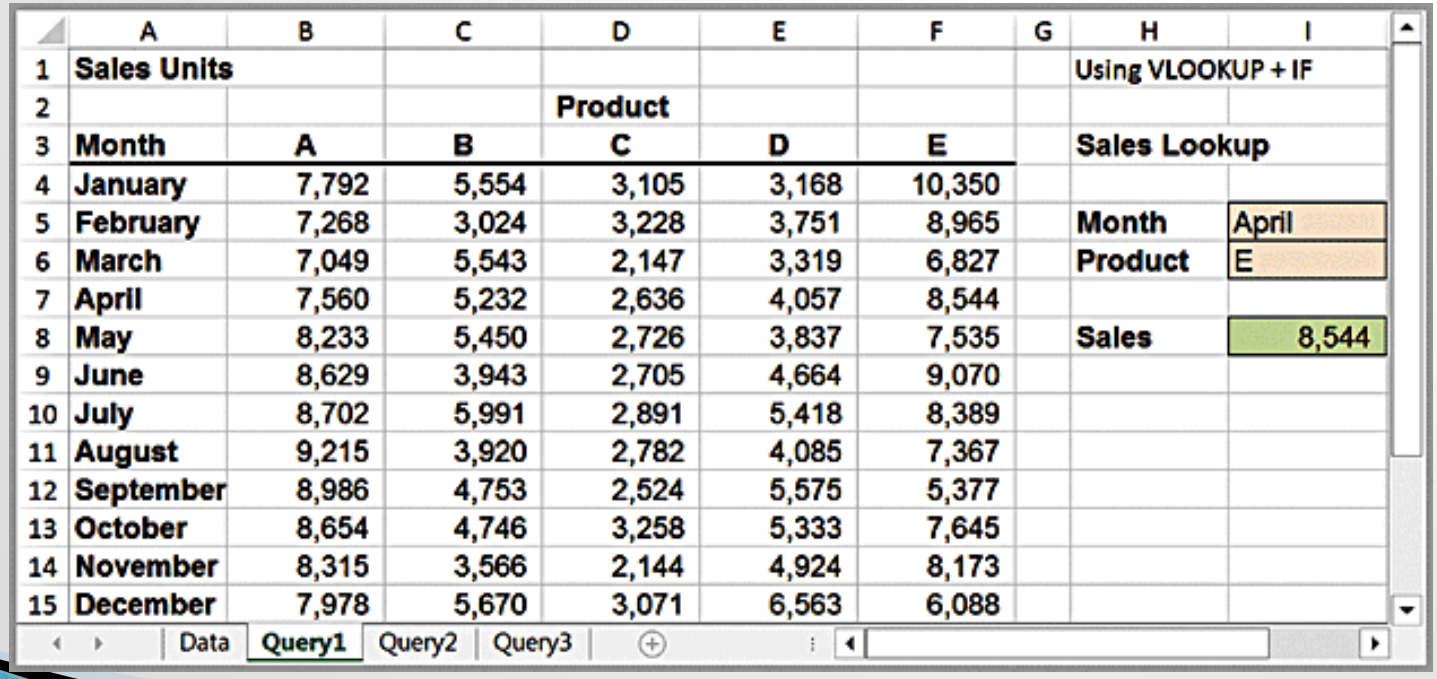
Suppose we wish to design a simple query application to input the month and product name, and retrieve the corresponding sales. The three additional worksheets in the workbook show how to do this in three different ways.

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | Sales Units | | | | | |
| 2 | | | | Product | | |
| 3 | Month | A | B | C | D | E |
| 4 | January | 7,792 | 5,554 | 3,105 | 3,168 | 10,350 |
| 5 | February | 7,268 | 3,024 | 3,228 | 3,751 | 8,965 |
| 6 | March | 7,049 | 5,543 | 2,147 | 3,319 | 6,827 |
| 7 | April | 7,560 | 5,232 | 2,636 | 4,057 | 8,544 |
| 8 | May | 8,233 | 5,450 | 2,726 | 3,837 | 7,535 |
| 9 | June | 8,629 | 3,943 | 2,705 | 4,664 | 9,070 |
| 10 | July | 8,702 | 5,991 | 2,891 | 5,418 | 8,389 |
| 11 | August | 9,215 | 3,920 | 2,782 | 4,085 | 7,367 |
| 12 | September | 8,986 | 4,753 | 2,524 | 5,575 | 5,377 |
| 13 | October | 8,654 | 4,746 | 3,258 | 5,333 | 7,645 |
| 14 | November | 8,315 | 3,566 | 2,144 | 4,924 | 8,173 |
| 15 | December | 7,978 | 5,670 | 3,071 | 6,563 | 6,088 |

Data | Query1 | Query2 | Query3

# Example 2.6: Using VLOOKUP + IF

The Query1 worksheet uses the VLOOKUP function with embedded IF statements. The formulas in cell I8 is: =VLOOKUP(I5,A4:F15,IF(I6="A",2,IF(I6="B",3,IF(I6="C",4,IF(I6="D",5,IF(I6="E",6))))),FALSE)



| | A | B | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Sales Units | | | | | | | Using VLOOKUP + IF | |
| 2 | | | | Product | | | | | |
| 3 | Month | A | B | C | D | E | | Sales Lookup | |
| 4 | January | 7,792 | 5,554 | 3,105 | 3,168 | 10,350 | | | |
| 5 | February | 7,268 | 3,024 | 3,228 | 3,751 | 8,965 | | Month | April |
| 6 | March | 7,049 | 5,543 | 2,147 | 3,319 | 6,827 | | Product | E |
| 7 | April | 7,560 | 5,232 | 2,636 | 4,057 | 8,544 | | | |
| 8 | May | 8,233 | 5,450 | 2,726 | 3,837 | 7,535 | | Sales | 8,544 |
| 9 | June | 8,629 | 3,943 | 2,705 | 4,664 | 9,070 | | | |
| 10 | July | 8,702 | 5,991 | 2,891 | 5,418 | 8,389 | | | |
| 11 | August | 9,215 | 3,920 | 2,782 | 4,085 | 7,367 | | | |
| 12 | September | 8,986 | 4,753 | 2,524 | 5,575 | 5,377 | | | |
| 13 | October | 8,654 | 4,746 | 3,258 | 5,333 | 7,645 | | | |
| 14 | November | 8,315 | 3,566 | 2,144 | 4,924 | 8,173 | | | |
| 15 | December | 7,978 | 5,670 | 3,071 | 6,563 | 6,088 | | | |

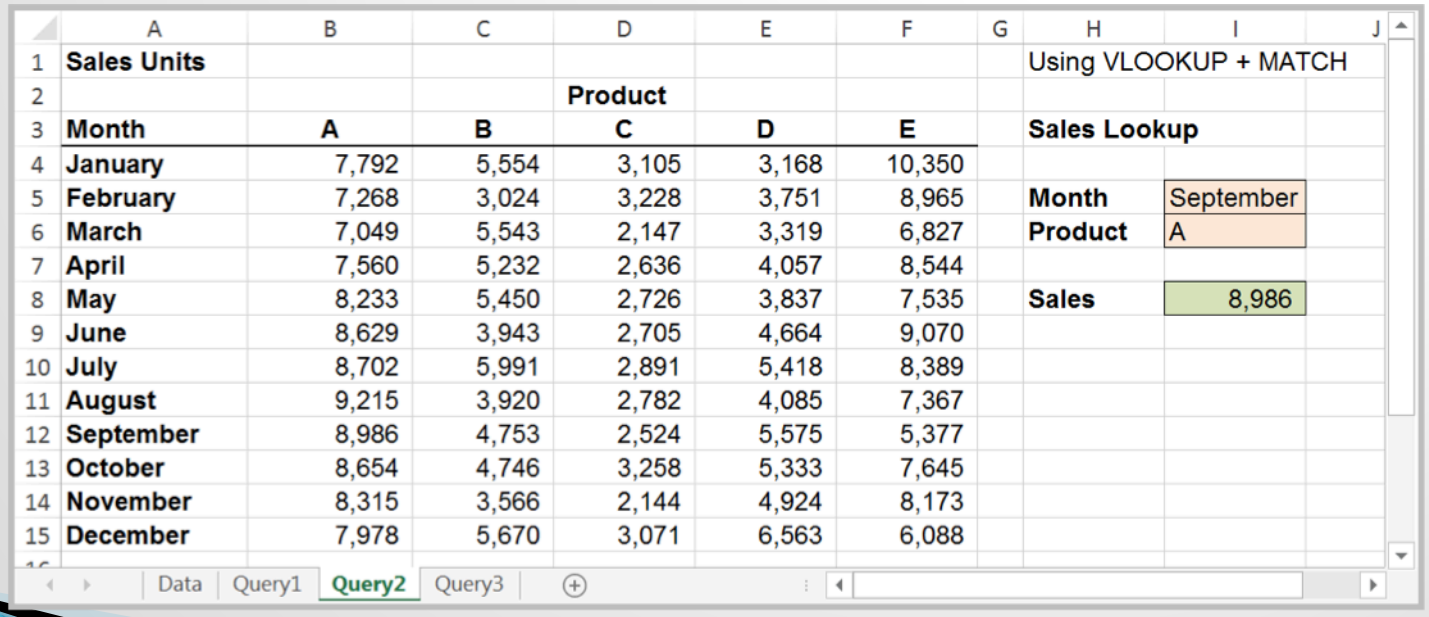
Data | Query1 | Query2 | Query3

# Example 2.6: Using VLOOKUP + MATCH

The formula in cell I8 is:

=VLOOKUP(I5,A4:F15,MATCH(I6,B3:F3,FALSE)+1,FALSE)

In this case, the MATCH function is used to identify the column in the table corresponding to the product name in cell I6. Note the use of the "+1" to shift the relative column number of the product to the correct column number in the lookup table.



| | A | B | C | D | E | F | G | H | I | J |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | **Sales Units** | | | | | | | Using VLOOKUP + MATCH | | |
| 2 | | | | **Product** | | | | | | |
| 3 | **Month** | **A** | **B** | **C** | **D** | **E** | | **Sales Lookup** | | |
| 4 | **January** | 7,792 | 5,554 | 3,105 | 3,168 | 10,350 | | | | |
| 5 | **February** | 7,268 | 3,024 | 3,228 | 3,751 | 8,965 | | **Month** | September | |
| 6 | **March** | 7,049 | 5,543 | 2,147 | 3,319 | 6,827 | | **Product** | A | |
| 7 | **April** | 7,560 | 5,232 | 2,636 | 4,057 | 8,544 | | | | |
| 8 | **May** | 8,233 | 5,450 | 2,726 | 3,837 | 7,535 | | **Sales** | 8,986 | |
| 9 | **June** | 8,629 | 3,943 | 2,705 | 4,664 | 9,070 | | | | |
| 10 | **July** | 8,702 | 5,991 | 2,891 | 5,418 | 8,389 | | | | |
| 11 | **August** | 9,215 | 3,920 | 2,782 | 4,085 | 7,367 | | | | |
| 12 | **September** | 8,986 | 4,753 | 2,524 | 5,575 | 5,377 | | | | |
| 13 | **October** | 8,654 | 4,746 | 3,258 | 5,333 | 7,645 | | | | |
| 14 | **November** | 8,315 | 3,566 | 2,144 | 4,924 | 8,173 | | | | |
| 15 | **December** | 7,978 | 5,670 | 3,071 | 6,563 | 6,088 | | | | |

Data | Query1 | Query2 | Query3

# Example 2.6: Using INDEX + MATCH

The formula in cell I8 is:

=INDEX(A4:F15,MATCH(I5,A4:A15,FALSE),MATCH(I6,B3:F3,FALSE)+1)

The MATCH functions are used as arguments in the INDEX function to identify the row and column numbers in the table based on the month and product name. The INDEX function then retrieves the value in the corresponding row and column.

| | A | B | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|
| 1 | **Sales Units** | | | | | | | Using INDEX + MATCH | |
| 2 | | | | **Product** | | | | | |
| 3 | **Month** | **A** | **B** | **C** | **D** | **E** | | **Sales Lookup** | |
| 4 | **January** | 7,792 | 5,554 | 3,105 | 3,168 | 10,350 | | | |
| 5 | **February** | 7,268 | 3,024 | 3,228 | 3,751 | 8,965 | | **Month** | August |
| 6 | **March** | 7,049 | 5,543 | 2,147 | 3,319 | 6,827 | | **Product** | B |
| 7 | **April** | 7,560 | 5,232 | 2,636 | 4,057 | 8,544 | | | |
| 8 | **May** | 8,233 | 5,450 | 2,726 | 3,837 | 7,535 | | **Sales** | 9,215 |
| 9 | **June** | 8,629 | 3,943 | 2,705 | 4,664 | 9,070 | | | |
| 10 | **July** | 8,702 | 5,991 | 2,891 | 5,418 | 8,389 | | | |
| 11 | **August** | 9,215 | 3,920 | 2,782 | 4,085 | 7,367 | | | |
| 12 | **September** | 8,986 | 4,753 | 2,524 | 5,575 | 5,377 | | | |
| 13 | **October** | 8,654 | 4,746 | 3,258 | 5,333 | 7,645 | | | |
| 14 | **November** | 8,315 | 3,566 | 2,144 | 4,924 | 8,173 | | | |
| 15 | **December** | 7,978 | 5,670 | 3,071 | 6,563 | 6,088 | | | |

Data | Query1 | Query2 | **Query3**

# Homework 2 - – Email the answers to me (albert.kalim@asbury.edu) by Sunday, 5/22/22, by 11:59 p.m. ET 10 points total (2.5 points each)

- 1. Chapter 2, Problem 3 (President's Inn Guest)
- 2. Chapter 2, Problem 5 (Purchase Orders)
- 3. Chapter 2, Problem 6 (Pharmaceutical)
- 4. Chapter 2, Problem 10 (General Appliance)