MICROSOFT EXCEL

Functions

Contents

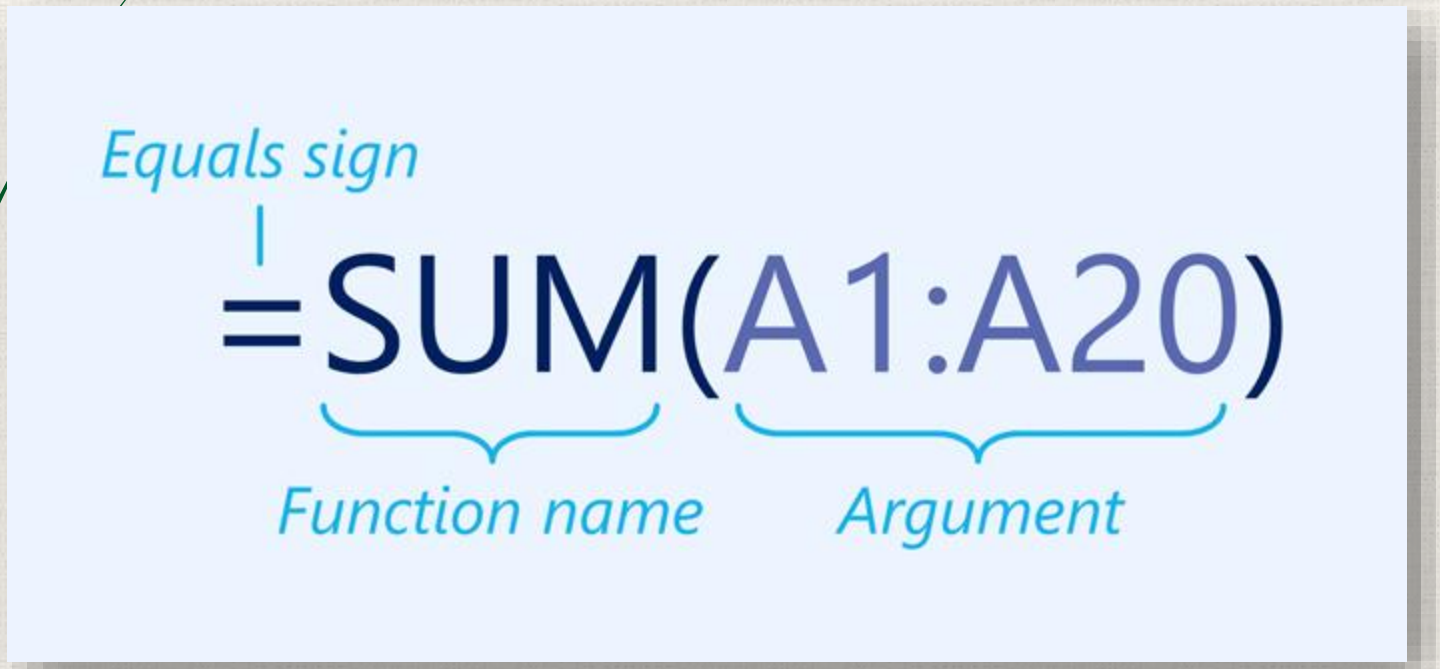
- Introduction
  - The parts of a function
  - Working with arguments
- Creating a Function
  - To create a function using the AutoSum command
  - To enter a function manually
- The Function Library
  - To insert a function from the Function Library
- The Insert Function Command
  - To use the Insert Function command
- Practice

Introduction

- A **function** is a **predefined formula** that performs calculations using specific values in a particular order. Excel includes many common functions that can be used to quickly find the **sum**, **average**, **count**, **maximum value**, and **minimum value** for a range of cells. In order to use functions correctly, you'll need to understand the different **parts of a function** and how to create **arguments** to calculate values and cell references.

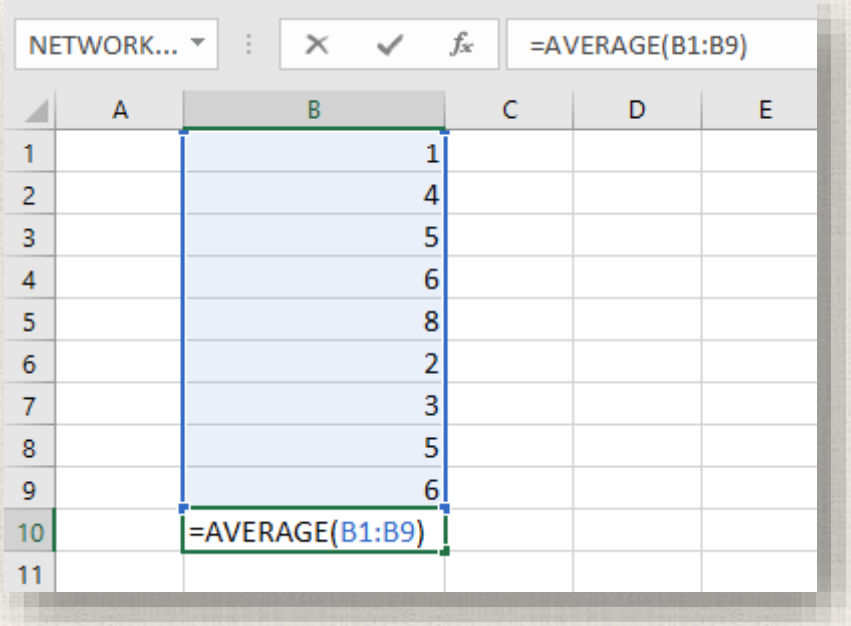
The Parts of a Function

- In order to work correctly, a function must be written a specific way, which is called the **syntax**. The basic syntax for a function is the **equals sign (=)**, the **function name** (SUM, for example), and one or more **arguments**. Arguments contain the information you want to calculate. The function in the example below would add the values of the cell range A1:A20.



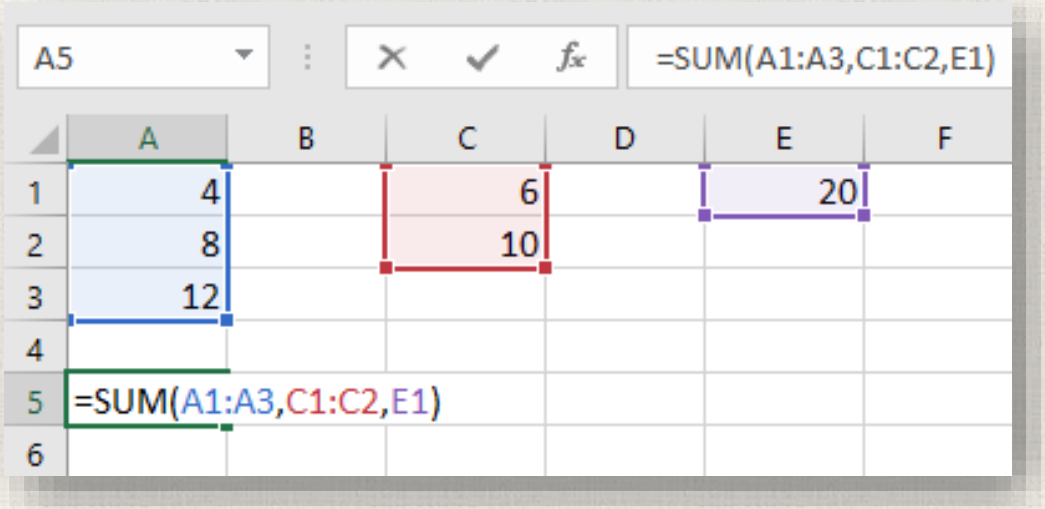
Working with arguments

- Arguments can refer to both **individual cells** and **cell ranges** and must be enclosed within **parentheses**. You can include one argument or multiple arguments, depending on the syntax required for the function.
- For example, the function **=AVERAGE(B1:B9)** would calculate the **average** of the values in the cell range B1:B9. This function contains only one argument.



Working with arguments

- Multiple arguments must be separated by a **comma**. For example, the function **=SUM(A1:A3, C1:C2, E1)** will **add** the values of all of the cells in the three arguments.



Creating a Function

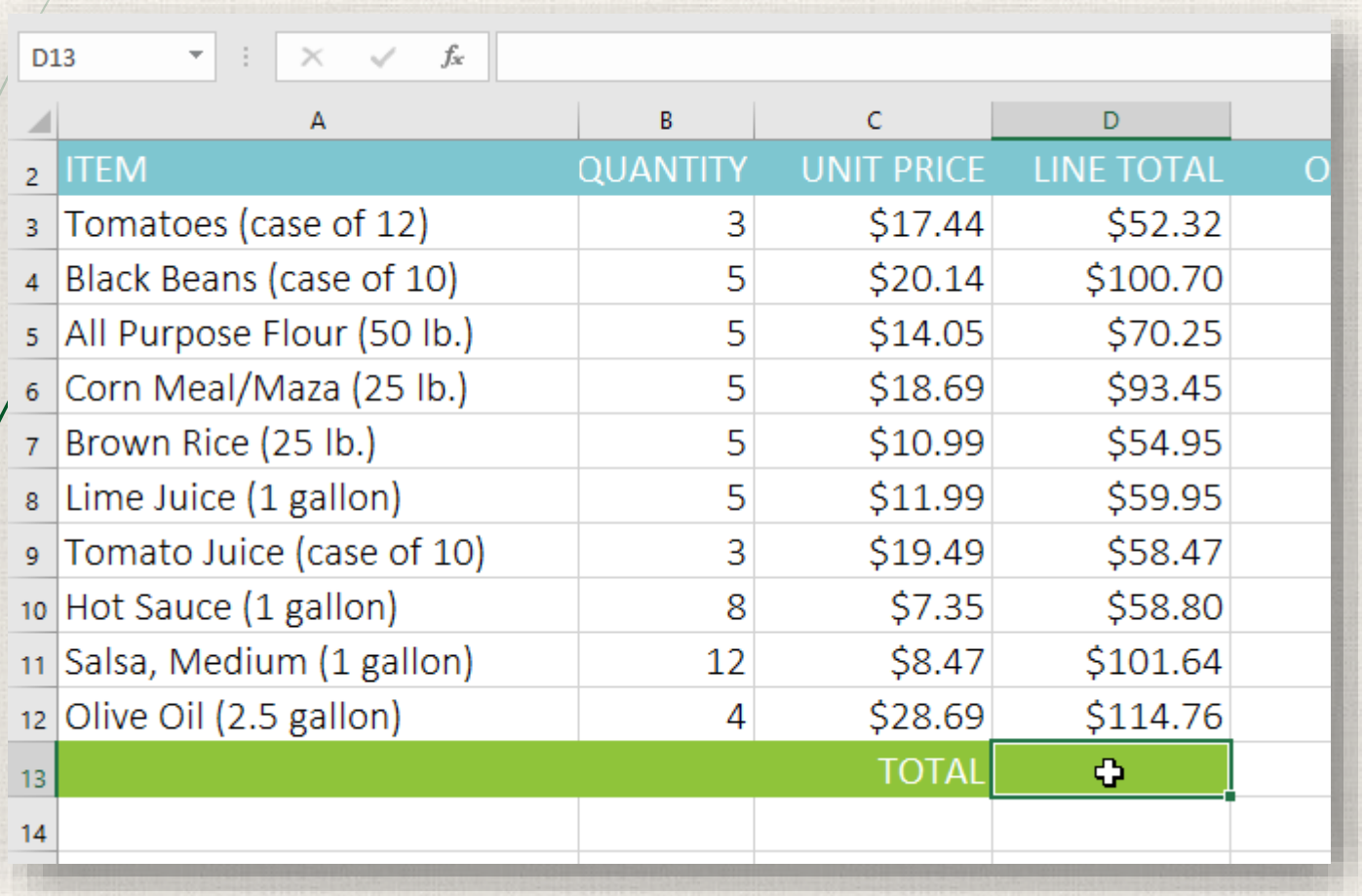
- There are a variety of functions available in Excel. Here are some of the most common functions you'll use:
  - **SUM**: This function **adds** all of the values of the cells in the argument.
  - **AVERAGE**: This function determines the **average** of the values included in the argument. It calculates the sum of the cells and then divides that value by the number of cells in the argument.
  - **COUNT**: This function **counts** the number of cells with numerical data in the argument. This function is useful for quickly counting items in a cell range.
  - **MAX**: This function determines the **highest cell value** included in the argument.
  - **MIN**: This function determines the **lowest cell value** included in the argument.

Creating a function using the AutoSum command

- The **AutoSum** command allows you to automatically insert the most common functions into your formula, including SUM, AVERAGE, COUNT, MIN, and MAX. In the example below, we'll use the **SUM** function to calculate the **total cost** for a list of recently ordered items.

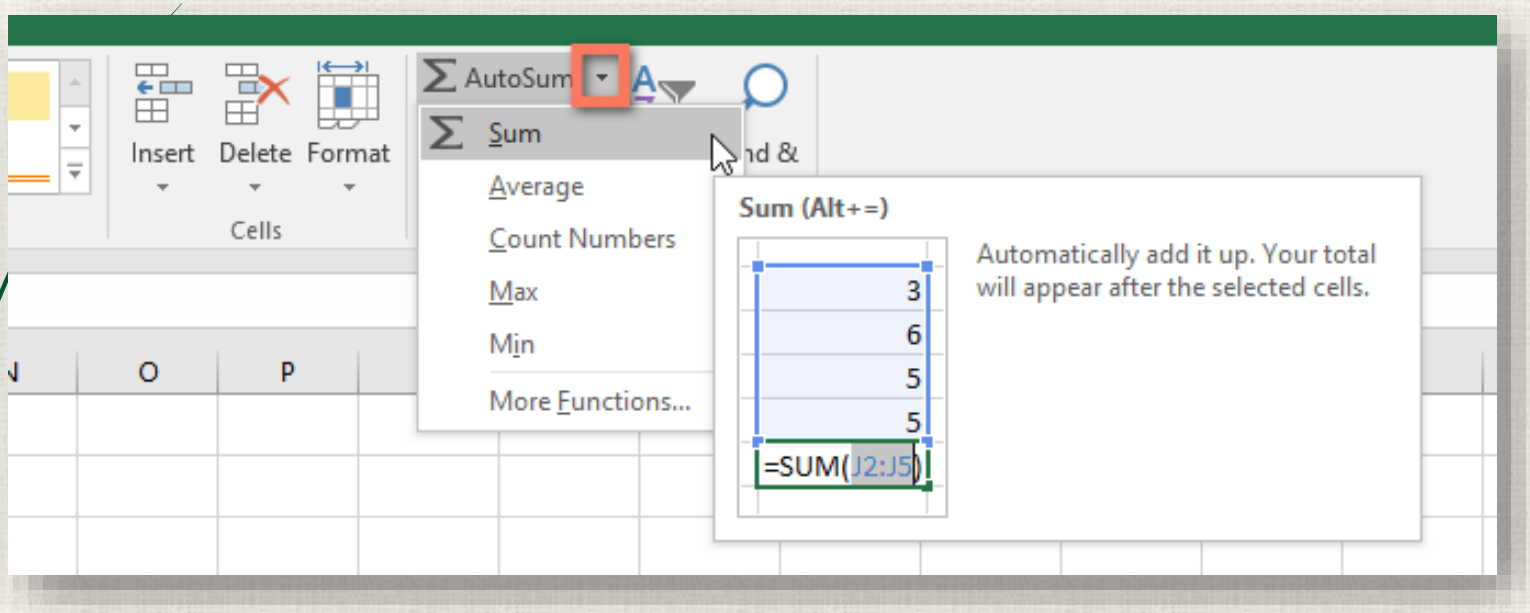
Creating a function using the AutoSum command

1. Select the **cell** that will contain the function. In our example, we'll select cell **D13**.



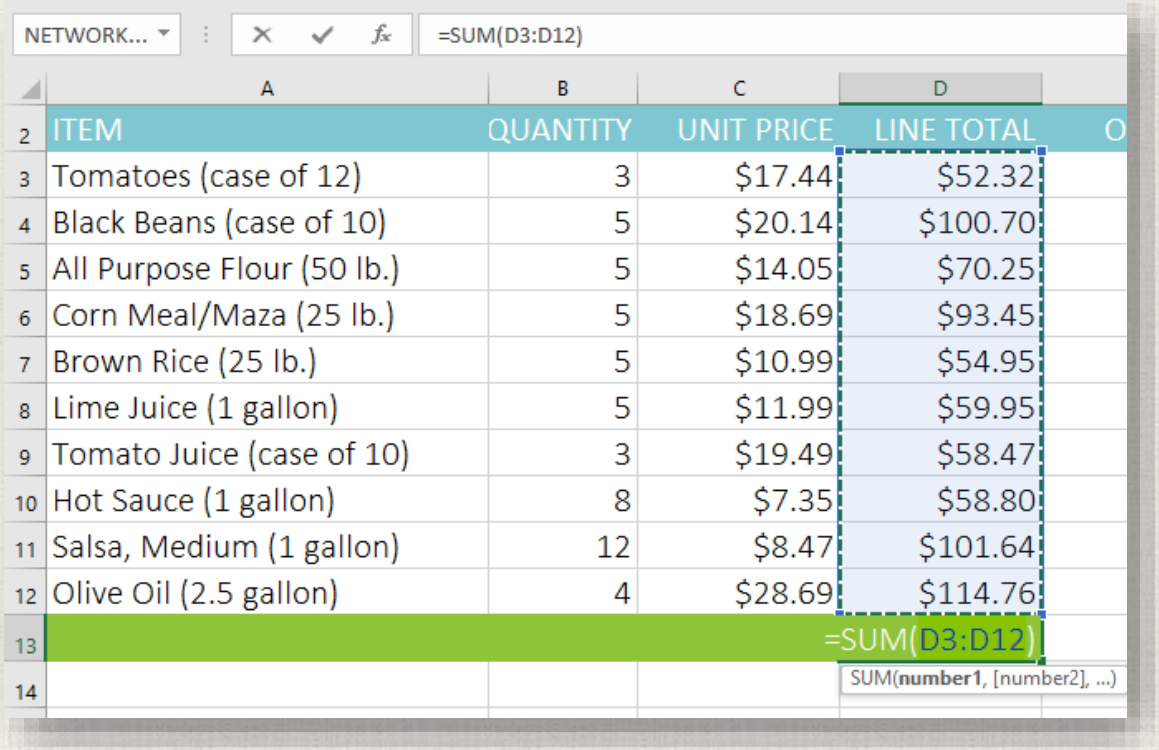
Creating a function using the AutoSum command

2. In the **Editing** group on the **Home** tab, click the **arrow** next to the **AutoSum** command. Next, choose the **desired function** from the drop-down menu. In our example, we'll select **Sum**.



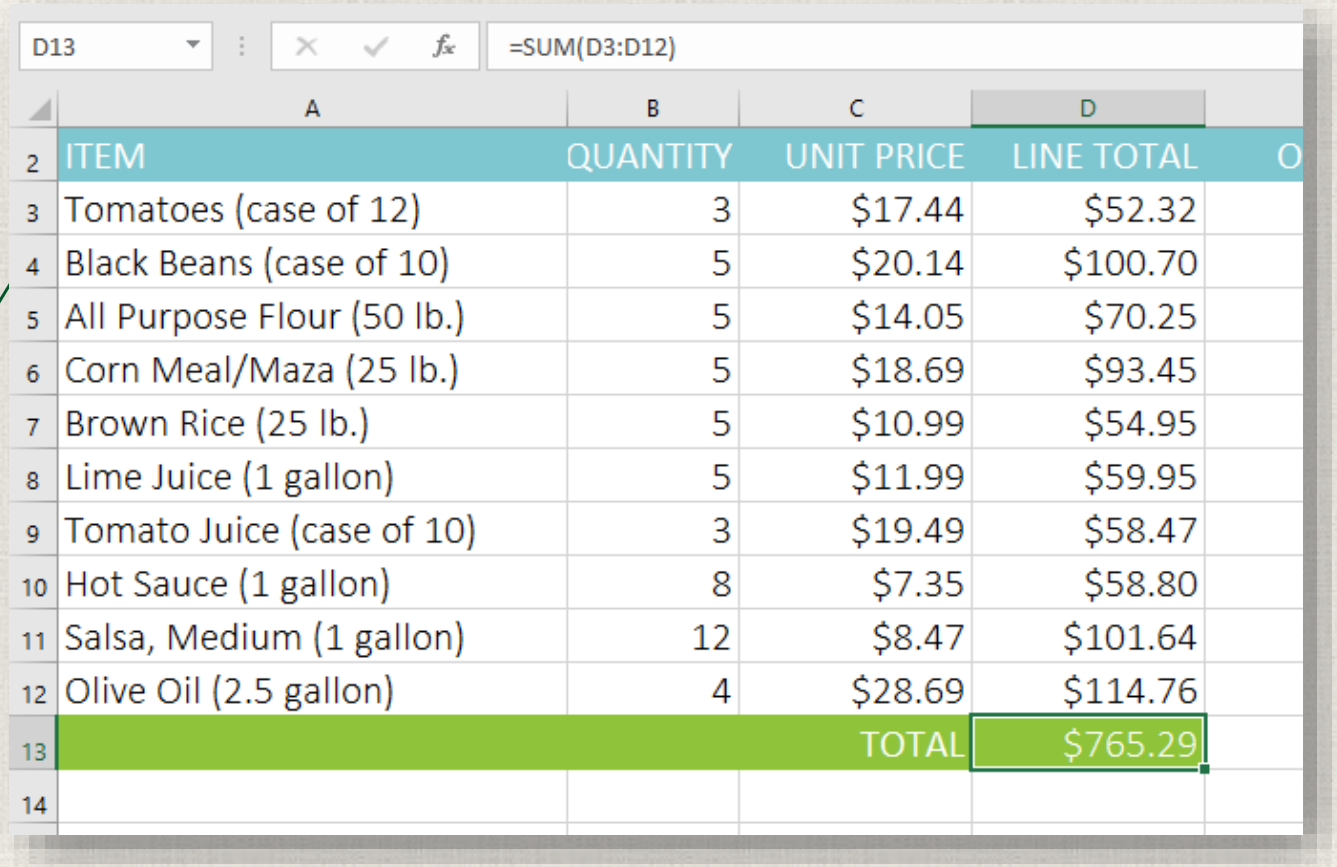
Creating a function using the AutoSum command

3. Excel will place the **function** in the cell and automatically select a **cell range** for the argument. In our example, cells **D3:D12** were selected automatically; their values will be **added** to calculate the total cost. If Excel selects the wrong cell range, you can manually enter the desired cells into the argument.



Creating a function using the AutoSum command

4. Press **Enter** on your keyboard. The function will be **calculated**, and the **result** will appear in the cell. In our example, the sum of D3:D12 is **$765.29**.

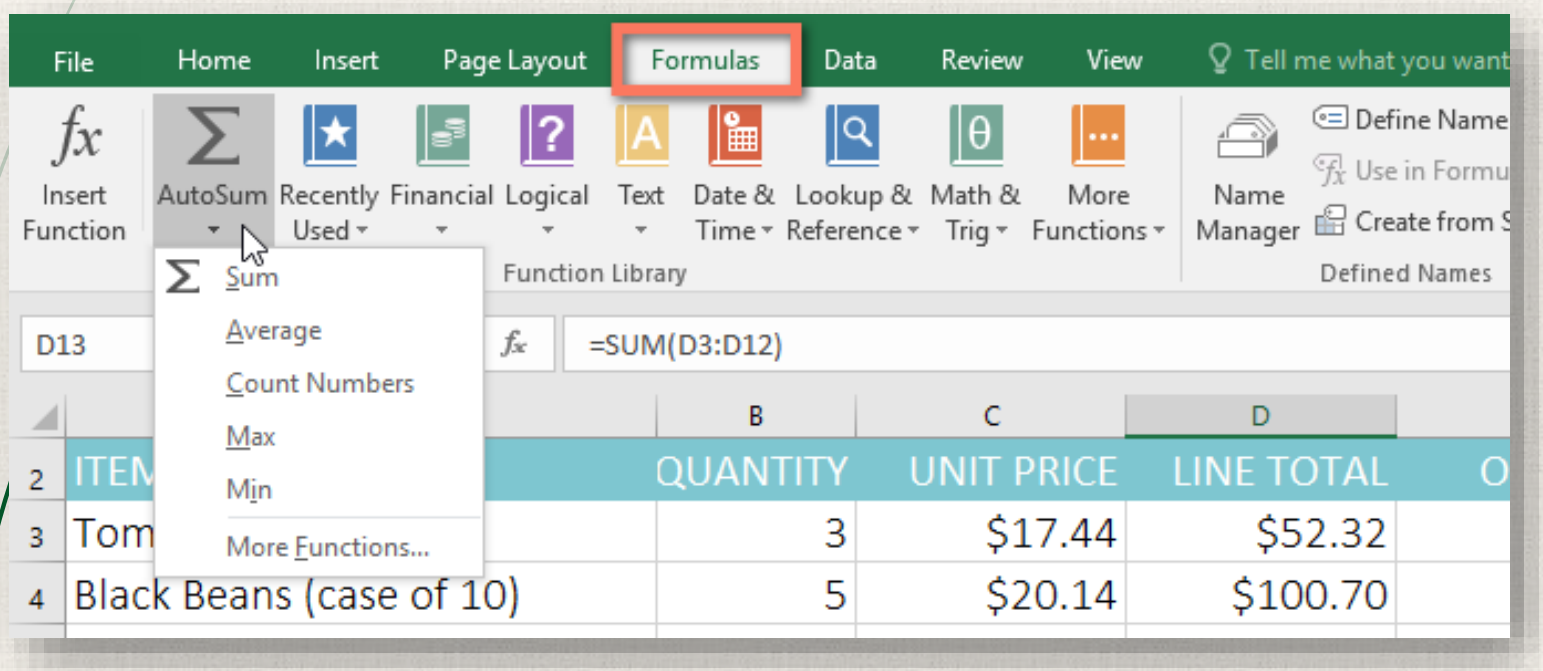


D13 =SUM(D3:D12)

	A	B	C	D
2	ITEM	QUANTITY	UNIT PRICE	LINE TOTAL
3	Tomatoes (case of 12)	3	$17.44	$52.32
4	Black Beans (case of 10)	5	$20.14	$100.70
5	All Purpose Flour (50 lb.)	5	$14.05	$70.25
6	Corn Meal/Maza (25 lb.)	5	$18.69	$93.45
7	Brown Rice (25 lb.)	5	$10.99	$54.95
8	Lime Juice (1 gallon)	5	$11.99	$59.95
9	Tomato Juice (case of 10)	3	$19.49	$58.47
10	Hot Sauce (1 gallon)	8	$7.35	$58.80
11	Salsa, Medium (1 gallon)	12	$8.47	$101.64
12	Olive Oil (2.5 gallon)	4	$28.69	$114.76
13			TOTAL	$765.29
14				

Creating a function using the AutoSum command

- The **AutoSum** command can also be accessed from the **Formulas** tab on the **Ribbon**.

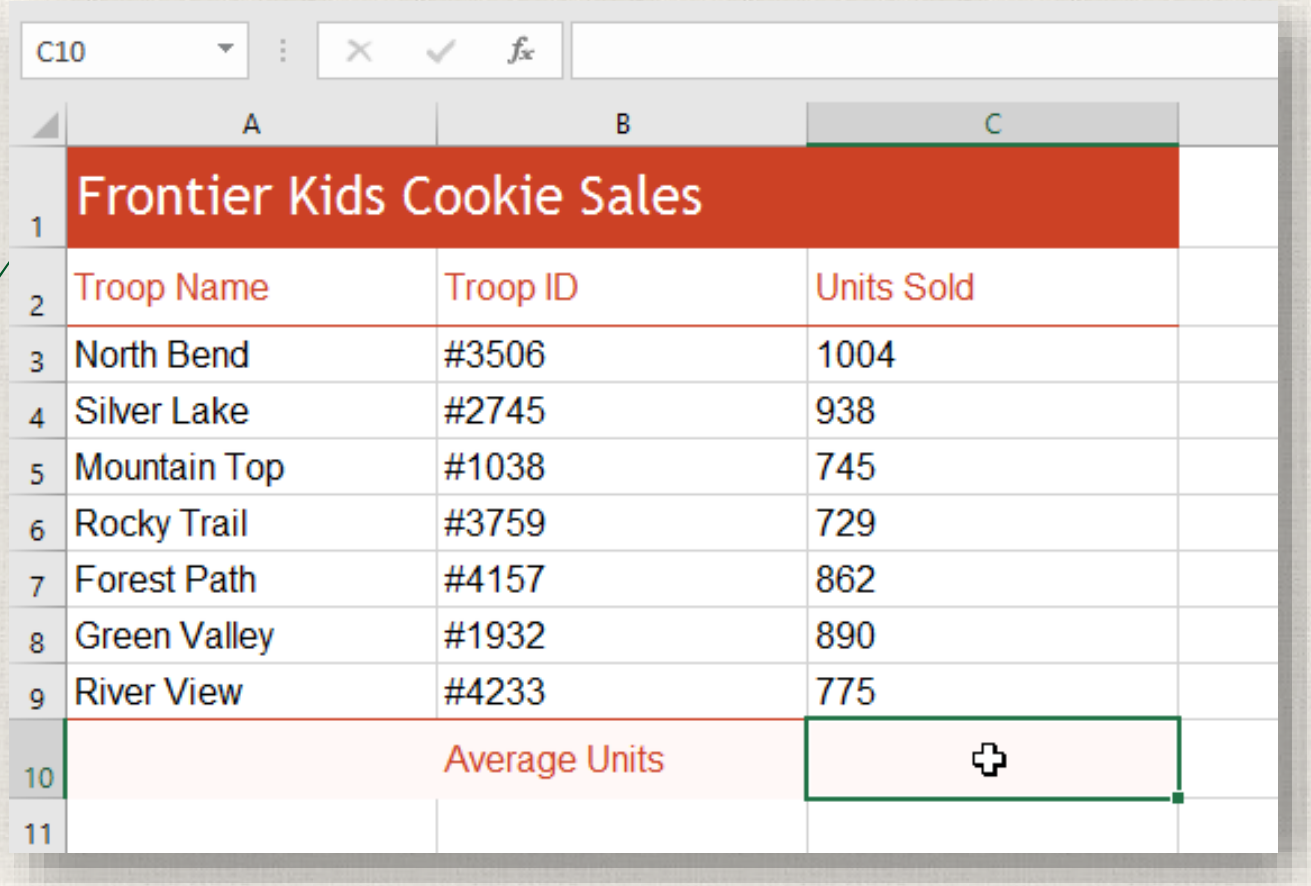


Entering a function manually

- If you already know the function name, you can easily type it yourself. In the example below (a tally of cookie sales), we'll use the **AVERAGE** function to calculate the **average number of units sold** by each troop.

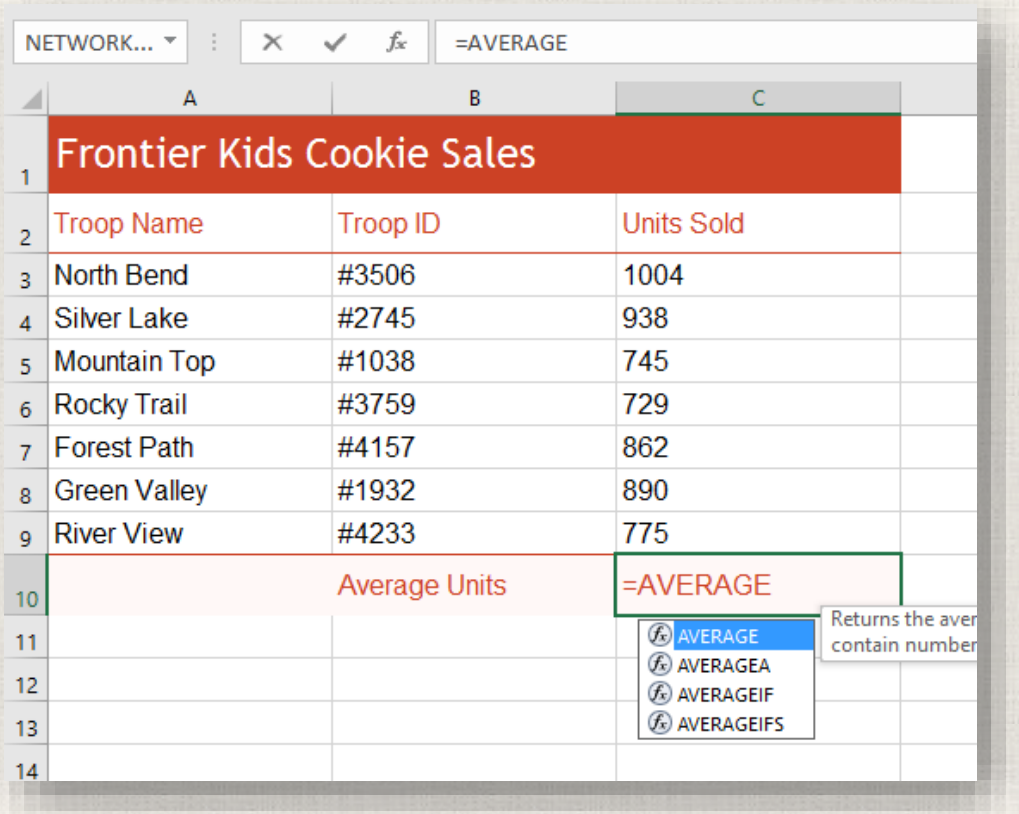
Entering a function manually

1. Select the **cell** that will contain the function. In our example, we'll select cell **C10**.



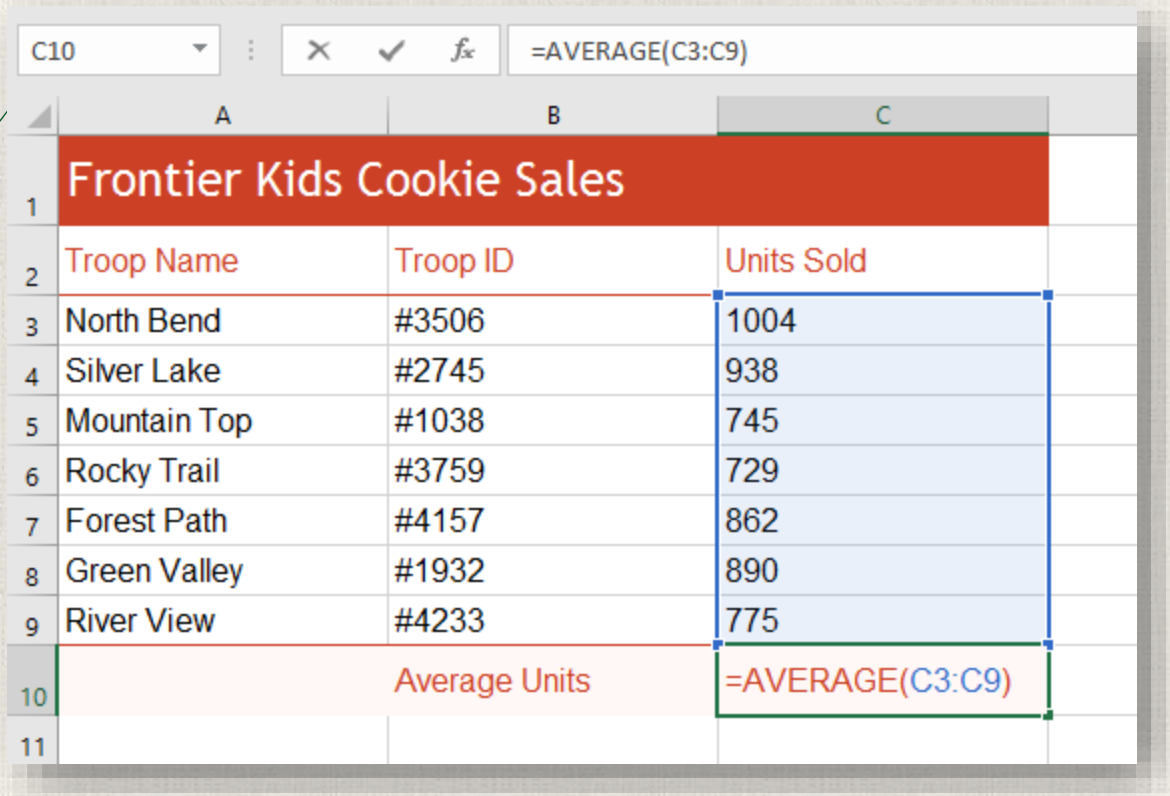
Entering a function manually

2. Type the **equals sign (=)**, and enter the desired **function name**. You can also select the desired function from the list of **suggested functions** that appears below the cell as you type. In our example, we'll type **=AVERAGE**.



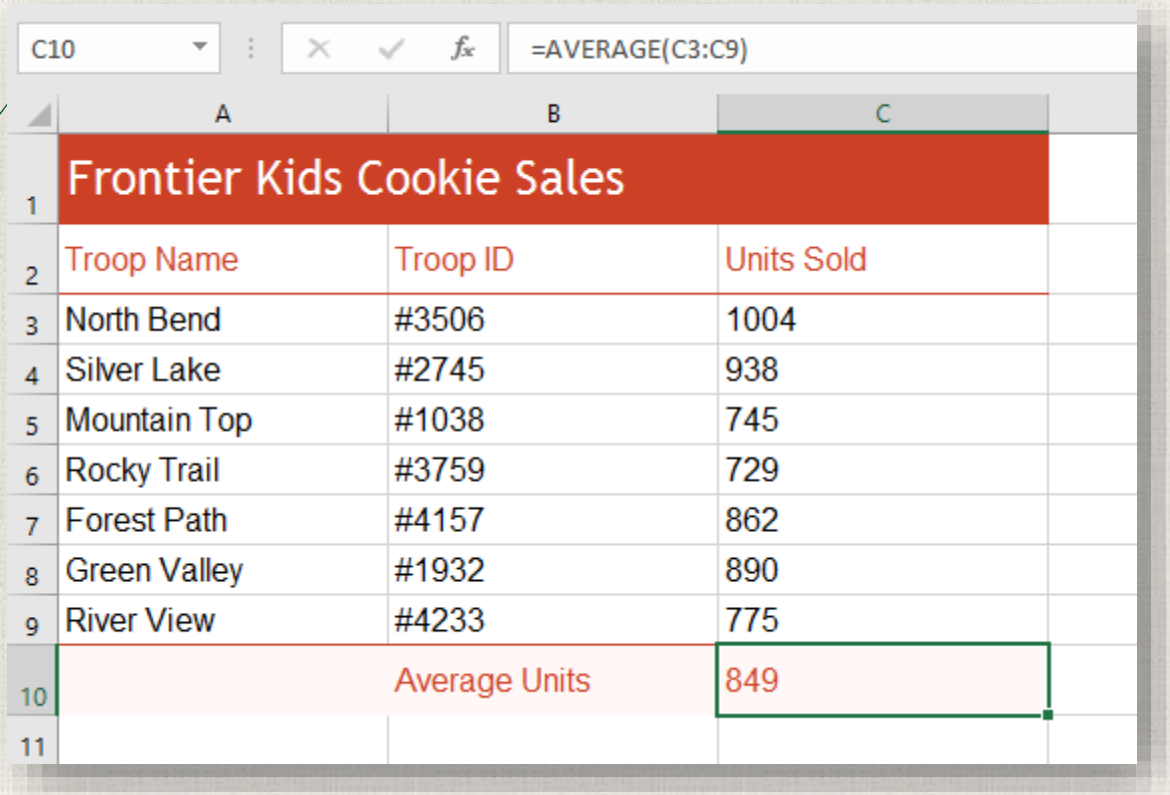
Entering a function manually

3. Enter the **cell range** for the argument inside **parentheses**. In our example, we'll type **(C3:C9)**. This formula will add the values of cells C3:C9, then divide that value by the total number of values in the range.



Entering a function manually

4. Press **Enter** on your keyboard. The function will be calculated, and the **result** will appear in the cell. In our example, the average number of units sold by each troop is **849**.



	A	B	C
1	Frontier Kids Cookie Sales		
2	Troop Name	Troop ID	Units Sold
3	North Bend	#3506	1004
4	Silver Lake	#2745	938
5	Mountain Top	#1038	745
6	Rocky Trail	#3759	729
7	Forest Path	#4157	862
8	Green Valley	#1932	890
9	River View	#4233	775
10		Average Units	849
11			

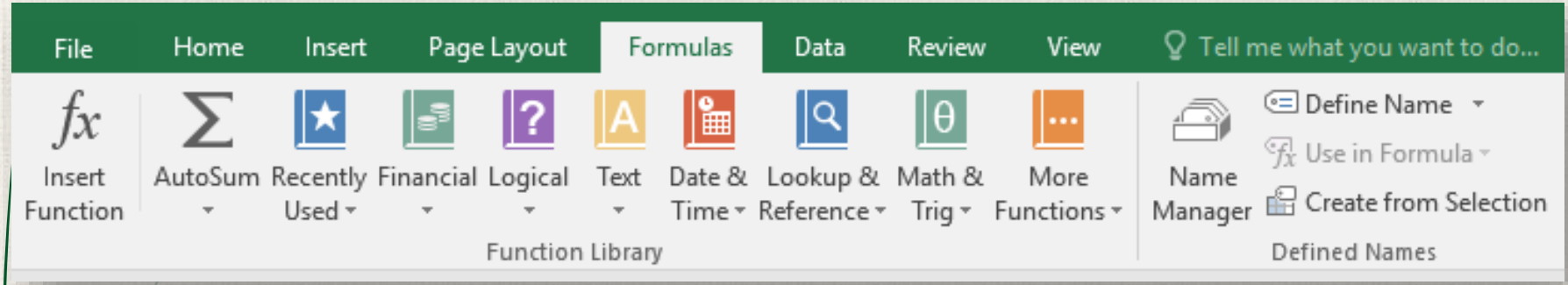
Warning

- Excel **will not always tell you** if your formula contains an error, so it's up to you to check all of your formulas.

The Function Library

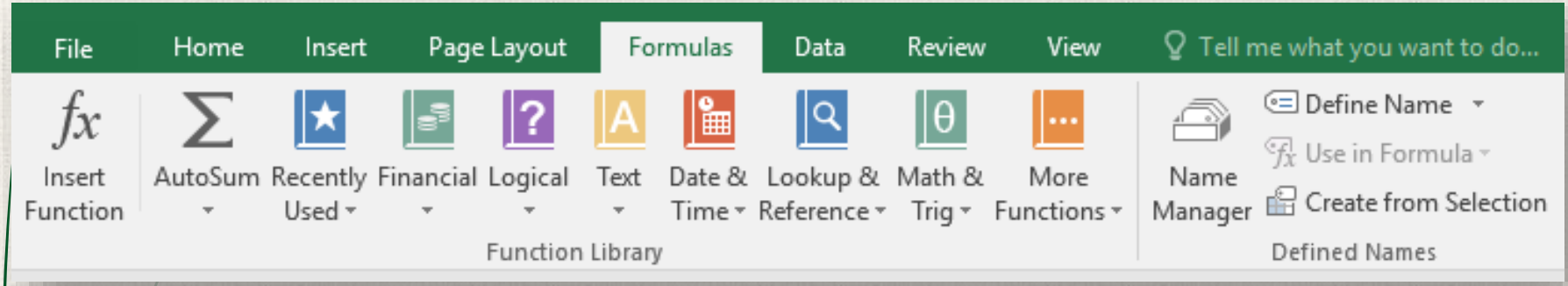
- While there are hundreds of functions in Excel, the ones you'll use the most will depend on the **type of data** your workbooks contain. There's no need to learn every single function, but exploring some of the different **types** of functions will help you as you create new projects. You can even use the **Function Library** on the **Formulas** tab to browse functions by category, such as **Financial, Logical, Text**, and **Date & Time**.
- To access the **Function Library**, select the **Formulas** tab on the **Ribbon**. Look for the **Function Library** group.

The Function Library



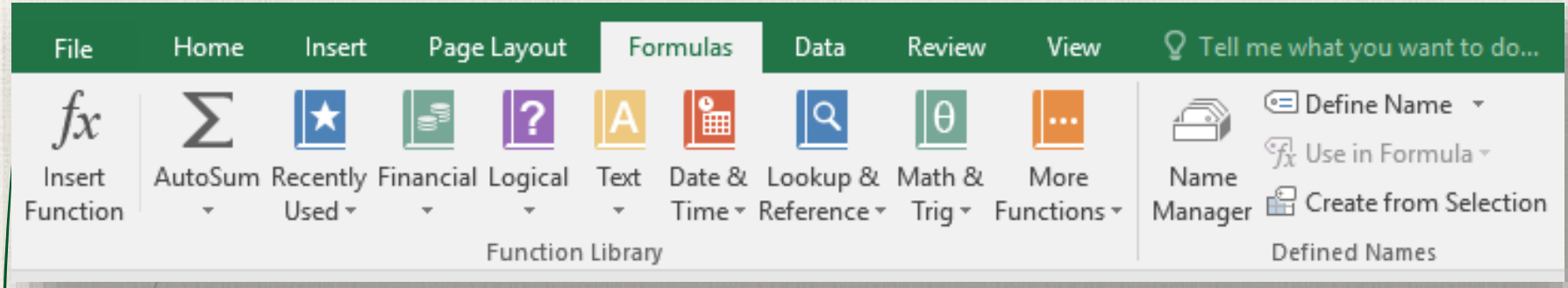
- **Insert Function :** If you're having trouble finding the right function, the Insert Function command allows you to search for functions using keywords.
- **AutoSum Command :** The AutoSum command allows you to automatically return results for common functions, like SUM, AVERAGE, and COUNT.
- **Recently Used :** The Recently Used command gives you access to functions you've recently worked with.
- **Financial :** The Financial category contains functions for financial calculations like determining a payment (PMT) or interest rate for a loan (RATE).

The Function Library



- **Logical** : Functions in the Logical category check arguments for a value or condition. For example, if an order is more than $50, add $4.99 for shipping; if it is more than $100, do not charge for shipping (IF).
- **Text :** The Text category contains functions that work with the text in arguments to perform tasks, such as converting text to lowercase (LOWER) or replacing text (REPLACE).
- **Date & Time :** The Date & Time category contains functions for working with dates and time and will return results like the current date and time (NOW) or the seconds (SECOND).

The Function Library

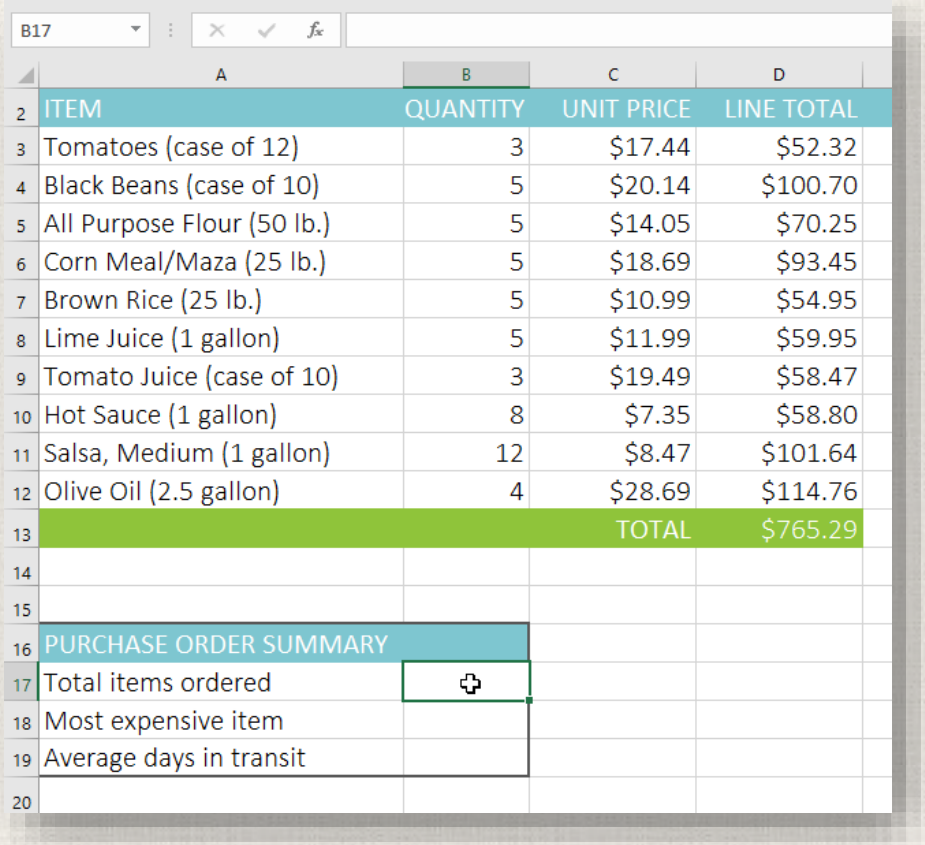


- **Lookup & Reference :** The Lookup & Reference category contains functions that will return results for finding and referencing information. For example, you can add a hyperlink to a cell (HYPERLINK) or return the value of a particular row and column intersection (INDEX).
- **Math & Trig :** The Math & Trig category includes functions for numerical arguments. For example, you can round values (ROUND), find the value of Pi (PI), multiply (PRODUCT), and subtotal (SUBTOTAL).
- **More Functions :** More Functions contains additional functions under categories for Statistical, Engineering, Cube, Information, and Compatibility.

To insert a function from the Function Library

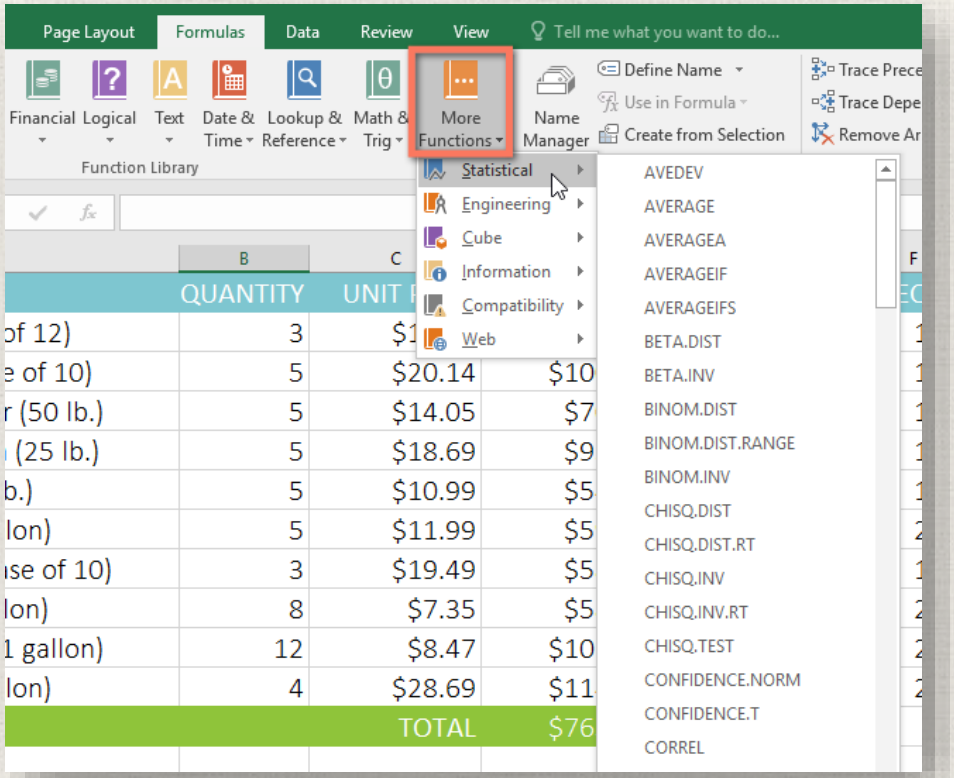
- In the example below, we'll use the COUNTA function to count the total number of items in the **Items** column. Unlike COUNT, **COUNTA** can be used to tally cells that contain data of any kind, not just numerical data.

1. Select the **cell** that will contain the function. In our example, we'll select cell **B17**.



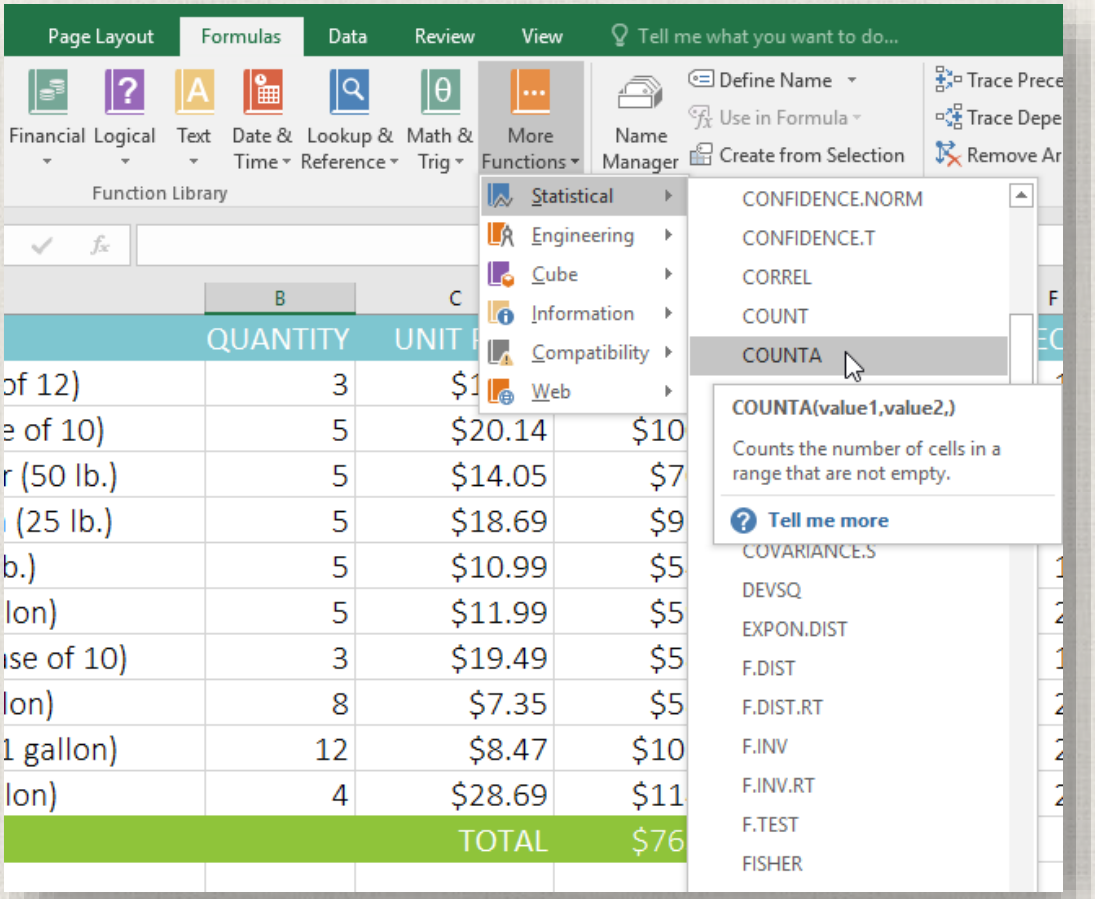
To insert a function from the Function Library

2. Click the **Formulas** tab on the **Ribbon** to access the **Function Library**.
3. From the **Function Library** group, select the desired **function category**. In our example, we'll choose **More Functions**, then hover the mouse over **Statistical**.



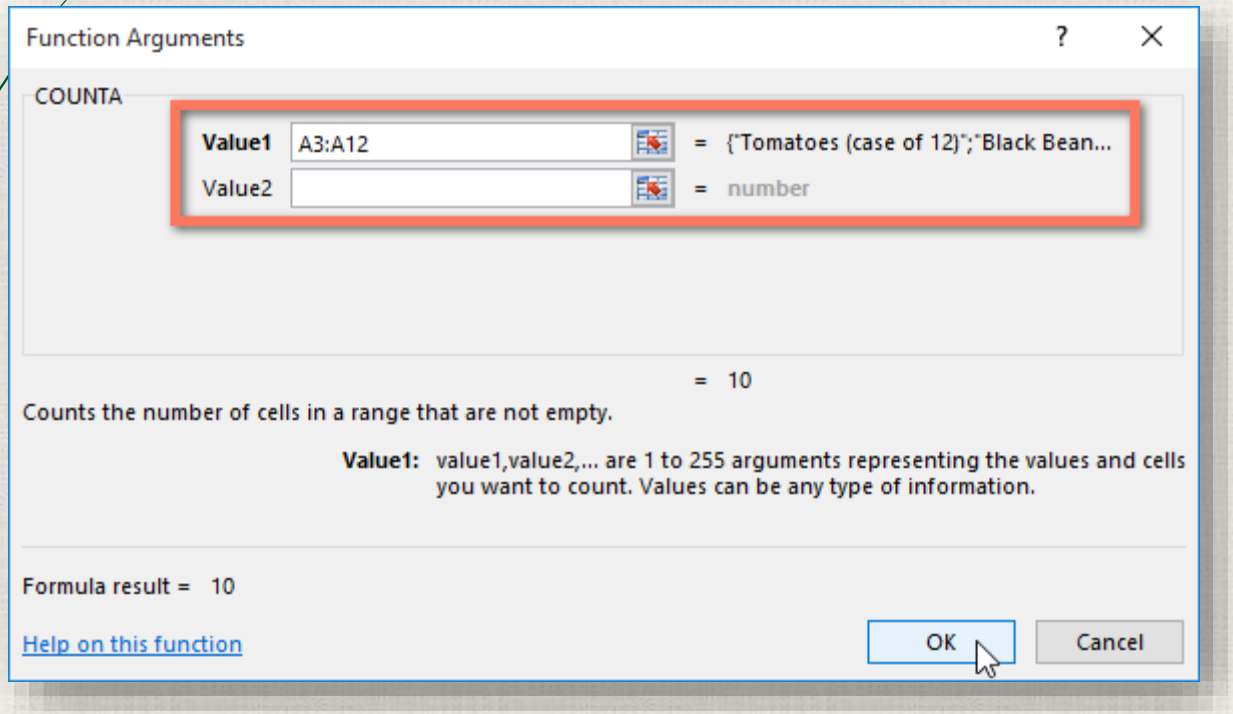
To insert a function from the Function Library

4. Select the **desired function** from the drop-down menu. In our example, we'll select the **COUNTA** function, which will count the number of cells in the **Items** column that are not empty.



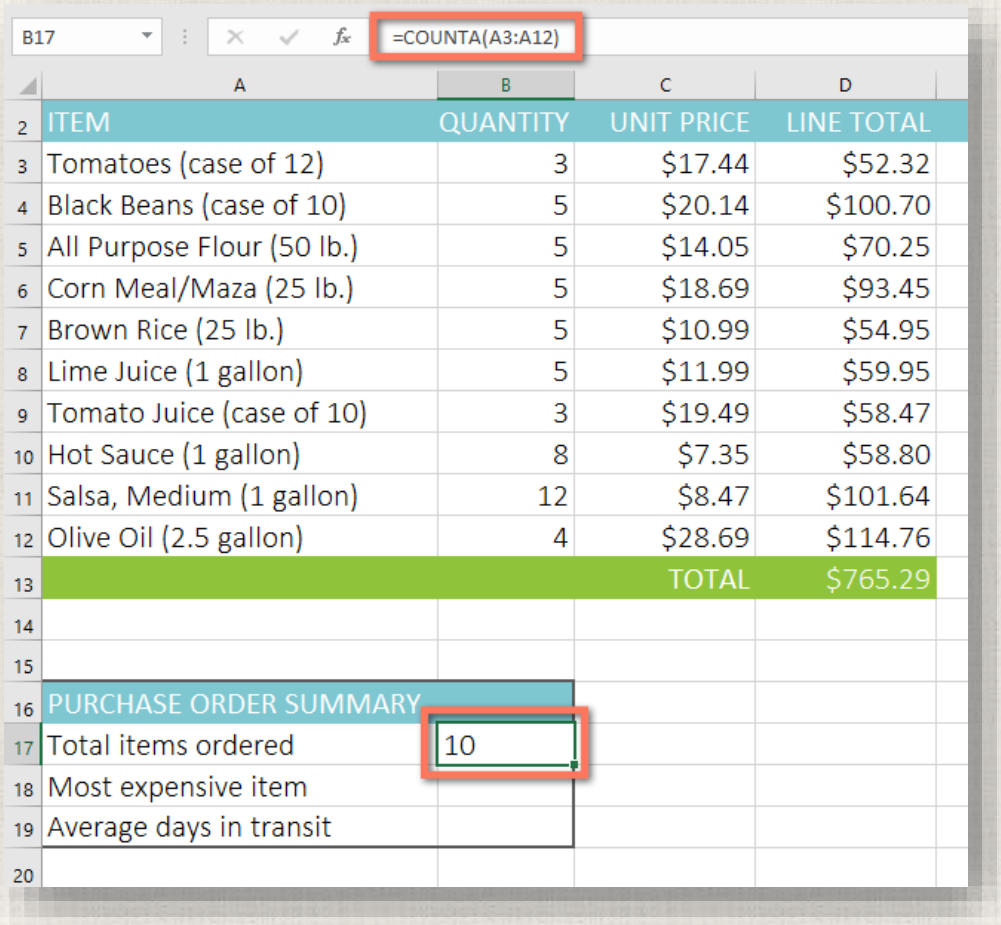
To insert a function from the Function Library

5. The **Function Arguments** dialog box will appear. Select the **Value1** field, then enter or select the desired cells. In our example, we'll enter the cell range **A3:A12**. You may continue to add arguments in the **Value2** field, but in this case we only want to count the number of cells in the cell range **A3:A12**.
6. When you're satisfied, click **OK**.



To insert a function from the Function Library

7. The function will be **calculated**, and the **result** will appear in the cell. In our example, the result shows that a total of **10 items** were ordered.



	A	B	C	D
2	ITEM	QUANTITY	UNIT PRICE	LINE TOTAL
3	Tomatoes (case of 12)	3	$17.44	$52.32
4	Black Beans (case of 10)	5	$20.14	$100.70
5	All Purpose Flour (50 lb.)	5	$14.05	$70.25
6	Corn Meal/Maza (25 lb.)	5	$18.69	$93.45
7	Brown Rice (25 lb.)	5	$10.99	$54.95
8	Lime Juice (1 gallon)	5	$11.99	$59.95
9	Tomato Juice (case of 10)	3	$19.49	$58.47
10	Hot Sauce (1 gallon)	8	$7.35	$58.80
11	Salsa, Medium (1 gallon)	12	$8.47	$101.64
12	Olive Oil (2.5 gallon)	4	$28.69	$114.76
13			TOTAL	$765.29
14				
15				
16	PURCHASE ORDER SUMMARY			
17	Total items ordered	10		
18	Most expensive item			
19	Average days in transit			
20				

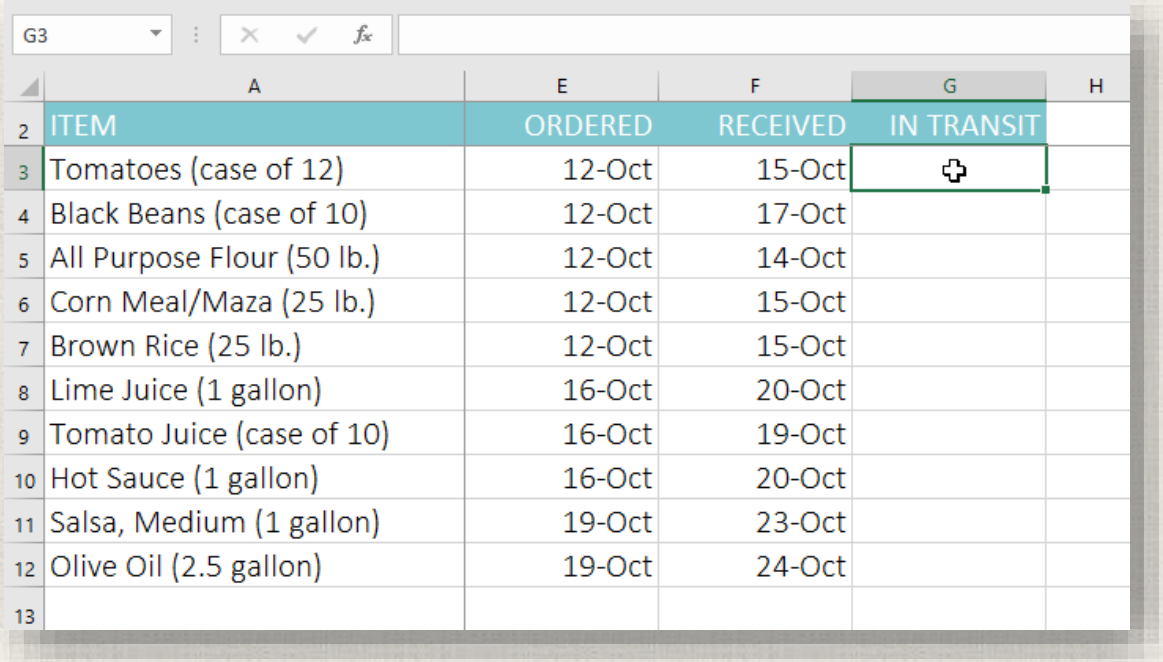
The Insert Function command

- While the Function Library is a great place to browse for functions, sometimes you may prefer to **search** for one instead. You can do so using the **Insert Function** command. It may take some trial and error depending on the type of function you're looking for; however, with practice, the Insert Function command can be a powerful way to find a function quickly.

Using the Insert Function command:

- In the example below, we want to find a function that will calculate the **number of business days** it took to receive items after they were ordered. We'll use the dates in columns **E** and **F** to calculate the delivery time in column **G**.

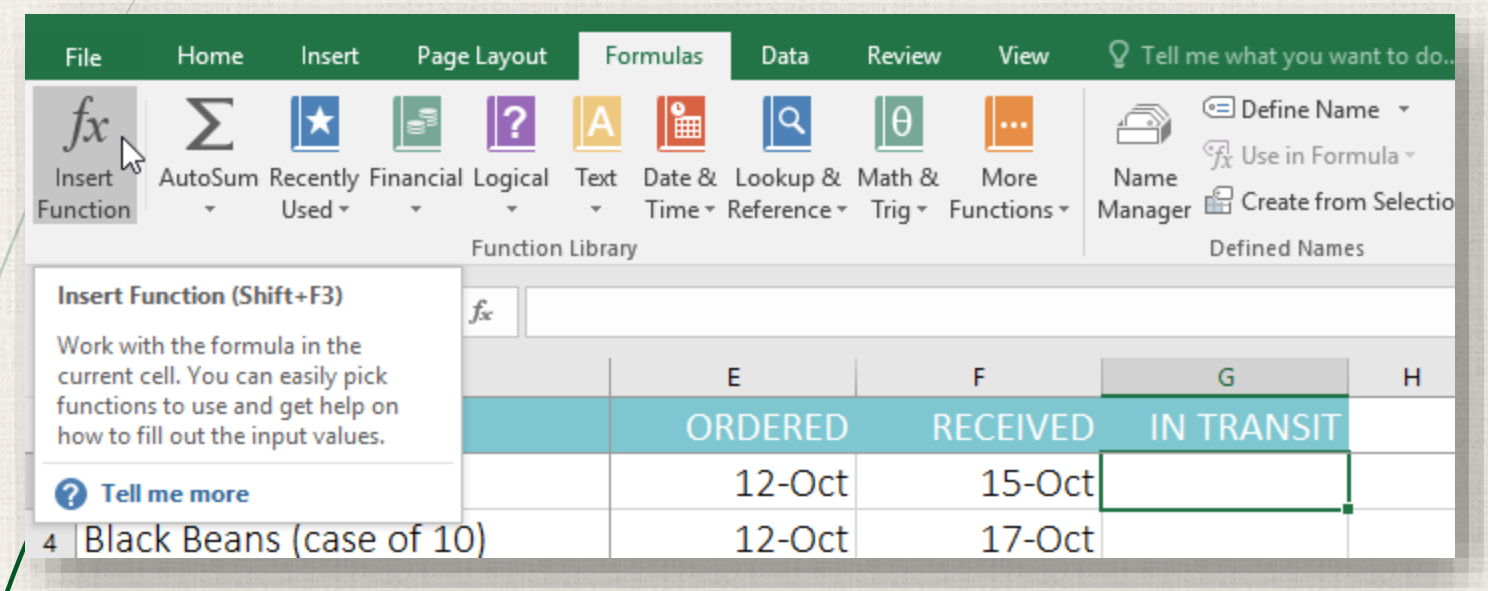
1. Select the **cell** that will contain the function. In our example, we'll select cell **G3**.



	A	E	F	G	H
2	ITEM	ORDERED	RECEIVED	IN TRANSIT	
3	Tomatoes (case of 12)	12-Oct	15-Oct		
4	Black Beans (case of 10)	12-Oct	17-Oct		
5	All Purpose Flour (50 lb.)	12-Oct	14-Oct		
6	Corn Meal/Maza (25 lb.)	12-Oct	15-Oct		
7	Brown Rice (25 lb.)	12-Oct	15-Oct		
8	Lime Juice (1 gallon)	16-Oct	20-Oct		
9	Tomato Juice (case of 10)	16-Oct	19-Oct		
10	Hot Sauce (1 gallon)	16-Oct	20-Oct		
11	Salsa, Medium (1 gallon)	19-Oct	23-Oct		
12	Olive Oil (2.5 gallon)	19-Oct	24-Oct		
13					

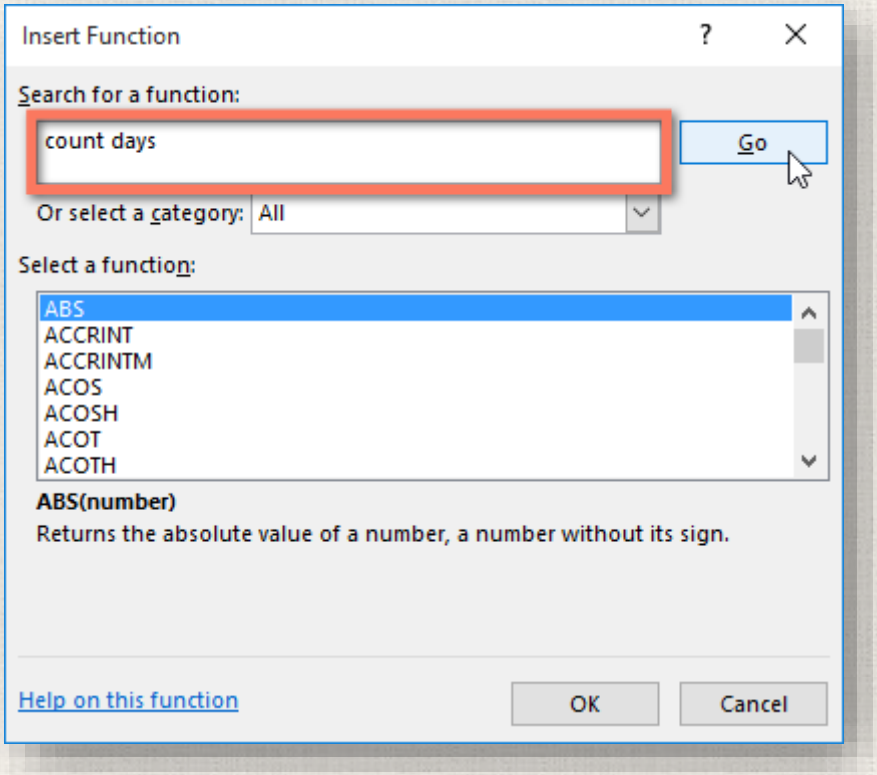
Using the Insert Function command:

2. Click the **Formulas** tab on the **Ribbon**, then click the **Insert Function** command.



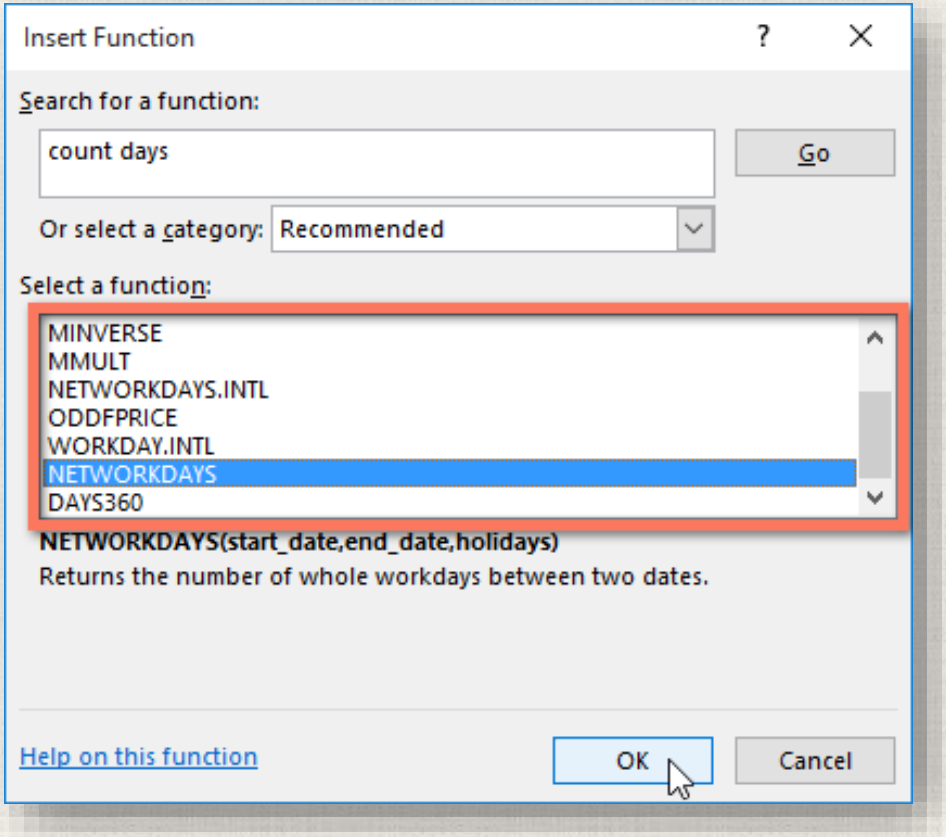
Using the Insert Function command:

3. The **Insert Function dialog** box will appear.
4. Type a few **keywords** describing the calculation you want the function to perform, then click **Go**. In our example, we'll type **count days**, but you can also search by selecting a **category** from the drop-down list.



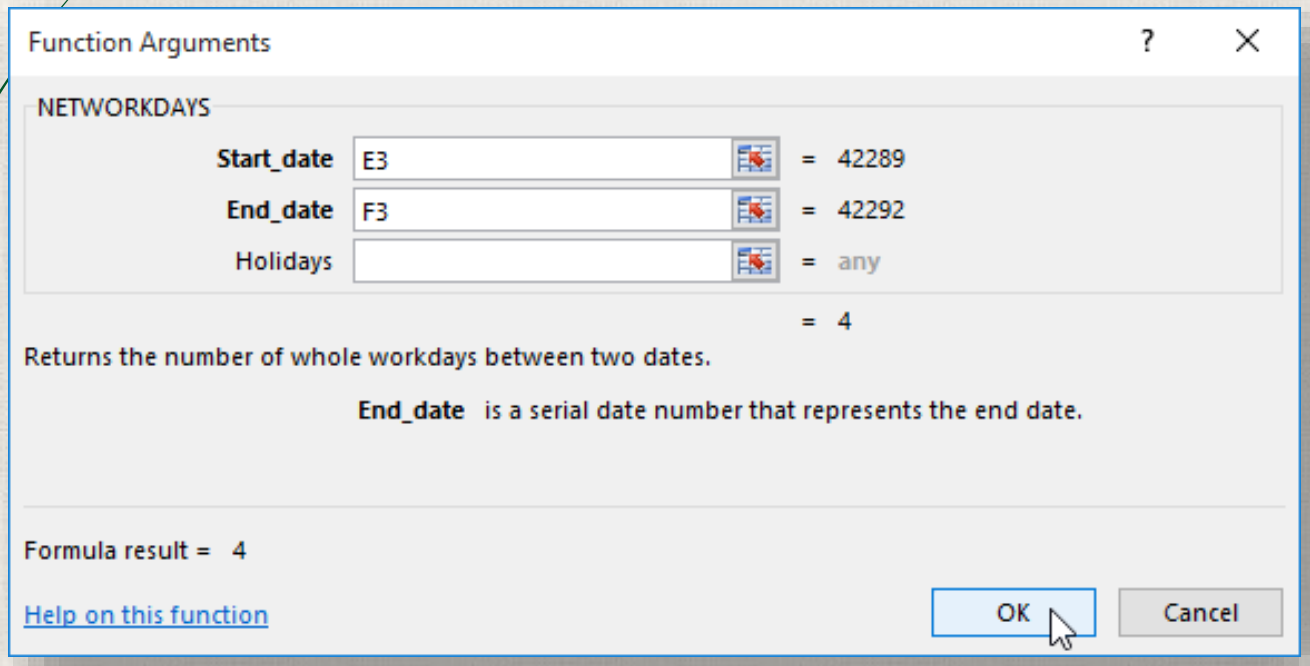
Using the Insert Function command:

5. Review the **results** to find the desired function, then click **OK**. In our example, we'll choose **NETWORKDAYS**, which will count the number of business days between the ordered date and received date.



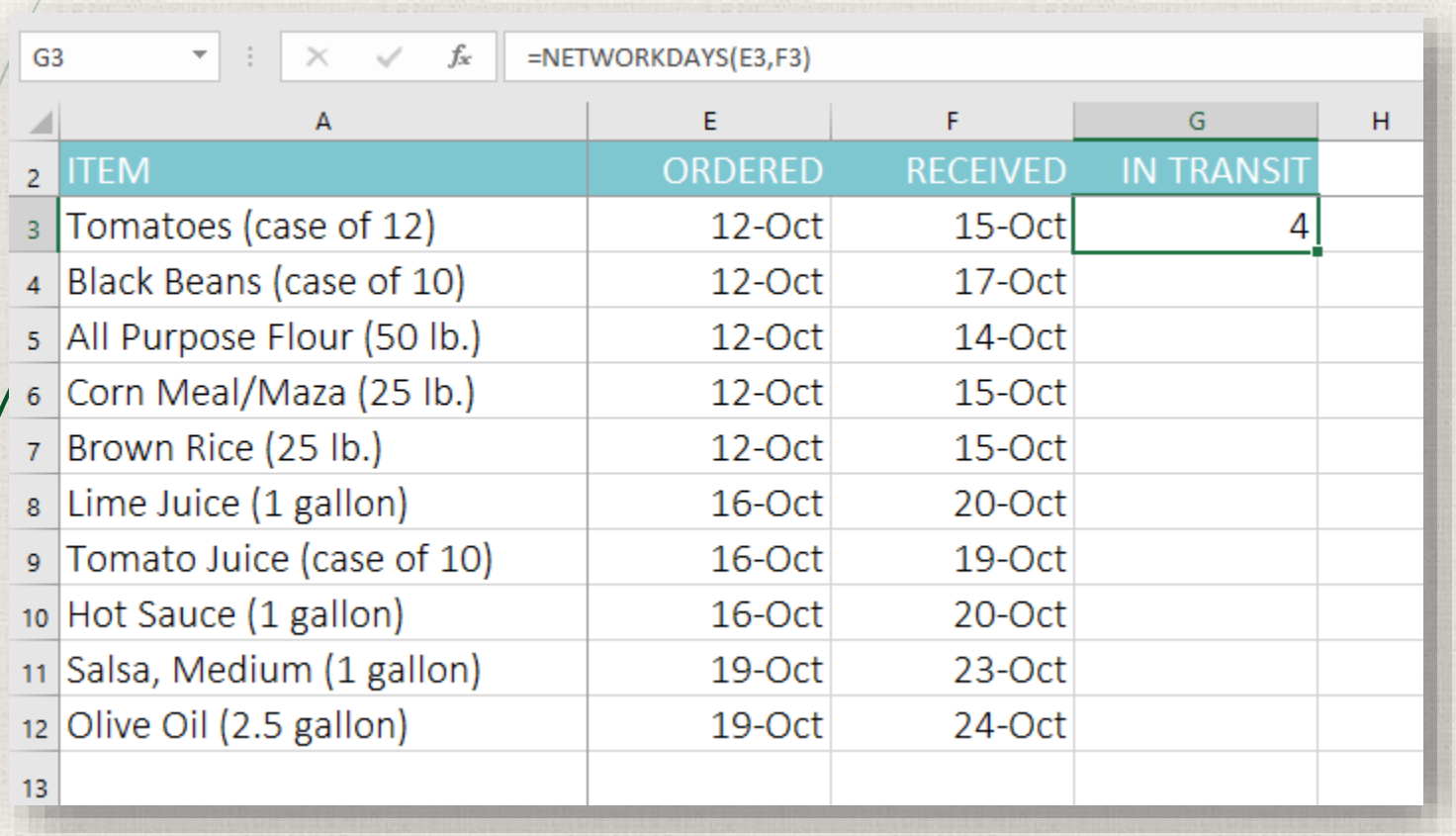
Using the Insert Function command:

6. The **Function Arguments** dialog box will appear. From here, you'll be able to enter or select the cells that will make up the arguments in the function. In our example, we'll enter **E3** in the **Start_date** field and **F3** in the **End_date** field.
7. When you're satisfied, click **OK**.



Using the Insert Function command:

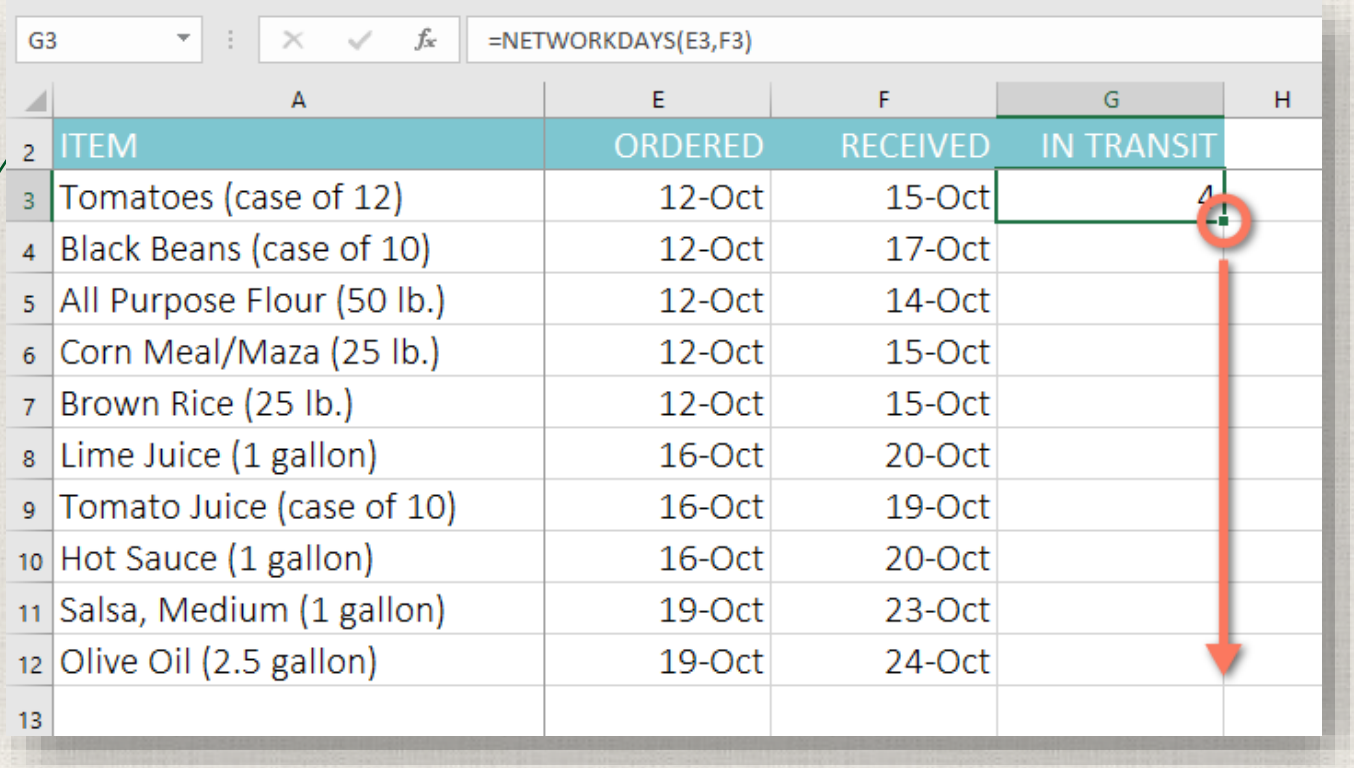
8. The function will be **calculated**, and the **result** will appear in the cell. In our example, the result shows that it took **four business days** to receive the order.



	A	E	F	G	H
2	ITEM	ORDERED	RECEIVED	IN TRANSIT	
3	Tomatoes (case of 12)	12-Oct	15-Oct	4	
4	Black Beans (case of 10)	12-Oct	17-Oct		
5	All Purpose Flour (50 lb.)	12-Oct	14-Oct		
6	Corn Meal/Maza (25 lb.)	12-Oct	15-Oct		
7	Brown Rice (25 lb.)	12-Oct	15-Oct		
8	Lime Juice (1 gallon)	16-Oct	20-Oct		
9	Tomato Juice (case of 10)	16-Oct	19-Oct		
10	Hot Sauce (1 gallon)	16-Oct	20-Oct		
11	Salsa, Medium (1 gallon)	19-Oct	23-Oct		
12	Olive Oil (2.5 gallon)	19-Oct	24-Oct		
13					

Using the Insert Function command:

- Like formulas, functions can be copied to adjacent cells. Simply select the **cell** that contains the function, then click and drag the **fill handle** over the cells you want to fill. The function will be copied, and values for those cells will be calculated relative to their rows or columns.



	A	E	F	G	H
2	ITEM	ORDERED	RECEIVED	IN TRANSIT	
3	Tomatoes (case of 12)	12-Oct	15-Oct	4	
4	Black Beans (case of 10)	12-Oct	17-Oct		
5	All Purpose Flour (50 lb.)	12-Oct	14-Oct		
6	Corn Meal/Maza (25 lb.)	12-Oct	15-Oct		
7	Brown Rice (25 lb.)	12-Oct	15-Oct		
8	Lime Juice (1 gallon)	16-Oct	20-Oct		
9	Tomato Juice (case of 10)	16-Oct	19-Oct		
10	Hot Sauce (1 gallon)	16-Oct	20-Oct		
11	Salsa, Medium (1 gallon)	19-Oct	23-Oct		
12	Olive Oil (2.5 gallon)	19-Oct	24-Oct		
13					

TEXT FUNCTIONS

TEXT FUNCTIONS

Function	Explanation	Example	Result
MID()	Returns the characters from the middle of a string, given a starting position and length. **MID**(text, start_num, num_chars)	=MID(E23,3,4)	rina
FIND()	Returns the starting position of one text string within another text string. **FIND** is case-sensitive. **FIND**(find_text, within_text, start_num)	=FIND("Daniel",E25)	8
LEN()	Returns the number of characters in a text string. **LEN**(text)	=LEN(E24)	9
LEFT()	Returns the specified number of characters from the start of a text string. **LEFT**(text, num_chars)	=LEFT(E24,4)	Lori
RIGHT()	Returns the specified number of characters from the end of a text string. **RIGHT**(text, num_chars)	=RIGHT(E24,4)	Best
PROPER()	Converts first letter of each text to uppercase. **PROPER**(text)	=PROPER(E23)	Karina Flores
CONCATENATE()	Joins several text strings into one text string. **CONCATENATE**(text1,text2,...)	=CONCATENATE(E26,F26)	Benjamin Richard
UPPER()	Converts text to uppercase. **UPPER**(text)	=UPPER(E26)	BENJAMIN
LOWER()	Converts text to lowercase. **LOWER**(text)	=LOWER(F26)	richard

karina flores

Lori Best

Martin Daniel Flowers

Benjamin Richard

Proper Function

- When you think about Excel functions, you probably think about performing calculations with numbers. While it's true that you can use functions to do lots of handy things with numbers in Excel, some functions can help you format text too.
- One good example is the **PROPER** function, which capitalizes **the first letter of every word** in a cell. If you have cells containing proper nouns, like names or titles, you can use the PROPER function to make sure everything is capitalized correctly.

Proper Function

- In the following worksheet, you can see that not everyone has been careful to capitalize the first and last names of the people they want to nominate, so the spreadsheet looks messy. You could go through the column and correct the names manually, but using the PROPER function will be faster and easier.

	A
1	**Students**
2	Richard newell
3	kevin Mann
4	gregory johnson
5	kristina Bridges
6	amy Coleman
7	sharyn houston
8	Lisa Kincade
9	joseph costello
10	John carpenter
11	william Franco
12	Sandra Stewart
13	dan Munoz
14	laura homan
15	terence Patterson

Proper Function

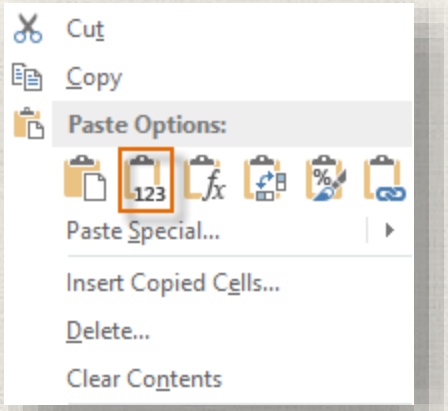
- In this example, the names of the nominees are in column A, so we'll put our formula in column B. In cell **B2**, we'll type a formula that tells Excel to capitalize the name in cell **A2**, which contains the first name on our list. The formula will look like this:

  =PROPER(A2)

	A	B
1	**Students**	**Corrected**
2	Richard newell	Richard Newell
3	kevin Mann	Kevin Mann
4	gregory johnson	Gregory Johnson
5	kristina Bridges	Kristina Bridges
6	amy Coleman	Amy Coleman
7	sharyn houston	Sharyn Houston
8	Lisa Kincade	Lisa Kincade
9	joseph costello	Joseph Costello
10	John carpenter	John Carpenter
11	william Franco	William Franco
12	Sandra Stewart	Sandra Stewart
13	dan Munoz	Dan Munoz
14	laura homan	Laura Homan
15	terence Patterson	Terence Patterson

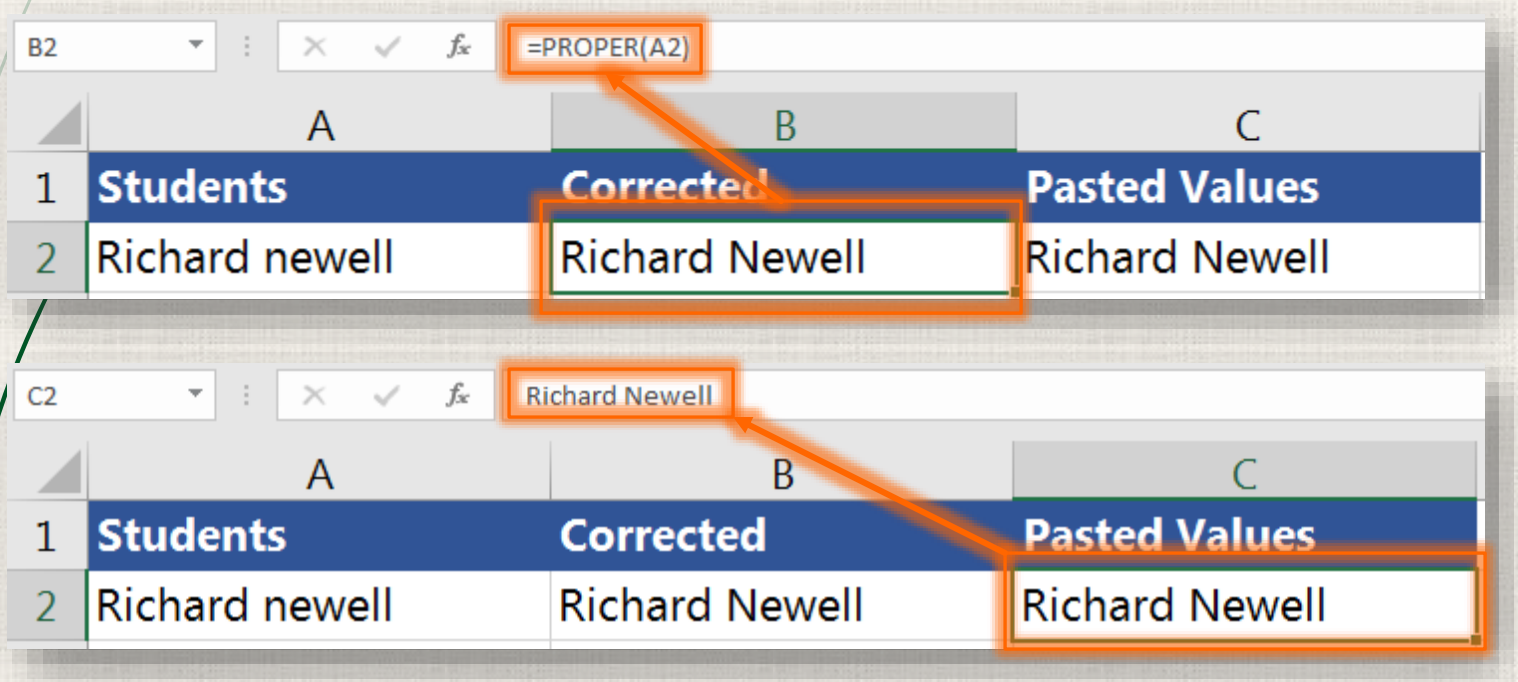
Proper Function

- Great! Now all the names of the award nominees are correctly capitalized in the spreadsheet. There's one problem, though: We still have the original uncapitalized names in column A. We can't delete column A because our formula in column B refers to it. Instead, we can **copy the values from column B** into a new column by using the **Paste Values** feature in Excel.
- To do this, select cells B2:B14 and click the **Copy** command (or press **Ctrl+C** on your keyboard). Then **right-click** the cell where you want to paste the values (**C2**, for example), then select the **Values** button from the menu that appears. If you're using Google Sheets, you can right-click and go to **Paste special > Paste values only**.



Proper Function

- Now we have a column that displays the corrected names but that doesn't depend on a formula or cell reference. This means we can delete our original columns (column A and column B).



Concatenate Function

- **CONCATENATE** lets you **combine two or more things in one cell**—and despite the long name, it's actually easy to use.
- Let's say we have a spreadsheet of contact information with last names and first names in separate columns, and we'd like to combine them to get each person's full name. In the image below, you can see that the **first names** are in **column B** and the **last names** are in **column A**. Our formula will go in cell **E2**.

	A	B	C	D	E	F
1	Last Name	First Name	City	State	Full Name	City & State
2	Carter	Josephine	Richmond	VA		
3	Logan	Ray	Orlando	FL		
4	Matthews	Tricia	Huntington	NY		
5	Muro	Annie	Tulsa	OK		
6	Murray	Robert	Taos	NM		
7	Persinger	Mildred	Hoboken	NJ		
8	Ramer	Eva	Urbana	IL		
9	Rodriguez	Joe	Madison	WI		
10	Ryan	Amanda	Tacoma	WA		
11	Sanchez	Billy	Reno	NV		

Concatenate Function

- **CONCATENATE will combine exactly what you tell it to combine, and nothing more.** If you want punctuation, spaces, or any other details to appear in the cell, you'll need to tell CONCATENATE to include it. In this case, we want the names to have a space in between them (so it doesn't say **JosephineCarter**), so we'll need to add an argument that contains a space. This means we'll need three arguments:

  B2 (first name)

  " " (a space in quotation marks)

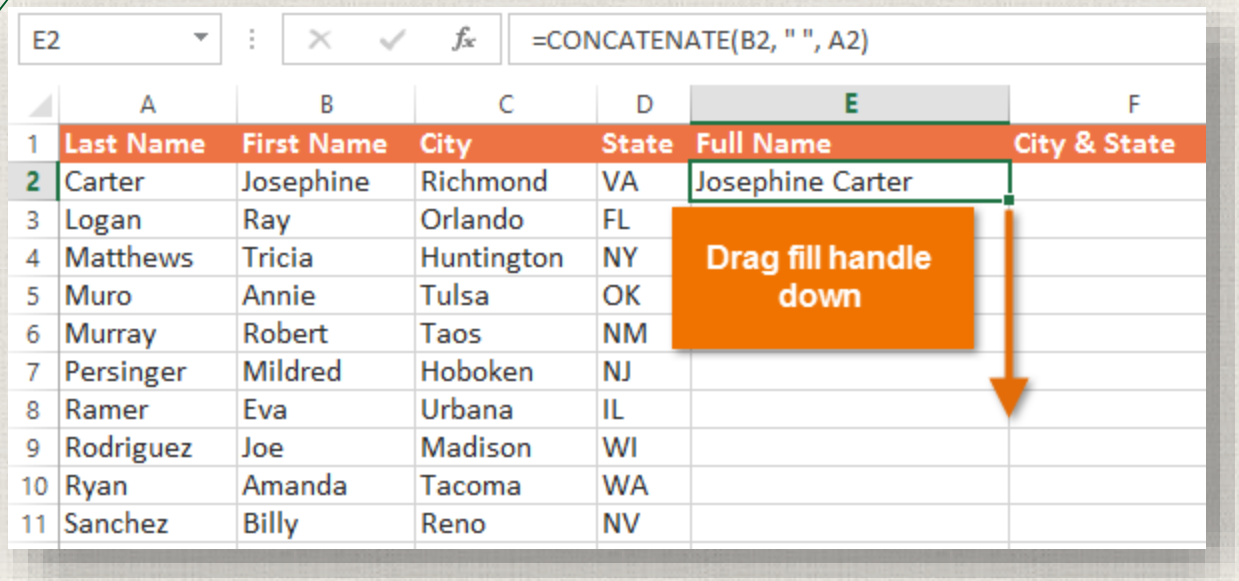
  A2 (last name)

- Now that we have our arguments, we can type the following formula into cell **E2**:

  =CONCATENATE(B2, " ", A2)

Concatenate Function

- That's it! When you press Enter, it should display the full name: **Josephine Carter**.
- Now you can **click and drag the fill handle down** through cell **E11**, and it should display the full name for each person.



	A	B	C	D	E	F
1	Last Name	First Name	City	State	Full Name	City & State
2	Carter	Josephine	Richmond	VA	Josephine Carter	
3	Logan	Ray	Orlando	FL		
4	Matthews	Tricia	Huntington	NY		
5	Muro	Annie	Tulsa	OK		
6	Murray	Robert	Taos	NM		
7	Persinger	Mildred	Hoboken	NJ		
8	Ramer	Eva	Urbana	IL		
9	Rodriguez	Joe	Madison	WI		
10	Ryan	Amanda	Tacoma	WA		
11	Sanchez	Billy	Reno	NV		

Concatenate Function

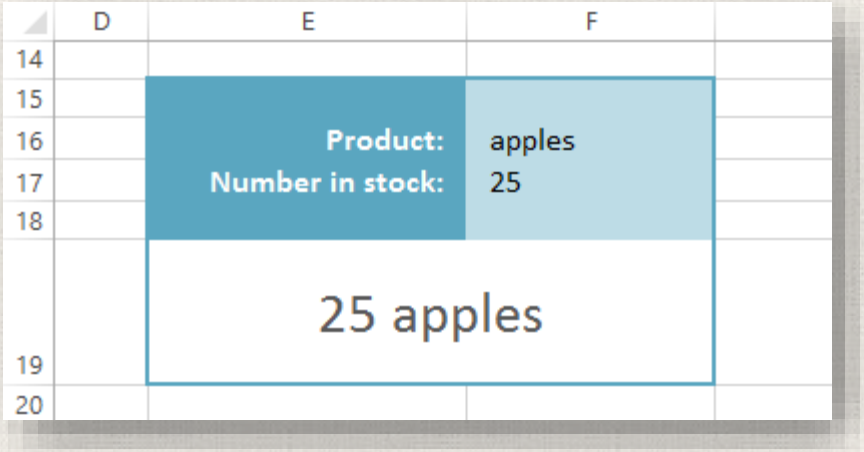
- Now, try using CONCATENATE to combine the **city** and **state** in column F so it looks like the image below.

	A	B	C	D	E	F
1	Last Name	First Name	City	State	Full Name	City & State
2	Carter	Josephine	Richmond	VA	Josephine Carter	Richmond, VA
3	Logan	Ray	Orlando	FL	Ray Logan	Orlando, FL
4	Matthews	Tricia	Huntington	NY	Tricia Matthews	Huntington, NY
5	Muro	Annie	Tulsa	OK	Annie Muro	Tulsa, OK
6	Murray	Robert	Taos	NM	Robert Murray	Taos, NM
7	Persinger	Mildred	Hoboken	NJ	Mildred Persinger	Hoboken, NJ
8	Ramer	Eva	Urbana	IL	Eva Ramer	Urbana, IL
9	Rodriguez	Joe	Madison	WI	Joe Rodriguez	Madison, WI
10	Ryan	Amanda	Tacoma	WA	Amanda Ryan	Tacoma, WA
11	Sanchez	Billy	Reno	NV	Billy Sanchez	Reno, NV

=CONCATENATE(C2, ", ", D2)

Concatenate Function

- You can even use CONCATENATE to combine numbers and text. For example, let's say we're using Excel to keep track of a store's inventory. We currently have 25 apples in stock, but **25** and **apples** are in separate cells. We want to combine them into one cell so that it looks like this:



Concatenate Function

- To do this, we'll need to combine three things:

  F17 (number in stock)

  " " (space)

  F16 (product name)

- Type the following formula into cell E19:

  =CONCATENATE(F17, " ", F16)

- Let's say we want it to say **We have 25 apples**. We'll just need to add an argument at the beginning that says **We have**:

  =CONCATENATE("We have ", F17, " ", F16)

DATE & TIME FUNCTIONS

DATE FUNCTIONS

Function	Explanation	Example	Result
DATE()	Returns a valid date according to Excel's date format. **DATE**(year, month, day)	=DATE(C17,D17,E17)	24/10/2018
DAY()	Returns the day of the month, a number from 1 to 31. **DAY**(number)	=DAY(F17)	15
DAYS()	Returns the number of days between two dates. **DAYS**(end_date, start_date)	=DAYS(F18,F17)	28
TODAY()	Return the current date. **TODAY**()	=TODAY()	22/03/2018
WEEKDAY()	Returns a number from 1 to 7, the day number of the week. *(1-Sunday through Saturday, 2-Monday through Sunday)* **WEEKDAY**(number, return_type)	=WEEKDAY(F17,1)	5
		=WEEKDAY(F17,2)	4
NETWORKDAYS()	Returns the number of whole workdays between two dates. **NETWORKDAYS**(start_date , end_date, holidays)	=NETWORKDAYS(F20,F21)	7
		=NETWORKDAYS(F20,F21,F22)	6

2018	10	24	15/03/2018
			12/04/2018
			05/04/2018
			15/04/2018
			09/04/2018

LOGICAL FUNCTIONS

LOGICAL FUNCTIONS

Function	Explanation	Example	Result
IF()	Checks whether a condition is met, and returns one value if TRUE, and another if FALSE. IF(logical_test, value_if_true, value_if_false)	=IF(A8>45,"Passed","Faied")	Passed
IFERROR()	Returns **value_if_error** if expression is an error and the value of the expression itself otherwise. IFERROR(value, value_if_error)	=IFERROR(A3/A2,"0")	0
		=IFERROR(A3/A10,"0")	2.5
IFNA()	Returns the value you specify if the expression resolves to **#N/A**, otherwise returns the result of the expression. IFERROR(value, value_if_na)	=IFERROR(VLOOKUP(55,$A$1:$A$15,1,FALSE),"Not Found")	Not Found
AND()	Checks whether all arguments are TRUE, and returns TRUE if all arguments are TRUE. IFERROR(logical1, logical2,...)	=AND(A2,A3)	FALSE
OR()	Checks whether any of the arguments are TRUE, and returns TRUE or FALSE. Returns FALSE only if all arguments are FALSE. IFERROR(logical1, logical2,...)	=OR(A2,A3)	TRUE
NOT()	Changes FALSE to TRUE, or TRUE to FALSE. IFERROR(logical)	=NOT(A2)	TRUE

0
5
8
-2
11
32
75
2

IF function : Donation

- There are several different columns, and each row contains the contact information for one person, along with the amount of money that person donated.

	A	B	C	D	E	F	G
1	Name	Email Address	Street Address	City	State	Zip	Donation
2	Silvia Armstrong	SilviaNArmstrong@email.com	2219 Andell Road	Columbus	OH	43215	$52.00
3	Kara Booker	KaraCBooker@email.com	839 Warner Street	Casper	WY	82601	$42.00
4	Susan Brown	SusanLBrown@email.com	4208 Fannie Street	Wharton	TX	77488	$20.00
5	Melissa Dales	MelissaSDales@email.com	1505 Saint Clair Street	Southaven	MS	38671	$25.00
6	Bessie Dandridge	BessieCDandridge@email.com	4884 Meadow Drive	Oklahoma City	OK	73102	$50.00
7	Barbara Fallis	BarbaraJFallis@email.com	4884 Cunningham Court	Farmington Hills	MI	48335	$79.00
8	Jeffrey Hegwood	JeffreyMHegwood@email.com	2663 Butternut Lane	Benton	IL	62812	$82.00
9	Cindy Hoch	CindyJHoch@email.com	4187 Straford Park	Lexington	KY	40507	$20.00
10	Judith Jamison	JudithMJamison@email.com	2433 Byers Lane	Marysville	CA	95901	$5.00
11	Kim Johnson	KimVJohnson@email.com	2037 Lochmere Lane	Plainville	CT	6062	$44.00
12	Laura Juarez	LauraDJuarez@email.com	1916 Horseshoe Lane	Fort Washington	PA	19034	$30.00
13	Kathleen McMullen	KathleenRMcMullen@email.com	1564 Daylene Drive	Ann Arbor	MI	48104	$76.00
14	Sherry Michel	SherryRMichel@email.com	1870 Franklin Avenue	Corpus Christi	TX	78401	$35.00
15	Dustin Moore	DustinSMoore@email.com	4793 Oakway Lane	Los Angeles	CA	90017	$41.00
16	Pedro Penton	PedroAPenton@email.com	4316 Pickens Way	Pecan Gap	TX	75469	$64.00

IF function : Donation

- So if a person donated at least $50, we need to write the words **Free Gift** in that donor's row. We could go through each row, look at the value, and then write **Free Gift** if the value is at least $50. But there are a lot of people on the spreadsheet, so it could be really time consuming to do this manually. What we really need is way to make our spreadsheet do this **automatically**.
- The **IF function** can simplify these kinds of problems. The IF function can look at a value in a cell to see if it meets a basic condition (in our example, **if the value is greater than or equal to 50**). Then it can then add text to an empty cell. Here, we can use the IF function to look at each person's donation, then add the words **Free Gift** if they donated at least $50. Basically, the IF function can do the same thing we were planning to do, only it can do it much faster.

IF function : Donation

=IF(G2>=50, "Free Gift", "")

Any formula starts with an **equals sign** (=)

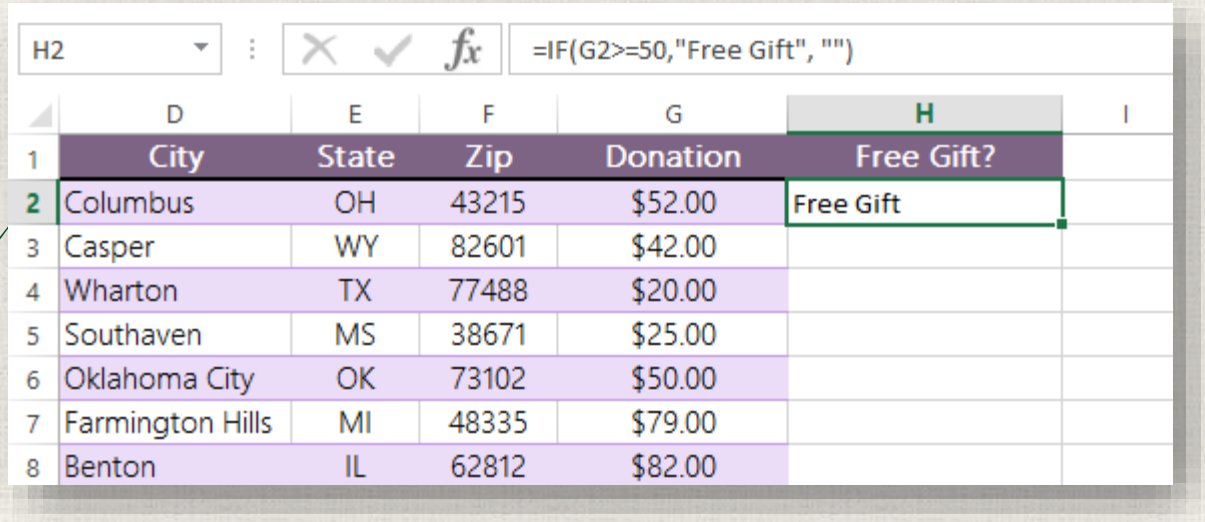
The first argument sets up the condition we want to test. For this formula, we want to know **if the value in cell G2 is greater than or equal to 50**.

The next argument will tell the function what to write if the condition is **true** (that is, if the value is greater than or equal to 50).

The final argument will tell the function what to write if the condition is **false**. We could make the function write something like **None** or **No**, but in this case we actually just want the cell to remain **blank**. To do this, we'll just type **double quotes ("") with no text inside**.

IF function : Donation

- OK, you're done! When you press Enter, the words **Free Gift** should appear in the cell.



	D	E	F	G	H	I
1	City	State	Zip	Donation	Free Gift?	
2	Columbus	OH	43215	$52.00	Free Gift	
3	Casper	WY	82601	$42.00		
4	Wharton	TX	77488	$20.00		
5	Southaven	MS	38671	$25.00		
6	Oklahoma City	OK	73102	$50.00		
7	Farmington Hills	MI	48335	$79.00		
8	Benton	IL	62812	$82.00		

IF function : Donation

- Now we can just drag the fill handle down to add the formula to the other cells in column H:

H2 | =IF(G2>=50,"Free Gift", "")

	D	E	F	G	H
1	City	State	Zip	Donation	Free Gift?
2	Columbus	OH	43215	$52.00	Free Gift
3	Casper	WY	82601	$42.00	
4	Wharton	TX	77488	$20.00	
5	Southaven	MS	38671	$25.00	
6	Oklahoma City	OK	73102	$50.00	Free Gift
7	Farmington Hills	MI	48335	$79.00	Free Gift
8	Benton	IL	62812	$82.00	Free Gift
9	Lexington	KY	40507	$20.00	
10	Marysville	CA	95901	$5.00	
11	Plainville	CT	6062	$44.00	
12	Fort Washington	PA	19034	$30.00	
13	Ann Arbor	MI	48104	$76.00	Free Gift
14	Corpus Christi	TX	78401	$35.00	
15	Los Angeles	CA	90017	$41.00	
16	Pecan Gap	TX	75469	$64.00	Free Gift
17	Lexington	KY	40507	$45.00	
18	Madison	WI	53703	$19.00	
19	San Jose	CA	95113	$36.00	
20	Waltham	MA	2154	$90.00	Free Gift
21	Rochelle Park	NJ	7662	$82.00	Free Gift

IF function : Invoice

- You probably know that we currently charge $5.99 for shipping on all of our orders. Well, we're going to start offering **free shipping if the subtotal of an order is at least $100**.

	A	B	C	D	E	F	G
1	Item ID	Item Price	Units	Subtotal			
2	4095	$12.99	2	$25.98		Tax Rate	7.5%
3	3427	$13.00	2	$26.00		Shipping Cost	$5.99
4	3653	$45.99	1	$45.99			
5			Tax	$7.35			
6			Order Subtotal	$105.32			
7			Shipping	$5.99			
8			Total Order Cost	$111.31			

IF function : Invoice

=IF(D6>=100, 0 , $G$3)

Any formula starts with an **equals sign** (=)

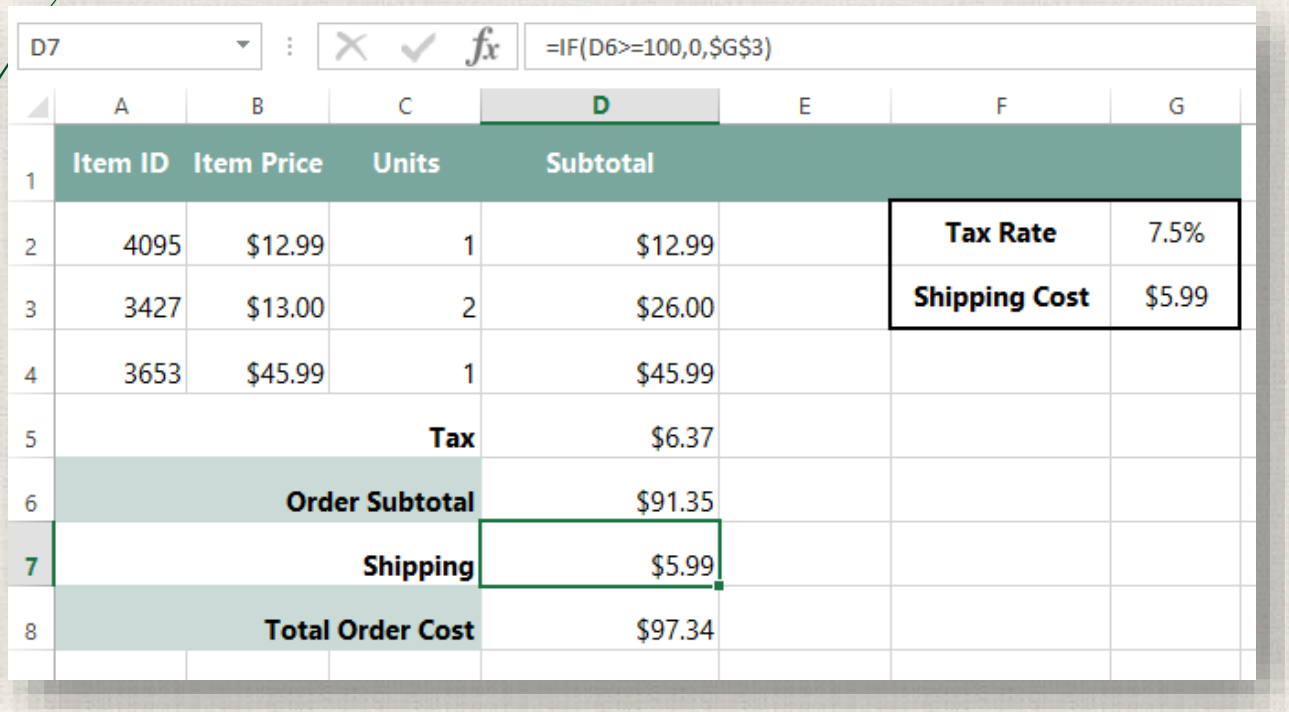
The first argument sets up the condition we want to test. For this formula, we want to know **if the value in cell D6 is greater than or equal to 100.**

The next argument will tell the function what to write if the condition is **true.** In this example, if the value is greater than or equal to 100, we want it to write "0" in the cell.

The final argument will tell the function what to write if the condition is **false**. In this example, if the value is less than 100, we want it to use the value from cell G3 (5.99). So our third argument will be **$G$3** (we'll use an absolute reference here just in case we move this formula in the future).

IF function : Invoice

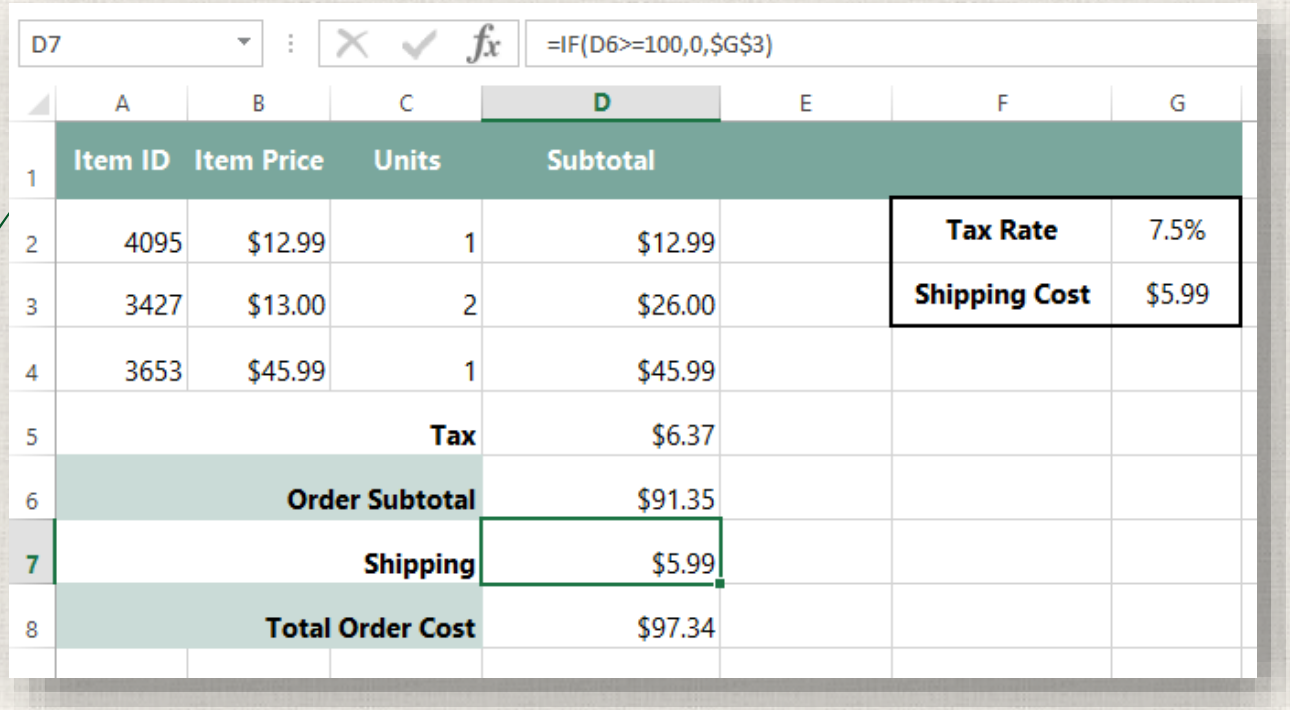
- If you entered the function correctly, the shipping cost should change from **$5.99** to **$0.00**. That's because the **Order Subtotal** is more than $100.
- If you want to test the function, change the value in cell **C2** from **2** to **1**. Because the Order Subtotal is now less than $100, the Shipping cost should change from $0.00 to $5.99.



	A	B	C	D	E	F	G
1	Item ID	Item Price	Units	Subtotal			
2	4095	$12.99	1	$12.99		Tax Rate	7.5%
3	3427	$13.00	2	$26.00		Shipping Cost	$5.99
4	3653	$45.99	1	$45.99			
5			Tax	$6.37			
6			Order Subtotal	$91.35			
7			Shipping	$5.99			
8			Total Order Cost	$97.34			

IF function : Invoice

- Let's say we want to change the shipping cost to $6.99. Select the cell we'd need to edit.



	A	B	C	D	E	F	G
1	Item ID	Item Price	Units	Subtotal			
2	4095	$12.99	1	$12.99		Tax Rate	7.5%
3	3427	$13.00	2	$26.00		Shipping Cost	$5.99
4	3653	$45.99	1	$45.99			
5			Tax	$6.37			
6			Order Subtotal	$91.35			
7			Shipping	$5.99			
8			Total Order Cost	$97.34			

MATH FUNCTIONS

MATH FUNCTIONS

	Function	Explanation	Example	Result
5	ABS()	Calculates the absolute value of a number.	=ABS(A2)	5
-2	PI()	Gives the value of Pi with an accuracy of 15 digits	=PI()	3.141592654
-7	RADIANS()	It converts degrees into radians.	=SIN(RADIANS(30))	0.5
6	DEGREES()	It converts radians into degrees.	=DEGREES(PI())	180
9	SIN()	Returns the sine of a given angle.	=SIN(RADIANS(30))	0.5
0	FACT()	Returns the factorial of a number. n factorial is the product of the numbers 1 through n. (n! = 1*2*3*...*n)	=FACT(A2)	120
5	INT()	Rounds the given real number down to the nearest integer.	=INT(A5)	6
3			=INT(A2)	5
2	MOD()	Returns the remainder after a number is divided by a divisor.	=MOD(A6,A9)	4
12.569	POWER()	Returns the result of a number raised to a power.	=POWER(2,4)	16
	PRODUCT()	Multiplies the numbers within the given range.	=PRODUCT(A9:A11)	30
	SUMPRODUCT()	Returns the sum of the products of corresponding ranges.	=SUMPRODUCT(A5:A7, A9:A11)	57
	ROUND()	Rounds a number to the specified number of digits.	=ROUND(A12,1)	12.6
	TRUNC()	Trunc removes the digits after the specified number of digits from the fractional part.	=TRUNC(A12,1)	12.5
	SUMIF()	Calculates the sum of the numbers within the range according to the given criteria.	=SUMIF(A2:A7, "<0")	-9
	SQRT()	Returns the square root of a number.	=SQRT(A6)	3
	RANDBETWEEN()	Returns an evenly distributed random number greater than or equal to the bottom and less than or equal to the top value.	=RANDBETWEEN(1,5)	1

STATISTICAL FUNCTIONS

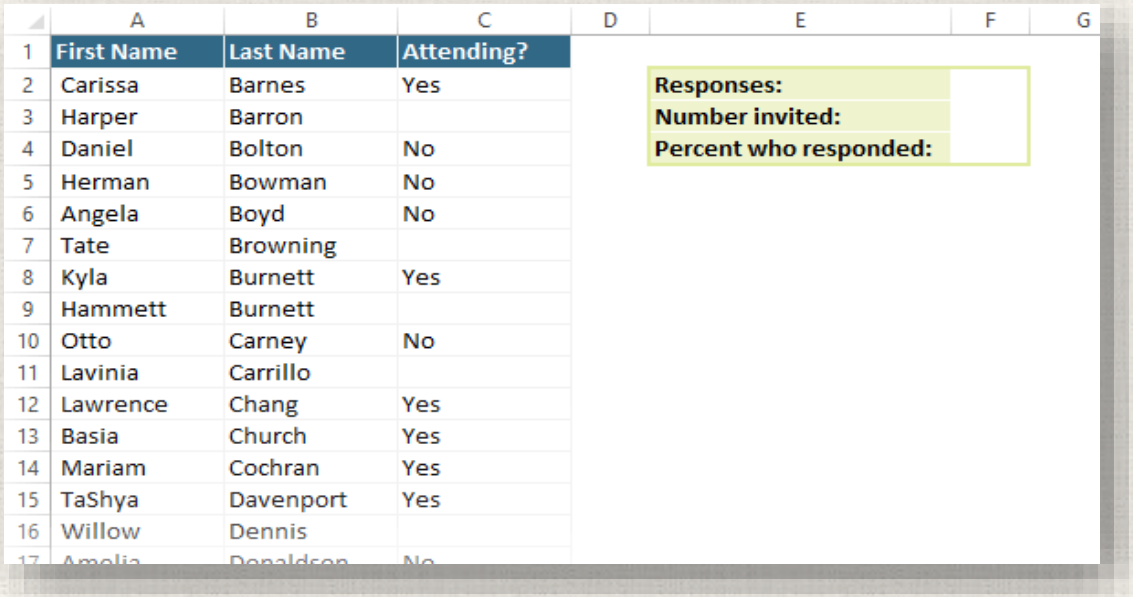
STATISTICAL FUNCTIONS

Function	Explanation	Example	Result
COUNT()	Counts the number of cells in a range that contain numbers. **COUNT**(value1,value2)	=COUNT(A3:A15)	8
COUNTA()	Counts the number of cells that are not empty within the selected range. **COUNTA**(value1,value2)	=COUNTA(A3:A15)	11
COUNTIF()	Counts the number of cells within a range that meet the given condition. **COUNTIF**(range,criteria)	=COUNTIF(A3:A15, ">10")	4
COUNTBLANK()	Counts the number of empty cells in a specified range of cells. **COUNTBLANK**(range)	=COUNTBLANK(A3:A15)	02/01/1900
MAX()	Returns the largest value in a set of values. Ignores logical values and text. **MAX**(number1, number2,...)	=MAX(A3:A15)	97
MIN()	Returns the smallest value in a set of values. Ignores logical values and text. **MIN**(number1, number2,...)	=MIN(A3:A15)	0
AVERAGE()	Returns the average of its arguments, which can be numbers or names, arrays, or references that contain numbers. **AVERAGE**(number1, number2,...)	=AVERAGE(A3:A15)	28.25

A
5
1
7
text
0
21
54
41
*
/
97

CountA Function

- COUNTA simply looks at a range of cells and tells you **how many of the cells contain data**. In other words, it looks for nonblank cells. This can be useful in a variety of situations.
- In our example, we're using Excel to plan an event. We've sent out invitations to everyone, and once we receive their responses, we'll type either **Yes** or **No** in column C. As you can see, column C still has some empty cells because we haven't heard back from everyone.



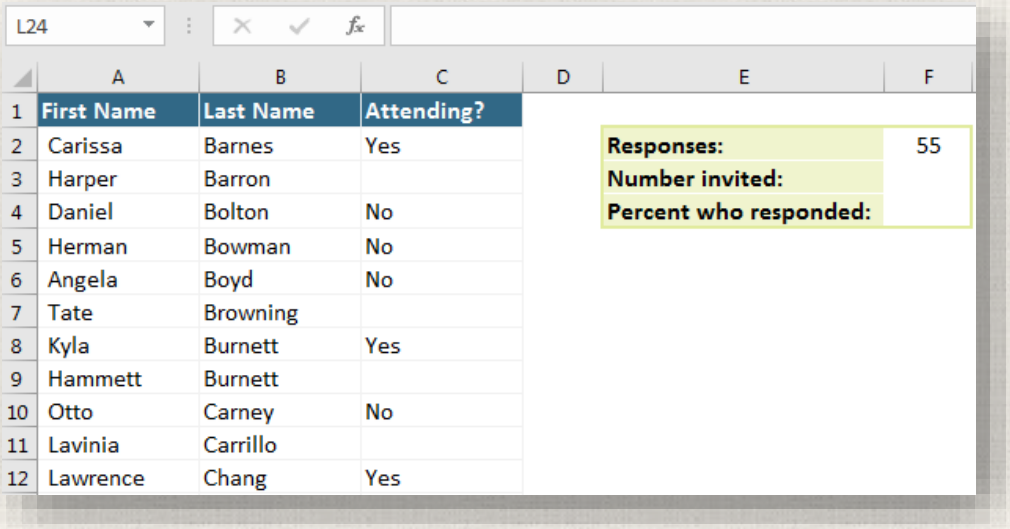
	A	B	C	D	E	F	G
1	First Name	Last Name	Attending?				
2	Carissa	Barnes	Yes		Responses:		
3	Harper	Barron			Number invited:		
4	Daniel	Bolton	No		Percent who responded:		
5	Herman	Bowman	No				
6	Angela	Boyd	No				
7	Tate	Browning					
8	Kyla	Burnett	Yes				
9	Hammett	Burnett					
10	Otto	Carney	No				
11	Lavinia	Carrillo					
12	Lawrence	Chang	Yes				
13	Basia	Church	Yes				
14	Mariam	Cochran	Yes				
15	TaShya	Davenport	Yes				
16	Willow	Dennis					
17	Amelia	Donaldson	No				

CountA Function

- Our responses are in cells **C2:C86**, but we can actually include a few extra rows in case we decide to invite more people:

=COUNTA(C2:C100)

- When you press Enter, you'll see that we've received **55 responses**. Here's the best part: We can continue to update this spreadsheet as we receive responses, and our function will **automatically recalculate** to give us the correct answer. Try entering **Yes** or **No** into some of the empty cells in column C, and you should see the value in cell **F2** change.

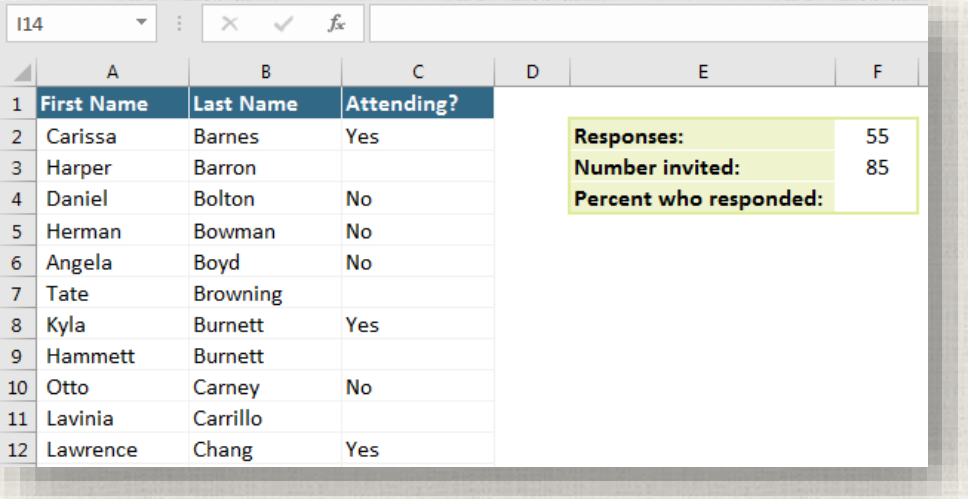


CountA Function

- We can also use COUNTA to calculate the **total number of people that were invited**. In cell **F3**, type the following function and press Enter:

=COUNTA(A2:A100)

- All we had to do was type a different range (A2:A100), and it counts all of the **first names**, giving us an answer of **85**. If you type more names at the bottom of the spreadsheet, Excel will automatically recalculate this value. However, if you type anything below row 100, you'll need to **update your ranges** so any new people are included.

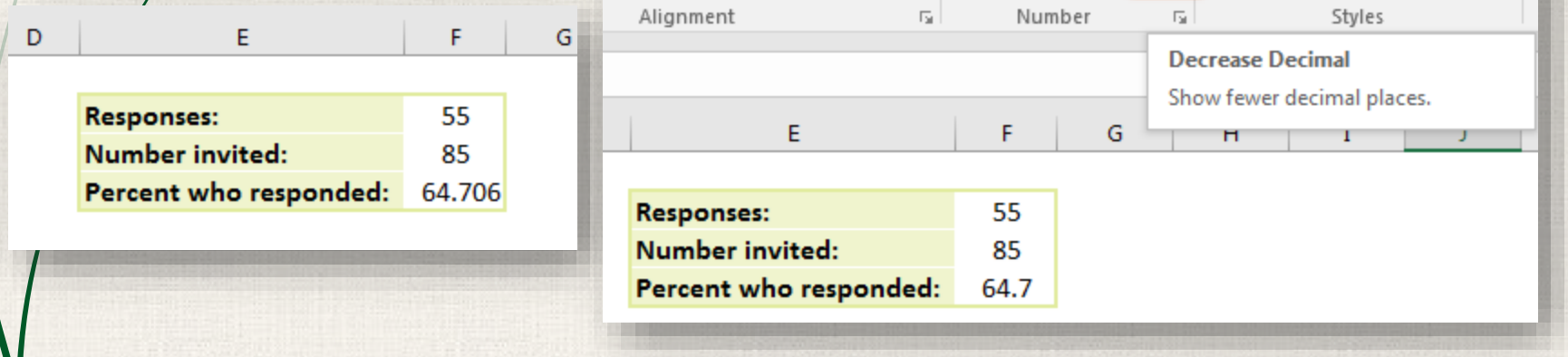


I14

	A	B	C	D	E	F
1	First Name	Last Name	Attending?			
2	Carissa	Barnes	Yes		Responses:	55
3	Harper	Barron			Number invited:	85
4	Daniel	Bolton	No		Percent who responded:	
5	Herman	Bowman	No			
6	Angela	Boyd	No			
7	Tate	Browning				
8	Kyla	Burnett	Yes			
9	Hammett	Burnett				
10	Otto	Carney	No			
11	Lavinia	Carrillo				
12	Lawrence	Chang	Yes			

CountA Function

- We now have the number of responses in cell **F2** and the total number of invitees in cell **F3**. It would be great to calculate the **percentage** of people who have responded. See if you can **write a formula in cell F4 that calculates the percentage**. (Show only one decimal by using decrease decimal.)



=(100*F2)/F3

LOOKUP FUNCTIONS

LOOKUP FUNCTIONS

Function	Explanation	Example	Result
ADDRESS()	Returns the address of a cell in a worksheet. **ADDRESS**(row_num, column_num)	=ADDRESS(20,5)	$E$20
FORMULATEXT()	Returns a formula as a string. **FORMULATEXT**(reference)	=E21*E22	30
LOOKUP()	Looks up a value either from a one-row or one-column range. **LOOKUP**(value, lookup_range, [result_range])	=LOOKUP(C24,B20:B22)	2000 Silver
HLOOKUP()	Looks up and retrieves data from a specific row in table. **HLOOKUP**(lookup_value, table_array, row_index_num, range_lookup)	=HLOOKUP(I22,I19:K20,2,FALSE)	Gold
VLOOKUP()	Looks up and retrieves data from a specific column in table. **VLOOKUP**(lookup_value, table_array, col_index_num, range_lookup)	=VLOOKUP(C25,B20:C22,2,FALSE)	Silver
HYPERLINK()	Creates a link to a document on your harddrive, a server, or on the internet. **HYPERLINK**(link_location, friendly_name)	Right-click over a cell, select Link from the list, and paste the link.	
TRANSPOSE()	Switches row into column and column into rows. **TRANSPOSE**(array)	Copy a cell range, go to Paste Special window and select TRANSPOSE option.	

Points	Level
1000	Bronze
2000	Silver
3000	Gold

text
5
6

Points	1000	2000	3000
Level	Bronze	Silver	Gold

Points	3000

Points	2350
Points	2000

VLOOKUP function : Invoice

- Our coworker asked if we could use the Product ID number to find the product **name** and **price** from the Products worksheet. Luckily, the **VLOOKUP function** can do this automatically.
- Before we write our function, we'll need to take a moment to think carefully about the arguments will tell VLOOKUP **what to search for** and **where to look**.

	A	B	C	D	E
1	**Product ID**	**Product Name**	**Product Price**	**Units**	**Subtotal**
2	MEA7879				$0.00
3	GAR7883				$0.00
4	CHE7888				$0.00
5				**Tax**	$0.00
6				**Order Subtotal**	$0.00
7				**Shipping**	$5.99
8				**Total Order Cost**	$5.99

Invoice | Products

VLOOKUP function : Invoice

=VLOOKUP(A2 , Products!$A$2:$B$16 , 2 , FALSE)

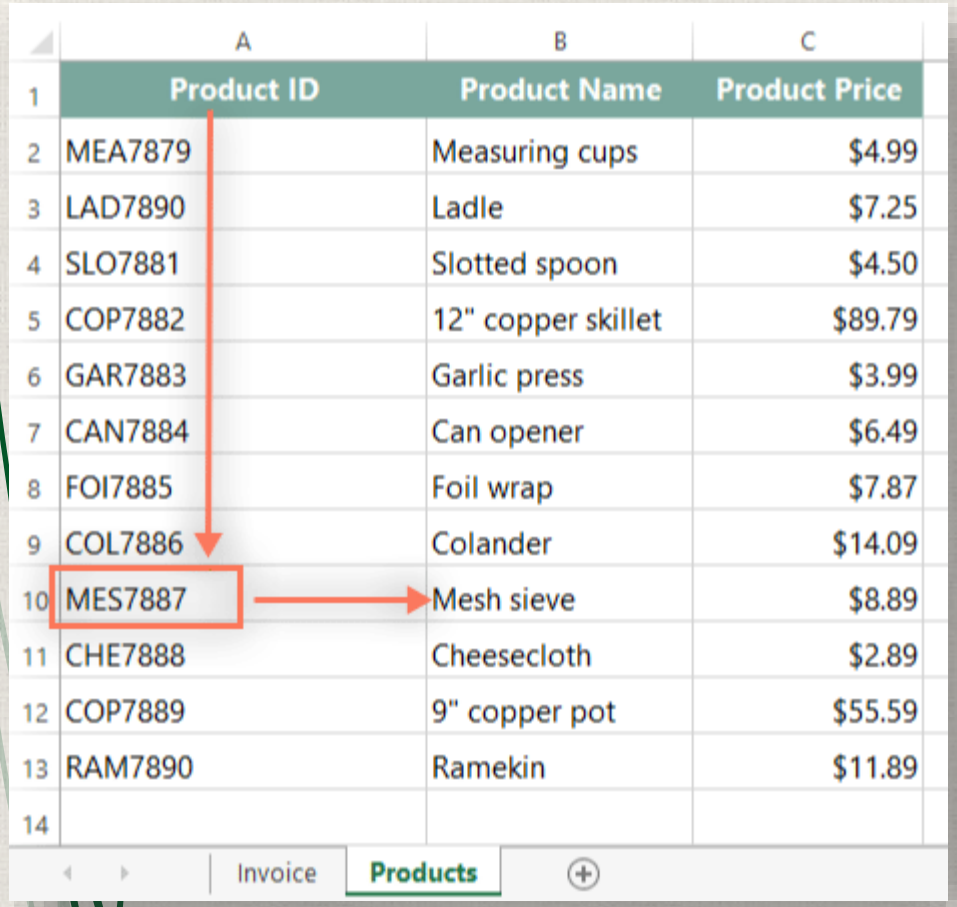
The **first argument** tells VLOOKUP **what to search for**. **In our example**, we're searching for the product ID number, which is in cell **A2**.

The **second argument** is a **cell range** that tells VLOOKUP **where to look for the value** from our first argument. **In our example**, we want it to search for this value in cell range **A2:C13** on the **Products** works heet, so our second argument is **Products!$A$2:$C$13.**

The **third argument** is the **column index number**. The first column in the cell range from the previous argument is **1**, the second column is **2**, and so on. **In our example**, we're looking for the **Product Name**. The names are stored in the **second column** of the cell range from the previous argument, so our third argument is **2**.

The **fourth argument** tells VLOOKUP if it should look for approximate or exact matches. If it is **TRUE**, it will look for **approximate** matches. If it is **FALSE**, it will look for **exact** matches. **In this example**, we're only looking for **exact** matches, so our fourth argument is **FALSE**.

VLOOKUP function : Invoice



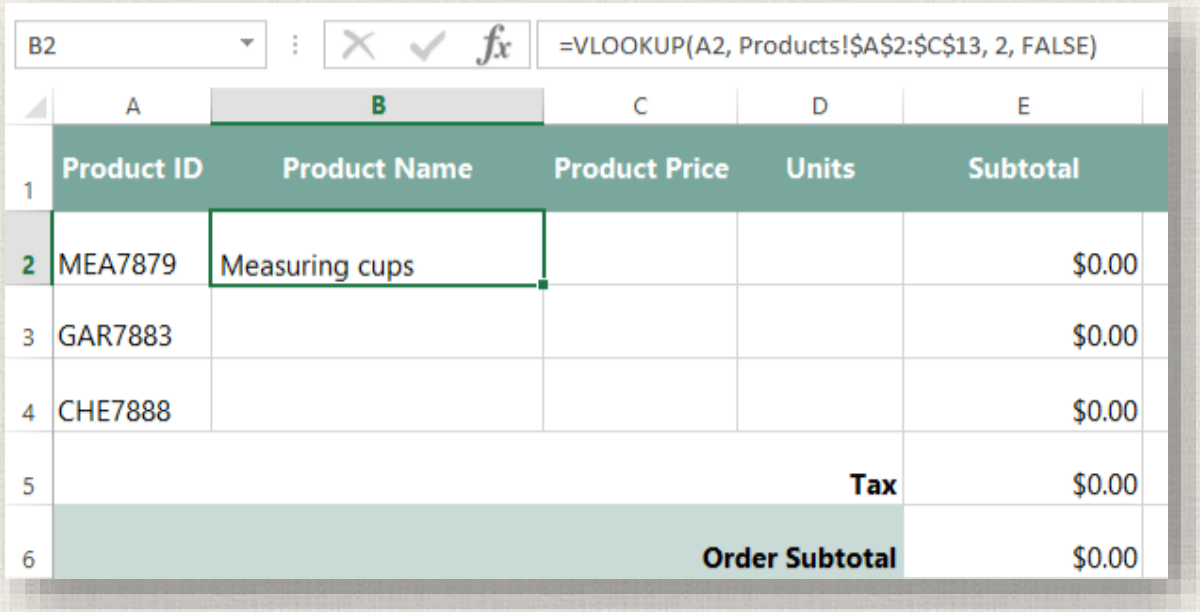
	A	B	C
1	Product ID	Product Name	Product Price
2	MEA7879	Measuring cups	$4.99
3	LAD7890	Ladle	$7.25
4	SLO7881	Slotted spoon	$4.50
5	COP7882	12" copper skillet	$89.79
6	GAR7883	Garlic press	$3.99
7	CAN7884	Can opener	$6.49
8	FOI7885	Foil wrap	$7.87
9	COL7886	Colander	$14.09
10	MES7887	Mesh sieve	$8.89
11	CHE7888	Cheesecloth	$2.89
12	COP7889	9" copper pot	$55.59
13	RAM7890	Ramekin	$11.89
14			

Invoice | Products

- Before we start using VLOOKUP, it will be helpful to know what it does.
- In our example, it will search for the **Product ID** number on the **Products** worksheet. It first searches **vertically down** the first column (VLOOKUP is short for "vertical lookup").
- When it finds the desired product ID, it **moves to the right** to find the product name and product price.

VLOOKUP function : Invoice

- Now that we have our arguments, we'll write our function in cell **B2**.
- If you entered the function correctly, the product name should appear: **Measuring cups**. If you want to test your function, change the Product ID number in cell A2 from **MEA7879** to **CHE7888**. The product name should change from **Measuring cups** to **Cheesecloth**.



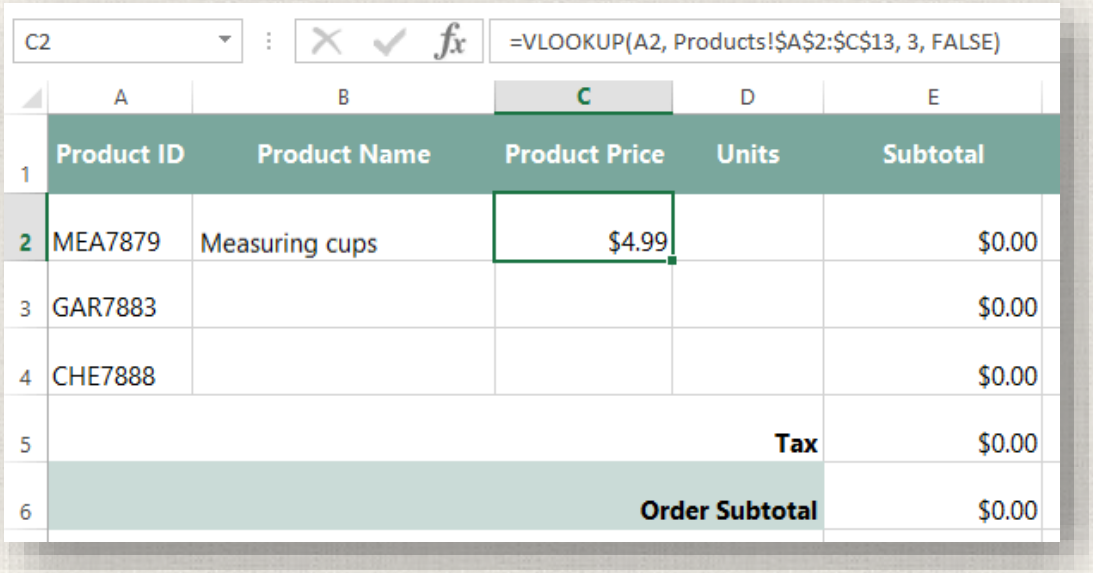
	A	B	C	D	E
1	Product ID	Product Name	Product Price	Units	Subtotal
2	MEA7879	Measuring cups			$0.00
3	GAR7883				$0.00
4	CHE7888				$0.00
5				Tax	$0.00
6				Order Subtotal	$0.00

VLOOKUP function : Invoice

- Next, we also want the **Product ID** to pull in the **product price**, so we'll use the VLOOKUP function again. Since we're using the same data, this function will be very similar to the one we just added. In fact, all we have to do is change the **third argument** to **3**. This will tell VLOOKUP to pull in the data from the third column, where the **product price** is stored:

  =VLOOKUP(A2, Products!$A$2:$C$13, 3, FALSE)

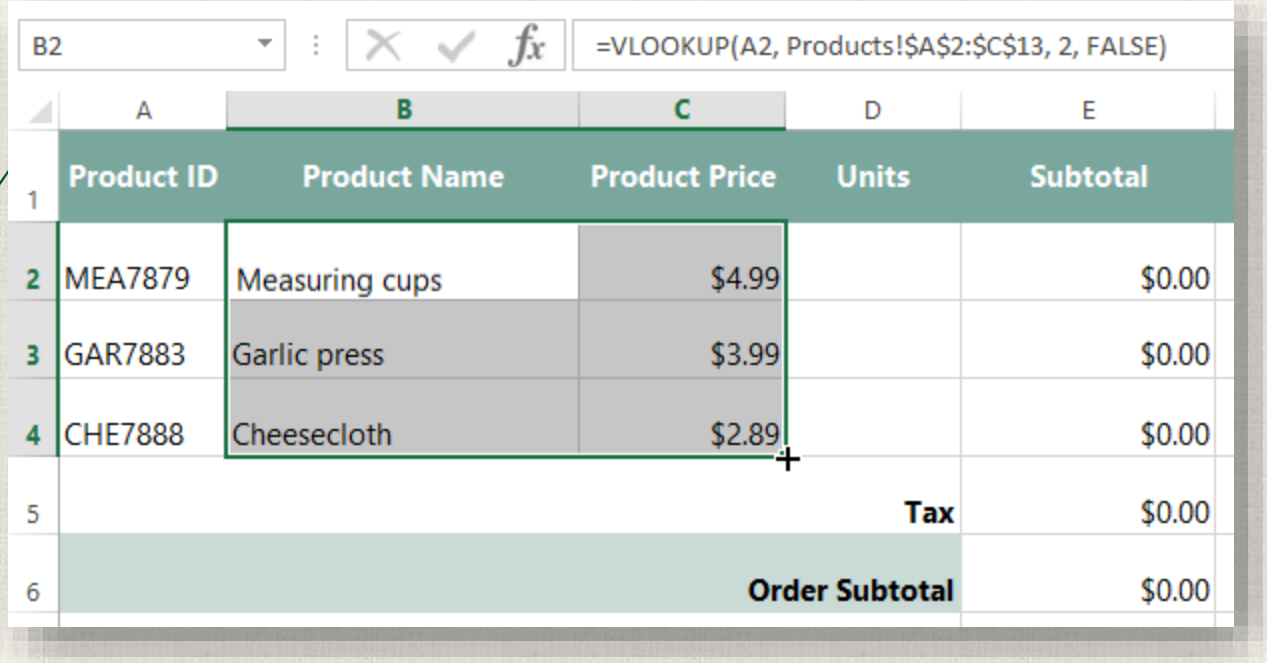
- OK, let's enter our new formula in cell C3:



	A	B	C	D	E
1	Product ID	Product Name	Product Price	Units	Subtotal
2	MEA7879	Measuring cups	$4.99		$0.00
3	GAR7883				$0.00
4	CHE7888				$0.00
5				Tax	$0.00
6				Order Subtotal	$0.00

VLOOKUP function : Invoice

- We've got our formulas working, so we can just **select cells B2 and C2** and then **drag the fill handle down** to copy the formulas to the other rows in the invoice. Now, each row is using VLOOKUP to find the Product Name and Product Price.



	A	B	C	D	E
1	Product ID	Product Name	Product Price	Units	Subtotal
2	MEA7879	Measuring cups	$4.99		$0.00
3	GAR7883	Garlic press	$3.99		$0.00
4	CHE7888	Cheesecloth	$2.89		$0.00
5				Tax	$0.00
6				Order Subtotal	$0.00

VLOOKUP function : Pets

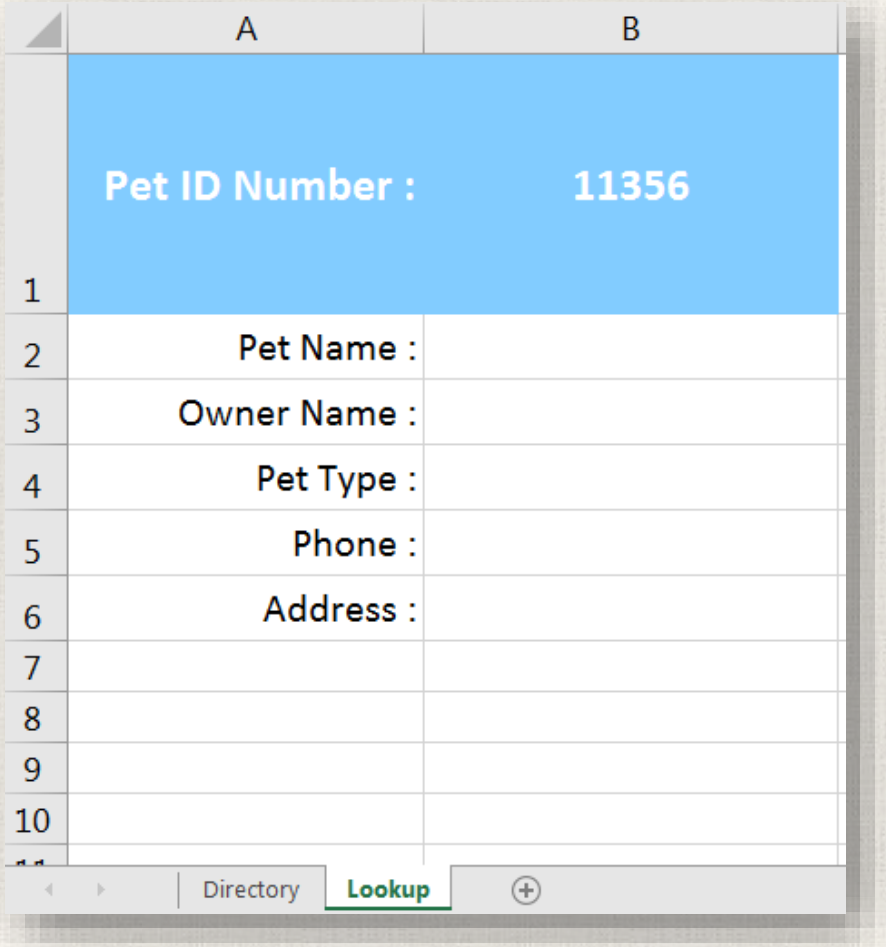
- Let's say a veterinarian's office is creating a spreadsheet to look up patient information. Here's the patient directory. This is where information will be pulled from:

	A	B	C	D	E	F
1	Patient ID	Pet Name	Owner Name	Pet Type	Phone	Address
2	124687	Waffles	Ahn	bunny	548-379-5689	313 Park Ave
3	15987	Bingo	Aloysius	cat	889-255-7986	914 First Avenue St
4	654123	Lupita	Amburn	bird	547-789-9768	254 Humboldt Pl
5	97685	Jerry	Arden	dog	275-987-6644	35 W Isabel St
6	64597	Fluffles	Aurelias	lizard	335-312-4675	3355 Harper Ave
7	13987	Barker	Baker	cat	738-966-7689	12 H Ave Unit 12
8	798465	Harley	Bendita	dog	462-654-9879	1098 Pence St
9	497789	Jackson	Bing	cat	335-868-2346	1049 Ocean Blvd
10	12478	Tippy	Brahm	turtle	584-654-1589	6598 Salisbury St
11	435789	Monster	Burns	fish	927-978-3565	109 Gardener Ct
12	11356	Catfish	Campana	dog	577-987-6868	63 Canon Court
13	1346598	Princess	Carpenter	dog	462-336-4658	987 Briarwood Pl
14	56798	Frank	Celas	bird	277-233-6654	12365 Kinston Ave
15						

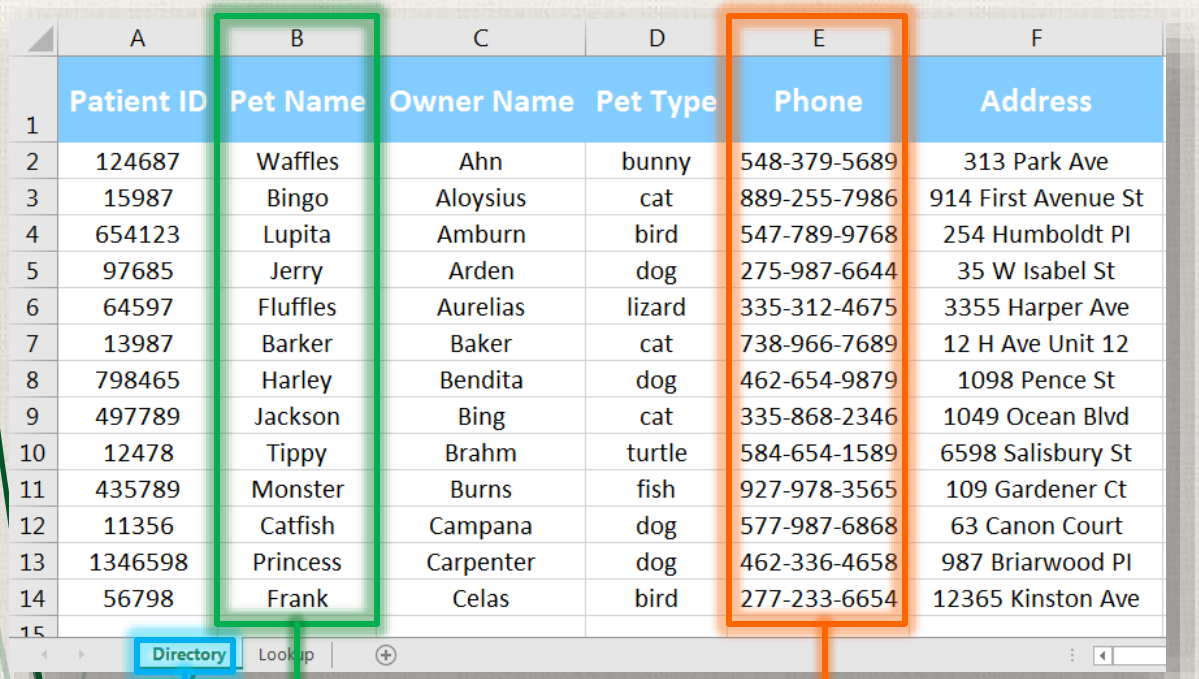
Directory | Lookup

VLOOKUP function : Pets

- Here's the patient lookup sheet. This is where the function will be inserted.

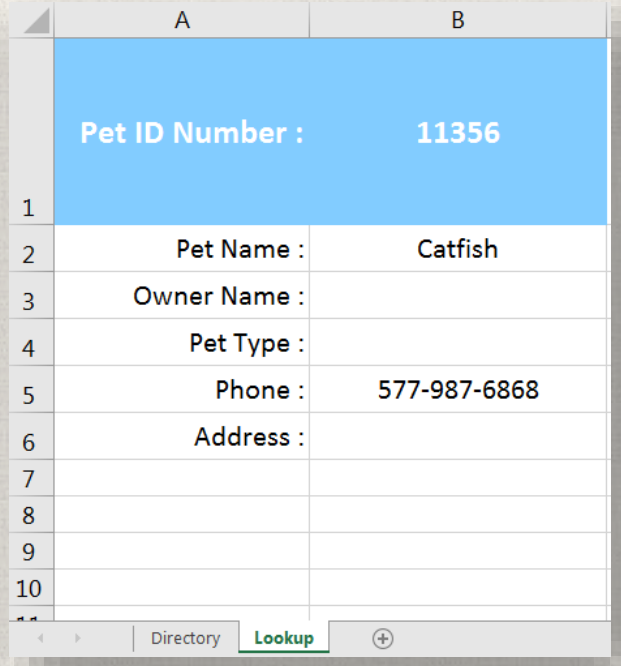


VLOOKUP function : Pets



	A	B	C	D	E	F
1	Patient ID	Pet Name	Owner Name	Pet Type	Phone	Address
2	124687	Waffles	Ahn	bunny	548-379-5689	313 Park Ave
3	15987	Bingo	Aloysius	cat	889-255-7986	914 First Avenue St
4	654123	Lupita	Amburn	bird	547-789-9768	254 Humboldt Pl
5	97685	Jerry	Arden	dog	275-987-6644	35 W Isabel St
6	64597	Fluffles	Aurelias	lizard	335-312-4675	3355 Harper Ave
7	13987	Barker	Baker	cat	738-966-7689	12 H Ave Unit 12
8	798465	Harley	Bendita	dog	462-654-9879	1098 Pence St
9	497789	Jackson	Bing	cat	335-868-2346	1049 Ocean Blvd
10	12478	Tippy	Brahm	turtle	584-654-1589	6598 Salisbury St
11	435789	Monster	Burns	fish	927-978-3565	109 Gardener Ct
12	11356	Catfish	Campana	dog	577-987-6868	63 Canon Court
13	1346598	Princess	Carpenter	dog	462-336-4658	987 Briarwood Pl
14	56798	Frank	Celas	bird	277-233-6654	12365 Kinston Ave

Directory | Lookup



	A	B
1	Pet ID Number :	11356
2	Pet Name :	Catfish
3	Owner Name :	
4	Pet Type :	
5	Phone :	577-987-6868
6	Address :	

Directory | Lookup

Pet Name	=VLOOKUP($B$1, Directory!$A$2:$F$14, 2, FALSE)
Phone	=VLOOKUP($B$1, Directory!$A$2:$F$14, 5, FALSE)

THE END