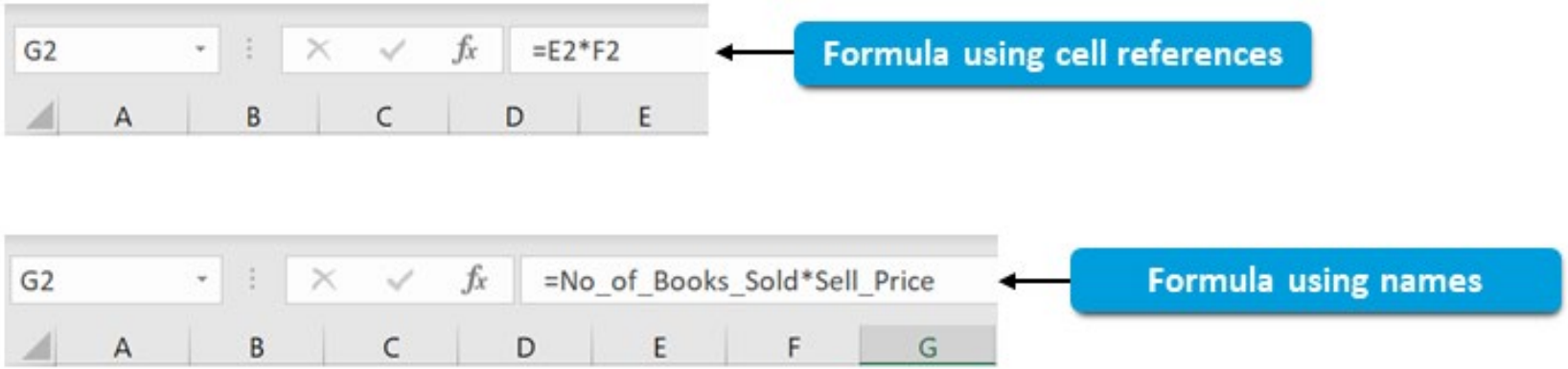
Work with Ranges

This course is based on the Microsoft Excel 2021-2024 Mastery Specialization by Logical Operations

Cell and Range Names

Cell names and range names are exactly what they sound like. They are meaningful names you assign to a given cell or range to make it easier to both understand what calculations are being performed in a formula and to reuse the references for a number of purposes. Take a look at the following image, which shows two versions of the same formula: one using cell references and one using names.



- Cell and range references aren't the only items you can name in Excel. You can name other objects, such as tables and even formulas themselves. Collectively, the names you assign to all of these items are known as *defined names*.
- It may be a good idea to indicate in a name whether the reference is absolute or relative, as the name will be displayed precisely as you created it, regardless of which type of reference it contains.

Cell and Range Names

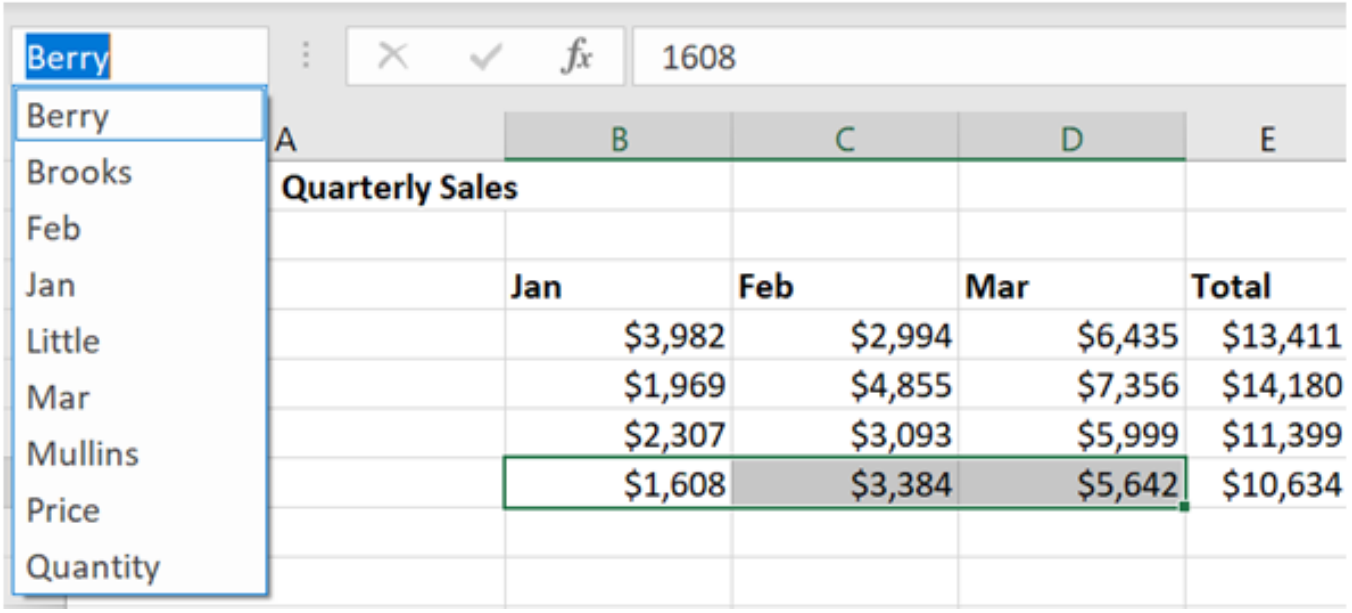
Although you can come up with an incredible array of different names, there are some rules you must follow:

- Names must **begin with a letter, an underscore, or a backslash**.
- After the first character, names **can contain letters, numbers, periods, and underscores**.
- Names **cannot contain spaces**.
- Names **cannot be the same as a cell or a range reference**. For example, you cannot use *A1* as a name.
- Names **have a defined scope**, either to a worksheet or a workbook, and must be unique within that scope.
- Names can contain **up to 255 total characters**.
- Excel **does not recognise casing differences** for names. So, within the same scope, you cannot, for example, create both *SalesTotals* and *salestotals* as names.
- You **can use a single letter as a name**, but you **cannot use either *C* or *R*** (either uppercase or lowercase) as these are used as shorthand for selecting an entire row or an entire column in other Excel features.

Names and the Name Box

There are several methods you can use to create names in Excel 2021. The most direct of these is to use the **Name Box**. To name a cell or a range, you can simply select the desired cell or range and type the desired name in the **Name Box**.

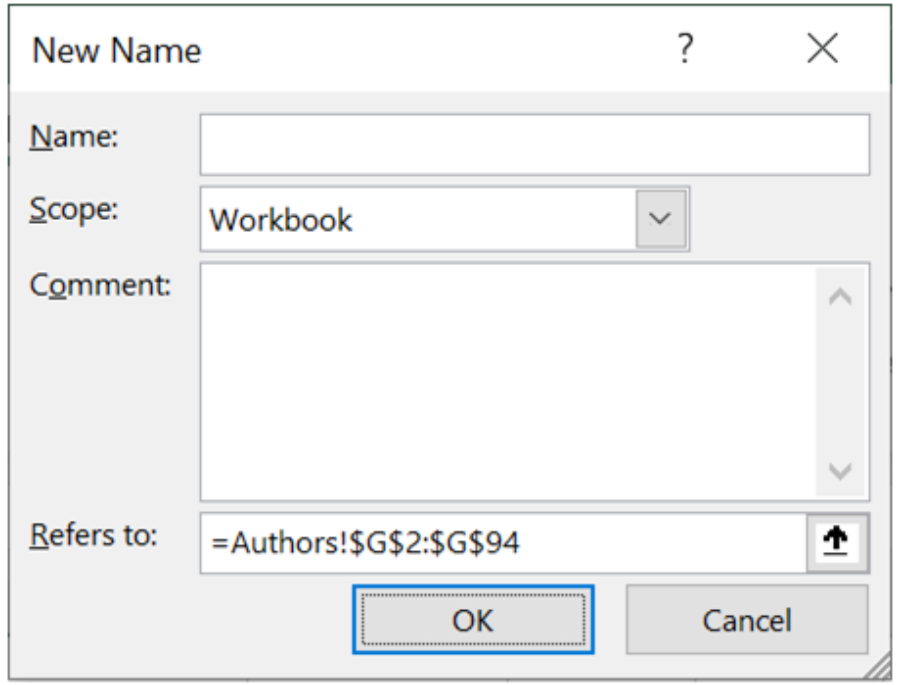
Once you've created named cells and ranges, you can access those cells and ranges from the **Name Box** drop-down menu. This is a quick way to select a cell or range that you've already named. Additionally, if you manually select a named cell or range on a worksheet, the name - not the cell reference - appears in the **Name Box**. Names created in the **Name Box**, by default, have "Workbook" as their scope.



The New Name Dialog Box

You can also name cells or ranges by using the **New Name** dialog box. The advantage here is that you have greater control over configuring precisely what the name refers to. You can access the **New Name** dialog box by selecting **Formulas → Define Name**.

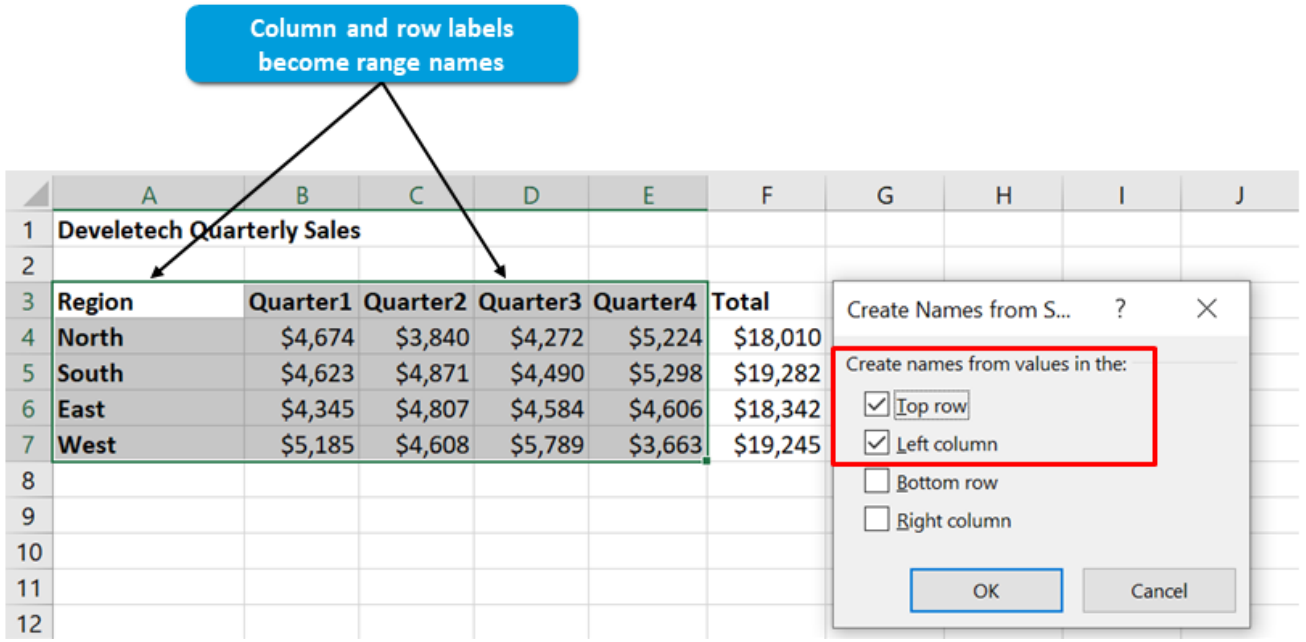
New Name Dialog Box Element	Enables You To
Name field	Enter a name for the cell or range.
Scope drop-down menu	Assign a scope to the name. This can be either the entire workbook or a particular worksheet. You cannot create two identical names within the same scope. You can, however, create identical names for both a worksheet and the workbook containing that worksheet. On the worksheet, the name that has the worksheet as its scope will take precedence. On all other worksheets, the name that has the workbook as its scope will take precedence.
Comment field	Enter a brief description of the named cell or range to help clarify its purpose.
Refers to field	View or edit the name's reference. Whatever cell or range is selected when you open the **New Name** dialog box will be displayed as an absolute reference in the **Refers to** field by default.



The Create from Selection Command

The **Create from Selection command** in Excel is a quick way to generate named ranges based on the labels in your worksheet. Instead of manually naming each range, you can select a block of data and let Excel automatically create names from the row or column headers. This works best when your worksheet has clearly defined labels, as Excel uses them to generate the names. If you select a single row or column, Excel creates one named range; if you select multiple rows and columns, it creates several named ranges. Importantly, the label cells themselves are not included in the range reference. By default, the scope of these names is set to the entire workbook.

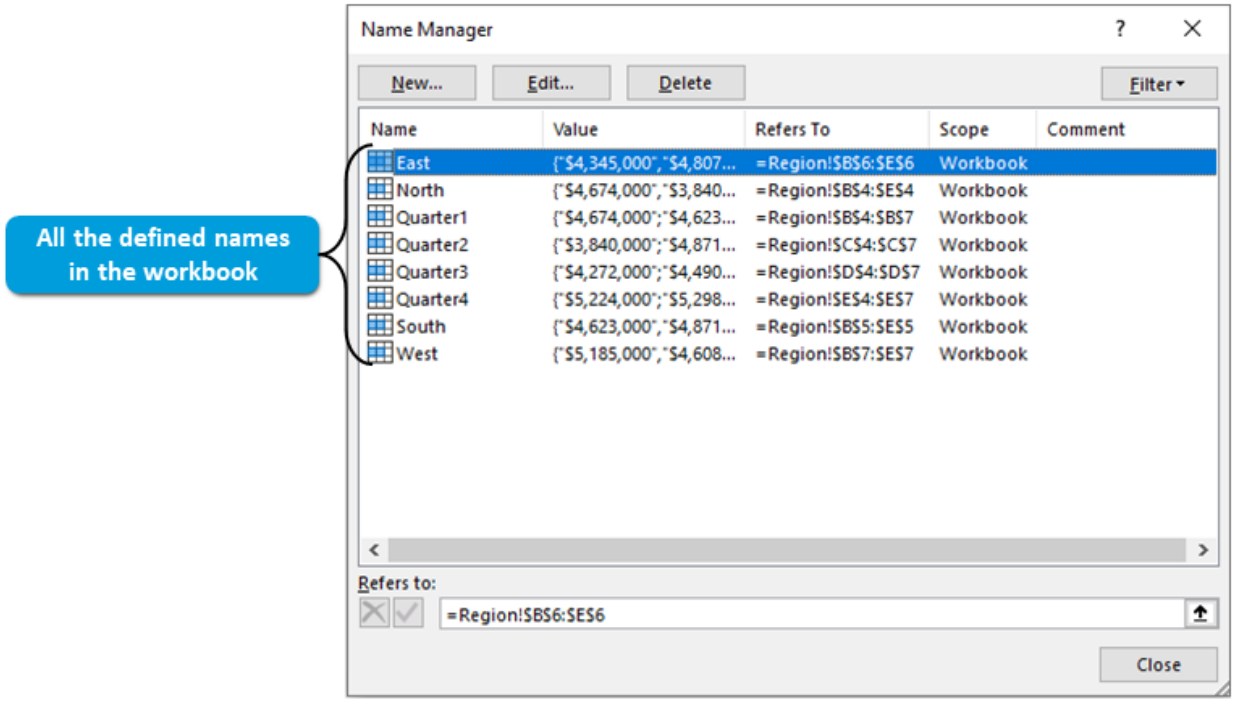
- You can access this feature from the **Formulas tab → Defined Names group → Create from Selection**, and it's especially useful for organizing large datasets.
- **Shortcut: Ctrl + Shift + F3**



The Name Manager Dialog Box

The Name Manager dialog box in Excel 2021 is the central tool for managing all named cells and ranges in a workbook. It allows you to view, rename, edit, and delete existing names, as well as create new ones through the New Name dialog box. This is particularly useful when workbooks change dynamically—for example, when rows are added to a dataset and named ranges need updating. However, the Name Manager cannot change the scope of a name (worksheet vs. workbook); to adjust scope, you must delete the existing name and create a new one. It also includes a Filter command, which lets you display only names with workbook scope, worksheet scope, or those containing errors, making it easier to manage large sets of named ranges.

- You can access the Name Manager via **Formulas → Name Manager**.
- **Shortcut: Ctrl + F3** to open the Name Manager dialog box.



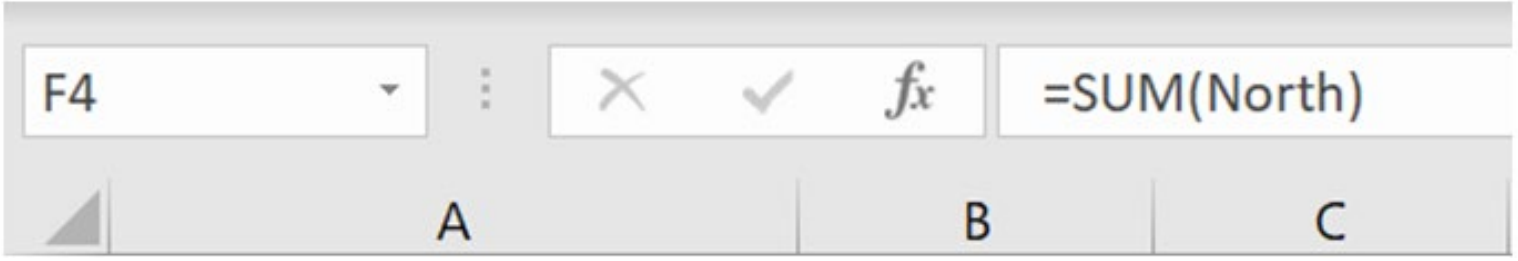
Activity 06.Naming and Editing Ranges

Cell and Range Names in Formulas

The real benefit of **named ranges** in Excel goes beyond navigation—it lies in simplifying formulas. Once a name is defined, you can use it in place of cell references, making formulas easier to read, faster to write, and less error-prone. Excel offers several ways to insert these names into formulas: you can **type the name manually**, use the **Use in Formula command**, or rely on **Formula AutoComplete** to quickly select from available names. These methods streamline working with complex datasets and ensure consistency across multiple formulas.

Shortcuts / Methods:

- **Manually type** the defined name in a formula.
- Use **Formulas → Use in Formula** to insert names.
- Use **Formula AutoComplete** while typing for quick selection.



Manually Entering Cell or Range Names

The most direct method for including cell or range names instead of references in formulas or functions is to simply type them. Wherever you would normally enter a cell or range reference, you can type a defined name instead. The formula will reference the cell or range by name, just as it would if you typed the cell or range reference, and your calculation results will be the same.

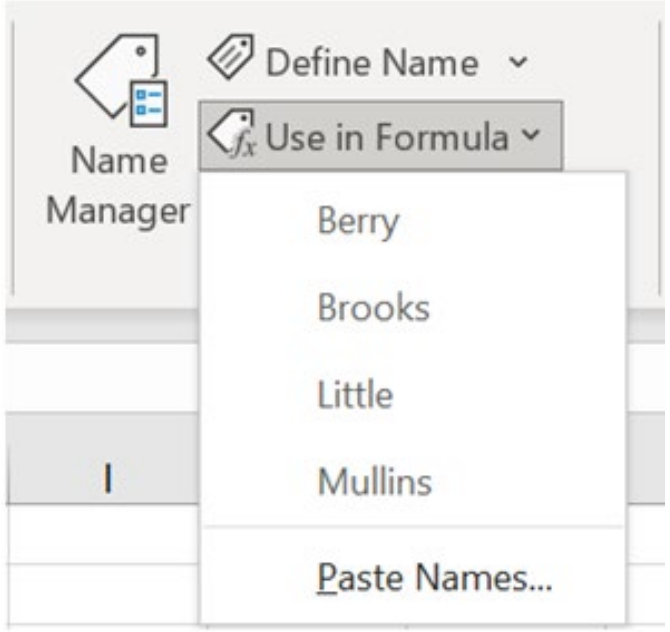
- It is important to note that you can still type the cell or range references for a named cell or range in a formula, and they will still appear as cell or range references.

You can also manually select a cell or range that you've applied a name to directly on a worksheet to enter it into a formula, just as you would with any unnamed range or cell. When you do this, Excel automatically displays the name, not the reference, though.

The Use in Formula Command Method

The **Use in Formula command** in Excel 2021 provides a convenient way to insert defined cell or range names directly into formulas and functions without manually typing them. Accessible from **Formulas → Use in Formula**, this command displays a drop-down list of all available names, allowing you to quickly select the one you need. Additionally, the drop-down includes the **Paste Names** option, which opens the Paste Name dialog box for easier selection. This feature not only speeds up formula creation but also reduces errors compared to manual entry.

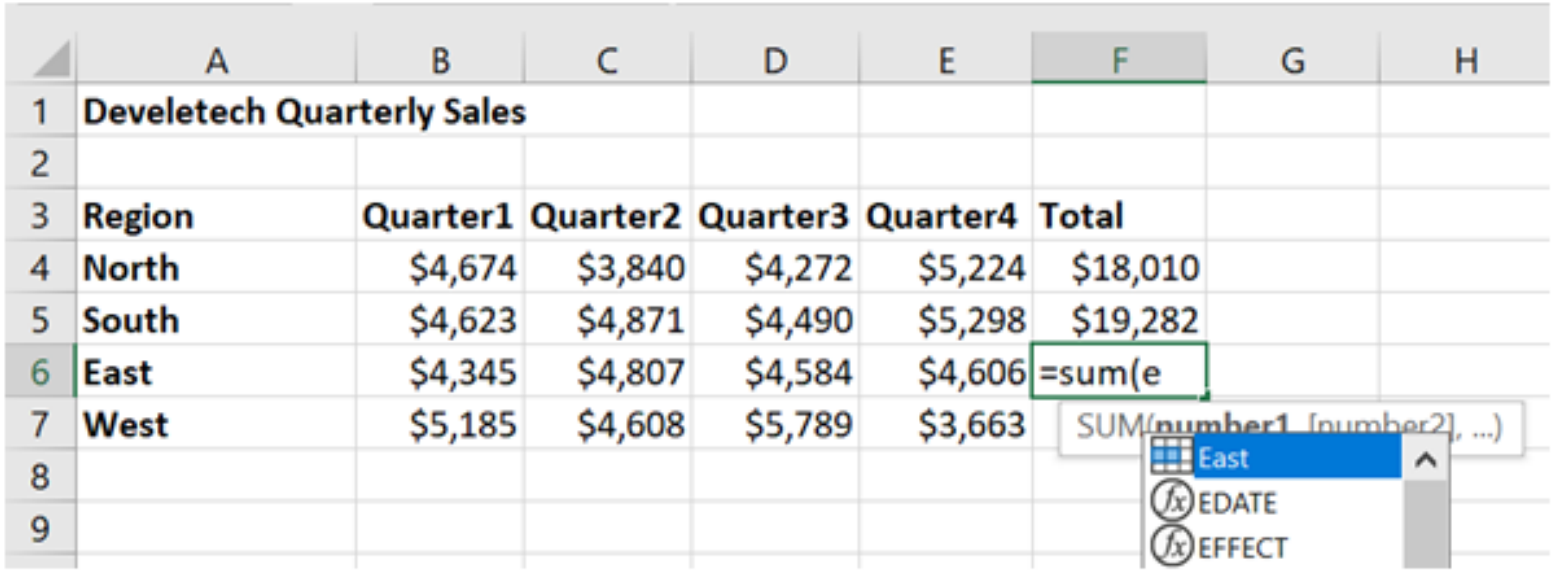
Shortcut: F3 to open the *Paste Name* dialog box instantly.



The Formula AutoComplete Method

The **Formula AutoComplete feature** in Excel doesn't just help with functions-it also works with **named cells and ranges**. As you type a formula in a cell or the Formula Bar, AutoComplete displays a pop-up menu showing valid names that match what you've started typing. You can then select the desired name instead of typing it fully, which saves time and reduces errors. The menu filters available names just like it does for functions, and you can distinguish between them by their icons: functions show the **Insert Function icon**, while defined names show a **highlighted spreadsheet row icon**. Once selected, you simply continue building the formula as usual.

- **Shortcut / Tip:** Just start typing a defined name in a formula, and **Formula AutoComplete** will suggest it automatically.



Activity 07.Using Defined Names in a Formula

Use Specialized Functions

Function Categories

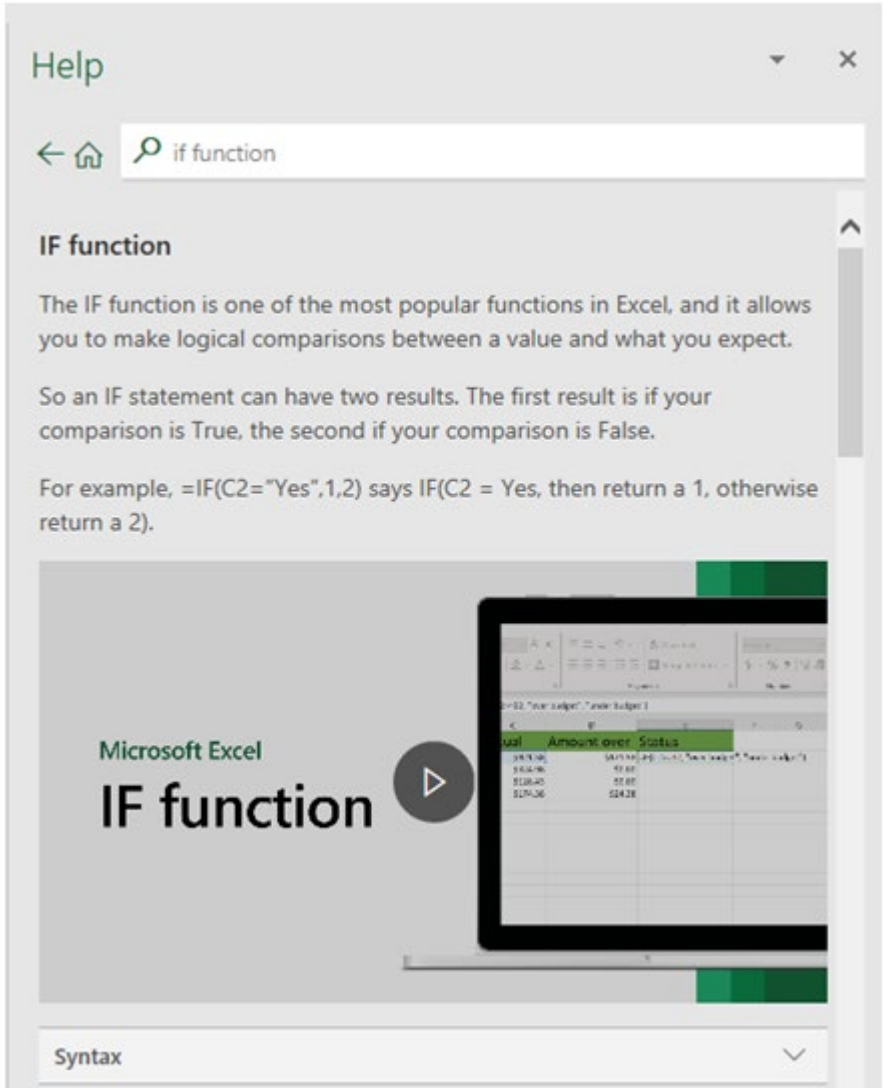
You will find every built-in Excel function in the **Function Library** group on the **Formulas** tab. Here, the vast collection of available functions is organized into task-related categories. These categories can be expanded by installing certain Excel add-ins.

- You have to access several of these categories via the **More Functions** drop-down menu in the **Function Library** group, as well as by selecting the **Insert Function** command.

The Excel Function Reference

The **Excel function reference** is a built-in Help resource that makes it easy to learn about functions you're unfamiliar with or to identify which function suits a particular task. It lists all functions by category and provides detailed entries that include a description of the function's purpose, its syntax and arguments, special considerations, and practical examples. This resource is especially useful when working with complex formulas or exploring new functions.

- You can access the function reference by searching for functions through the **Search field on the ribbon** or the **Search help field in the Help task pane**.
- **Shortcut: F1** to open the *Help task pane* and access the function reference.



Comparison Operator Basics

In Excel, **comparison operators** are mathematical symbols used to compare values (e.g., greater than, less than, equal to). They form the foundation for many functions and features, allowing you to set conditions and evaluate data. Understanding these operators is essential before diving deeper into function syntax.

Comparison Operator	Meaning
=	Equal to
<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to
<>	Not equal to

Function Syntax

As you progress in Excel, understanding **function syntax** becomes essential. Syntax defines the structure of a function, including its required and optional arguments. Building familiarity with commonly used functions reduces the need to constantly look them up and helps streamline workbook development.

Key Points:

- **Bold arguments** = required.
- **Square brackets [ ]** = optional.
- **Arguments** are always separated by **commas**.

➢ This foundation makes it easier to expand your knowledge of Excel's most frequently used functions and apply them efficiently.

- The **criteria** argument must be enclosed in quotation marks (" ") if it contains text, mathematical operators, or comparison operators. This is common among the various functions that contain the **criteria** argument.

The COUNTIF Function

Syntax: =COUNTIF(range,criteria)

- Use COUNTIF, one of the statistical functions, to count the number of cells that meet a criterion. For example, suppose you have a list of customers and one of the columns includes the city where each customer is located. With the COUNTIF function, you can count the number of times a particular city appears in the list.

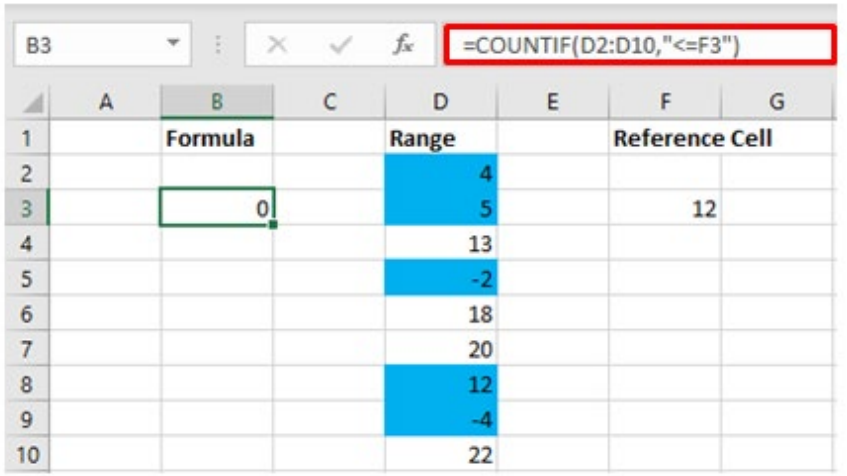
Criteria Argument Syntax

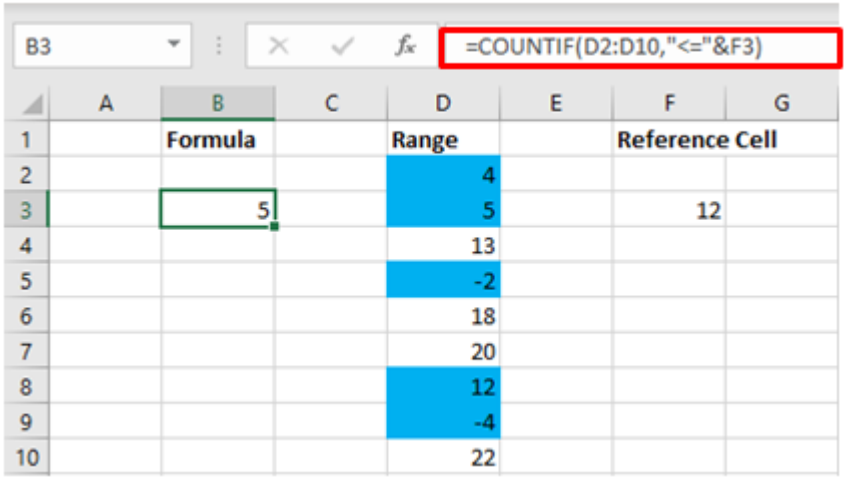
In Excel, the criteria argument for COUNTIF must be written as a text string in quotation marks, such as "< =12" to count values less than or equal to 12.

However, if you include a cell reference inside the quotes (like "< =F3"), Excel treats it as literal text rather than a reference, which returns no matches. The correct way is to concatenate the operator with the cell reference using the ampersand (&), for example:

=COUNTIF(A1:A10,"<="&F3)

This makes the criteria dynamic - if F3 contains 12, Excel counts cells less than or equal to 12; if you change F3 to 100, it instantly updates to count cells less than or equal to 100.





Automatic Workbook Calculation

By default, Excel 2021 uses **automatic calculation**, meaning formulas update instantly when source data changes. While this is convenient, in very large workbooks with thousands of rows and interdependent formulas, recalculation can slow performance and interrupt workflow. In such cases, you can switch to **manual calculation mode**, make your edits, and then update results only when needed using the **Calculate Now** command on the Formulas tab.

Calculation Option	Description
Automatic	Recalculates all dependent formulas every time you make a change to a value, formula, or name. This is the default calculation setting.
Automatic Except for Data Tables	Recalculates all dependent formulas, except data tables, every time you make a change to a value, formula, or name.
Manual	Turns off automatic recalculation and recalculates open workbooks only when you explicitly do so.
Calculate Now	Manually recalculates all open worksheets, including data tables, and updates all open chart sheets when **Manual** calculation is selected.
Calculate Sheet	Manually recalculates the active worksheet and any charts and chart sheets linked to the active worksheet.

Activity 08.Locating and Using Specialized Functions

Work with Logical Functions

Logical Functions

Logical functions in Excel are the foundation of data-driven decision making. They allow you to ask questions of your data and return either TRUE or FALSE, which can then be used to trigger calculations or apply different formulas depending on conditions. By incorporating logical tests into your formulas, you move beyond raw data and start building dynamic, rule-based analysis.

Key Idea:

- Logical functions (like IF, AND, OR, NOT) evaluate conditions.
- They return TRUE/FALSE values that can drive calculations.
- This enables Excel to perform different actions depending on whether criteria are met.

- In other words, logical functions transform Excel from a simple calculator into a decision-making tool for deeper analysis.

Logical Values

In Excel, **logical values** are a special data type returned by logical tests, and they can only be **TRUE** or **FALSE**. These values form the foundation of logic in computing and are essential for working with comparison operators and logical functions.

Key Points:

- Logical values always appear in **uppercase** (TRUE, FALSE). If you type “true” or “false,” Excel automatically converts them.
- They behave like numbers in formulas: **TRUE = 1**, **FALSE = 0** in certain contexts.
- They can be used as **arguments in functions** or as **results of logical tests**, helping establish whether criteria are met.

➢ This makes logical values powerful tools for building conditions and driving decision-making in formulas.

- The IF, AND, and OR functions discussed later in this topic all perform logical tests and can result in either TRUE or FALSE.

Comparison Operators

A type of Excel operator used to compare particular values to determine whether or not they meet some specified criteria.

Name	Comparison Operator	This Comparison Operator Determines Whether or Not
Equal to	=	The specified values are the same.
Greater than	>	The first value is greater than the second value.
Less than	<	The first value is less than the second value.
Greater than or equal to	>=	The first value is greater than or equal to the second value.
Less than or equal to	<=	The first value is less than or equal to the second value.
Not equal to	<>	The specified values are different.

IF Function

The **IF function** in Excel lets you perform conditional logic by returning one value if a condition is true and another if it's false.

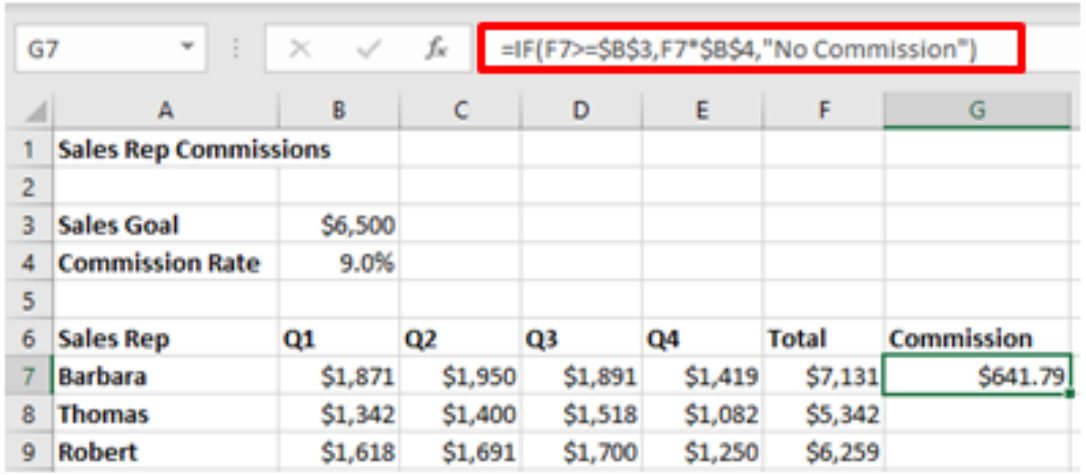
Syntax: =IF(logical_test, value_if_true, value_if_false)

- **logical_test** → the condition you want to check (e.g., Sales>=6500).
- **value_if_true** → the result if the condition is met (e.g., Sales*9%).
- **value_if_false** → the result if the condition is not met (e.g., 0).

In this example, the **IF function** evaluates whether the sales value in cell **F7** meets or exceeds the goal in **B3**.

- If **TRUE** (sales ≥ goal), Excel multiplies F7 by the commission rate in **B4** and returns the calculated commission.
- If **FALSE** (sales < goal), Excel displays the text **"No Commission"**.

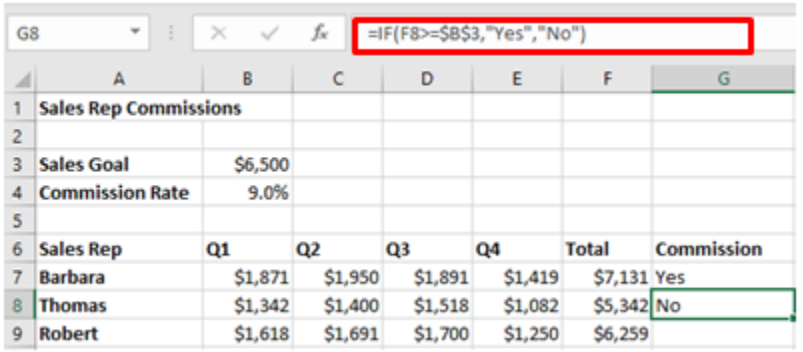
When copied to the next row, the formula adapts: for **F8**, since sales are below the goal, the logical test returns FALSE, and Excel outputs **"No Commission"** instead of a calculation.



	A	B	C	D	E	F	G
1	Sales Rep Commissions						
2							
3	Sales Goal	$6,500					
4	Commission Rate	9.0%					
5							
6	Sales Rep	Q1	Q2	Q3	Q4	Total	Commission
7	Barbara	$1,871	$1,950	$1,891	$1,419	$7,131	$641.79
8	Thomas	$1,342	$1,400	$1,518	$1,082	$5,342	
9	Robert	$1,618	$1,691	$1,700	$1,250	$6,259	

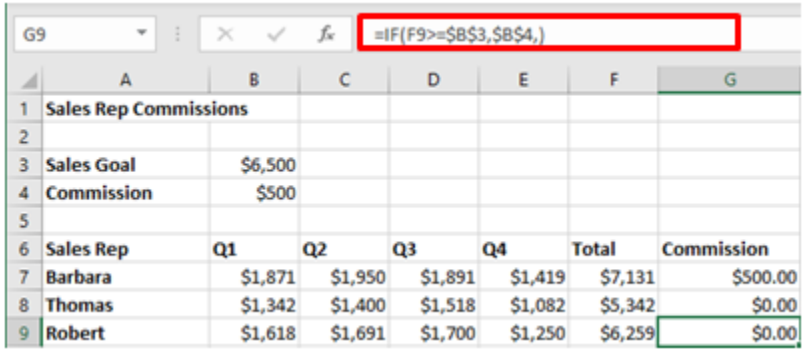
IF Function

Keep in mind that you don't always need an IF function to perform a calculation. You could simply use it to answer the question, "Does each sales rep get a commission?" Here's what you would enter.



	A	B	C	D	E	F	G
1	Sales Rep Commissions						
2							
3	Sales Goal	$6,500					
4	Commission Rate	9.0%					
5							
6	Sales Rep	Q1	Q2	Q3	Q4	Total	Commission
7	Barbara	$1,871	$1,950	$1,891	$1,419	$7,131	Yes
8	Thomas	$1,342	$1,400	$1,518	$1,082	$5,342	No
9	Robert	$1,618	$1,691	$1,700	$1,250	$6,259	

Or, you can simply ask the IF function to return the value in a particular cell if the condition is met. In this last example, assume the sales reps get a flat $500 commission only if their sales exceed $6,500.



	A	B	C	D	E	F	G
1	Sales Rep Commissions						
2							
3	Sales Goal	$6,500					
4	Commission	$500					
5							
6	Sales Rep	Q1	Q2	Q3	Q4	Total	Commission
7	Barbara	$1,871	$1,950	$1,891	$1,419	$7,131	$500.00
8	Thomas	$1,342	$1,400	$1,518	$1,082	$5,342	$0.00
9	Robert	$1,618	$1,691	$1,700	$1,250	$6,259	$0.00

Here, because the **value_if_true** argument contains a cell reference, the function returns the value in cell **B4** when the logical test returns the value TRUE. Also, as the **value_if_false** argument has been left off, the function returns a value of zero (0) in cases where the logical condition was not met.

Functions Similar to the IF Function

There are several other useful functions that can perform calculations based on logical comparisons.

Function Name	Function Definition	Function Arguments
SUMIF	You use the SUMIF function to sum the values in a range that meet criteria that you specify.	SUMIF(**range**,**criteria**, [sum_range])
SUMIFS	The SUMIFS function, one of the math and trig functions, adds all of its arguments that meet multiple criteria.	SUMIFS(**sum_range**,**criteria_range1**, **criteria1**,[criteria_range2,criteria2],...)
COUNTIF	Use COUNTIF, one of the statistical functions, to count the number of cells that meet a criterion.	COUNTIF(**range**,**criteria**)
COUNTIFS	Use COUNTIFS to count cells using multiple criteria.	COUNTIFS(**criteria_range1**,**criteria1**, [criteria_range2,criteria2],...)
AVERAGEIF	Use the AVERAGEIF function to return the average of all the cells in a range that meet a given criterion.	AVERAGEIF(**range**,**criteria**,[average_range])
AVERAGEIFS	Use the AVERAGEIFS function to return the average of all the cells that meet multiple criteria.	AVERAGEIFS(**average_range**,**criteria_range1**,**criteria1**,[criteria_range2,criteria2],...)

AND Function

The AND function in Excel evaluates multiple conditions and returns TRUE only if all arguments are true; if even one is false, the result is FALSE.

Syntax: =AND(logical1,[logical2],...,[logical30])

- logical1 → the first condition to test (required).
- logical2–logical30 → optional additional conditions.
- Up to 30 arguments can be tested.
- Arguments can be logical statements, cell references, ranges, or even mathematical expressions (e.g., 1+1=2 returns TRUE).

	A	B	C	D
1	10		**AND Function Examples**	**Formula Result**
2	15		=AND(A1<A2,A2<A3)	TRUE
3	20		=AND(A1<A2,A1>A3)	FALSE
4			=AND(A1<>A2,A1*2=A3)	TRUE

OR Function

The OR function works just like AND, but it only requires one condition to be true for the result to be TRUE. If all conditions are false, it returns FALSE.

Syntax:=OR(logical1,[logical2],...,[logical30])

- Supports up to 30 arguments.
- Arguments can be logical tests, cell references, ranges, or mathematical statements.
- Use OR when you want Excel to confirm if at least one requirement is met, such as checking if a sales rep achieved any of several targets.

	A	B	C	D
1	10		**OR Function Examples**	**Formula Result**
2	15		=OR(A1<A2,A2>A3)	TRUE
3	20		=OR(A1>A2,A1>A3)	FALSE
4			=OR(A1=A2,A1*2=A3)	TRUE

NOT Function

The **NOT function** in Excel reverses the result of a logical test: if the test is **TRUE**, NOT returns **FALSE**; if the test is **FALSE**, NOT returns **TRUE**. It's often used to flip conditions or adjust the behavior of other logical functions.

Syntax: =NOT(logical1)

	A	B	C	D
1	10		**NOT Function Examples**	**Formula Result**
2	15		=NOT(A1>0)	FALSE
3	20		=NOT(AND(A1<A2,A2<A3))	FALSE
4			=NOT(OR(A1>A2,A2>A3))	TRUE

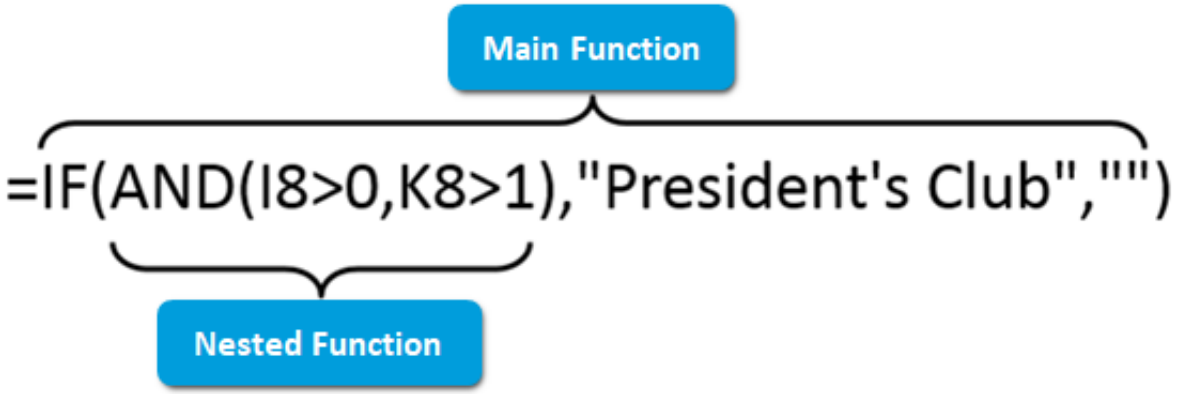
Activity 09.Working with Logical Functions

Nesting

In Excel, nesting means placing one function inside another as an argument, so the result of the inner function becomes input for the outer function. This allows you to build complex formulas that combine multiple calculations or logical tests in a single cell.

Key Points:

- You can nest multiple functions within one another.
- Excel 2021 supports up to 64 levels of nesting.
- A function inside another is called a second-level function, inside that is third-level, and so on.
- Nesting is what makes formulas like =IF(AND(A1>5,A2<20),"Pass","Fail") possible, where AND is nested inside IF.



Nested Function Syntax

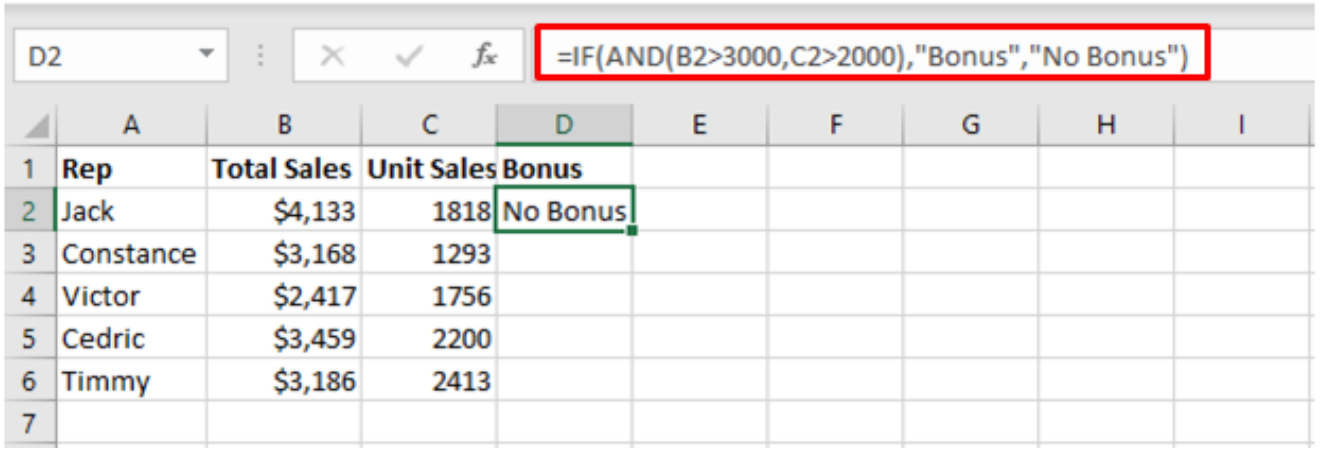
The key to nested functions in Excel is understanding how their syntax layers together. Each nested function must return the correct data type for the argument it's filling, and every function must have its own complete set of parentheses. You only place the equal sign before the *first-level function*, not the nested ones

Example 1 – Nested AND inside IF

Here's how the formula works when checking if sales reps qualify for a bonus based on *two criteria* (sales > 3000 and units sold > 2000):

Breakdown of the syntax:

- AND(F2>3000,G2>2000) → This nested function checks both conditions.
  - If both are TRUE, AND returns TRUE.
  - If either is FALSE, AND returns FALSE.
- IF(...,"Bonus","No Bonus") → The IF function then uses the result of AND:
- If TRUE → returns "Bonus".
- If FALSE → returns "No Bonus".



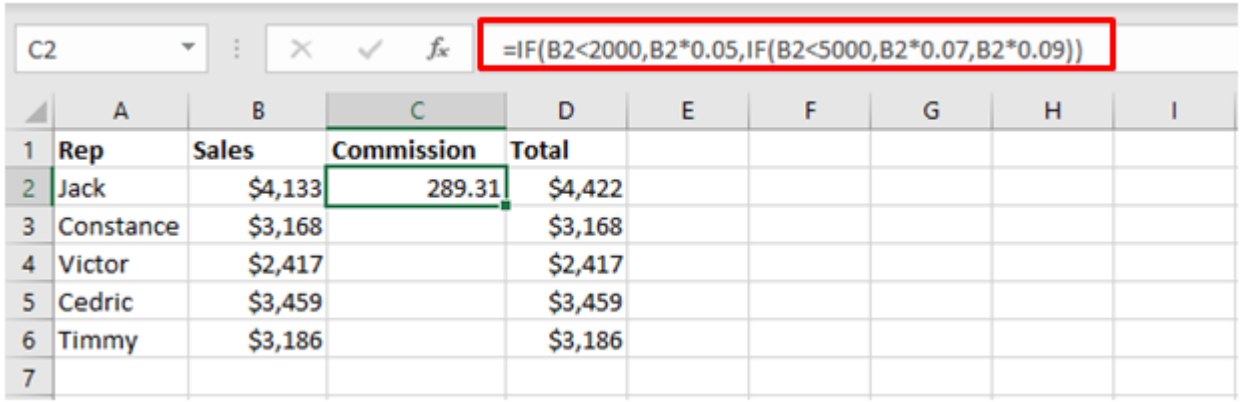
	A	B	C	D	E	F	G	H	I
1	Rep	Total Sales	Unit Sales	Bonus					
2	Jack	$4,133	1818	No Bonus					
3	Constance	$3,168	1293						
4	Victor	$2,417	1756						
5	Cedric	$3,459	2200						
6	Timmy	$3,186	2413						
7									

Nested Function Syntax

Here's how you can structure the formula when commission depends on three different sales thresholds: =IF(F2<2000,F2*5%,IF(F2<5000,F2*7%,F2*9%))

Breakdown of the syntax:

- **First IF (outer function):**
  - Logical test → F2<2000
  - If TRUE → commission = F2*5%
  - If FALSE → Excel moves to the second IF.
- **Second IF (nested inside the first):**
  - Logical test → F2<5000
  - If TRUE → commission = F2*7%
  - If FALSE → commission = F2*9%.
- **How it works in practice:**
  - Sales < 2000 → 5% commission.
  - Sales between 2000 and 4999 → 7% commission.
  - Sales ≥ 5000 → 9% commission.



	A	B	C	D	E	F	G	H	I
1	Rep	Sales	Commission	Total					
2	Jack	$4,133	289.31	$4,422					
3	Constance	$3,168		$3,168					
4	Victor	$2,417		$2,417					
5	Cedric	$3,459		$3,459					
6	Timmy	$3,186		$3,186					
7									

Guidelines for Combining Functions

Combining Functions with Nesting in Excel

When you start nesting functions, the key is to keep the syntax clean and logical. Each function must have its own complete set of parentheses, and the nested function must return the correct data type for the argument it's filling. Excel allows up to **64 levels of nesting**, which means you can build very sophisticated formulas—but clarity is essential.

Guidelines Recap

- ✅ Equal sign only before the **first-level function**.
- ✅ No equal sign for nested functions.
- ✅ Every function must have its own parentheses.
- ✅ Commas inside nested functions only separate arguments for that function.
- ✅ Nested functions must return the correct data type for the argument.
- ✅ You can combine nested functions with other calculations (e.g., multiply by constants or cell values).
- ✅ Multiple nested functions can exist within a single argument.

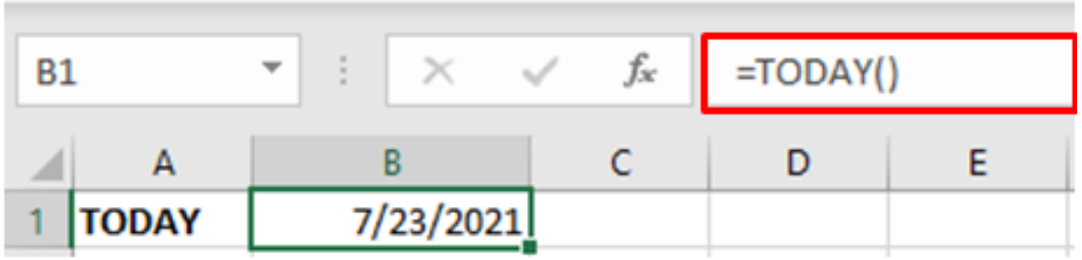
Activity 10. Combining Functions

Work with Date and Time Functions

The TODAY Function

Syntax: =TODAY()

- This function enters the current date in a cell. Unlike other functions, the TODAY function does not have any arguments. This functions result is termed volatile, which means that it changes every time the worksheet recalculates. For example, you may need to calculate the difference between a given date (order date, shipping date, or hire date) and today's date. Each time the workbook is opened, the function updates to the current date, thus updating the difference between the dates.

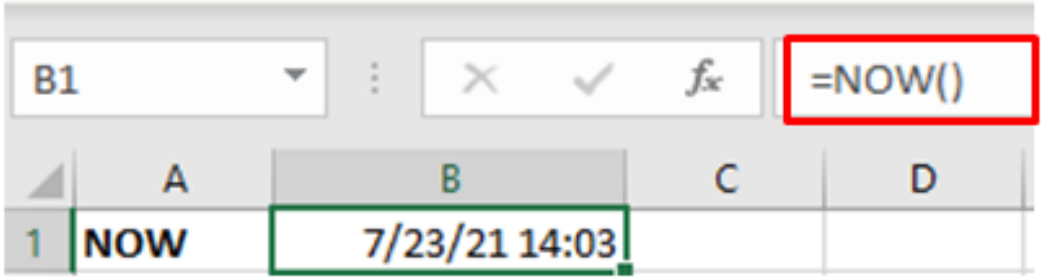


- If you do not wish the date to update when the workbook is opened, simply enter a static date or press **CTRL+;** to insert the current date.

The NOW Function

Syntax: =NOW()

Like the TODAY function, the NOW function has no arguments; it simply returns the current date and time in the cell you enter it into. You can use the value returned by this function to perform other calculations related to durations of time.



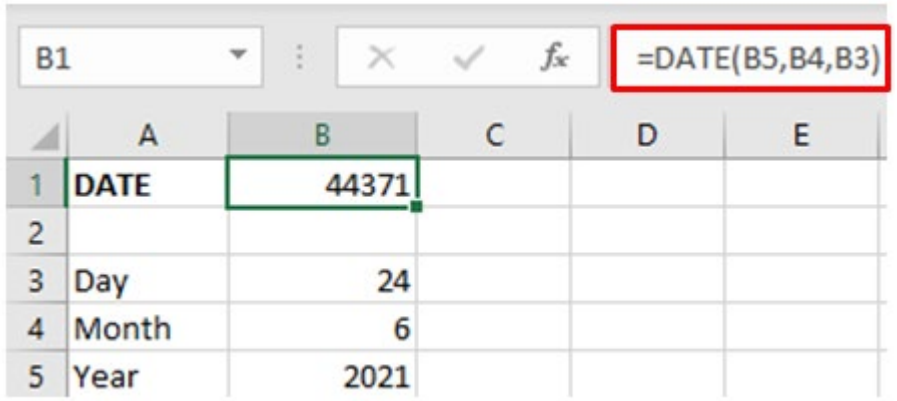
- When you enter the NOW function in a cell, Excel automatically formats the cell with a custom cell format used to accommodate both the date and the time. Here is the format: m/d/yyyy h:mm. Although the format displays only a single *m* for month and a single *d* for day, dates will appear in cells with both numbers for months and dates that contain two digits. If you alter the format to a different date or time format, you will alter the value in the cell.

The DATE Function

Syntax: =DATE(Year,Month,Day)

The **DATE function** in Excel is a powerful way to construct valid dates for calculations, rather than typing them directly as text. While it technically returns the **serial number** of the date (Excel's internal way of storing dates), it displays the result in the date format applied to the cell.

- **Year** → a four-digit year (e.g., 2026).
- **Month** → a number from 1 to 12 (January = 1, December = 12).
- **Day** → the day of the month (1–31).



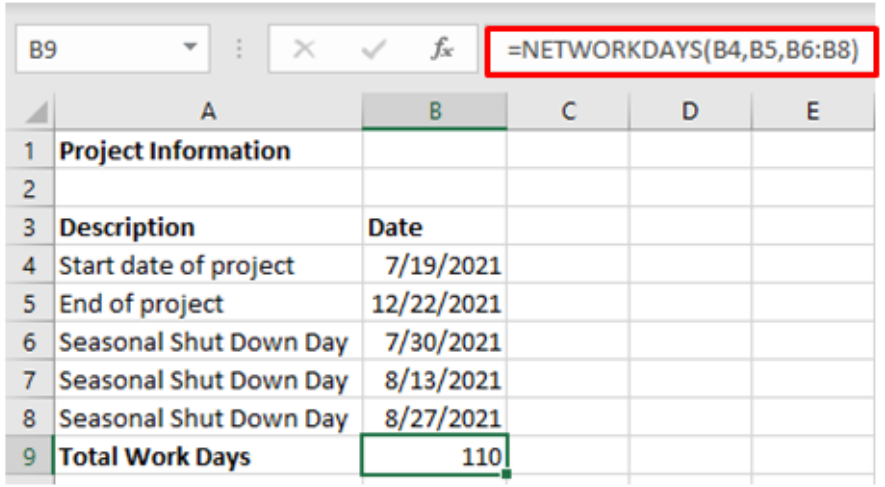
- Cell references for year, month, and day can be used instead of a static year, month, and day. This gives you the ability to change any of the arguments to suit your needs.

The NETWORKDAYS Function

The **NETWORKDAYS function** is designed to calculate the number of working days between two dates, automatically excluding weekends (Saturday and Sunday by default). You can also specify holidays to exclude them from the count.

Syntax: =NETWORKDAYS(start_date, end_date, [holidays])

- **start_date** → the first date in the range.
- **end_date** → the last date in the range.
- **holidays** → optional; a range of cells containing holiday dates to exclude.



	A	B	C	D	E
1	**Project Information**				
2					
3	**Description**	**Date**			
4	Start date of project	7/19/2021			
5	End of project	12/22/2021			
6	Seasonal Shut Down Day	7/30/2021			
7	Seasonal Shut Down Day	8/13/2021			
8	Seasonal Shut Down Day	8/27/2021			
9	**Total Work Days**	110			

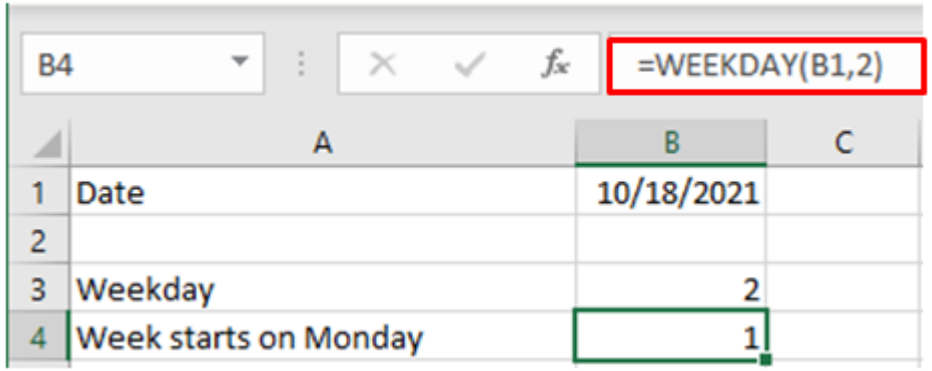
- In the United States, typical work days are Monday–Friday, and the weekend days are Saturday and Sunday. If you need to specify different working and weekend days, use the NETWORKDAYS.INTL function.

The WEEKDAY Function

The WEEKDAY function in Excel is used to determine the day of the week for a given date, returning an integer value. By default, Excel counts Sunday as 1 through Saturday as 7, but you can customize the numbering with the optional return_type argument.

Syntax:=WEEKDAY(serial_number,[return_type])

- serial_number → the date you want to evaluate (best entered with the DATE function or another date-returning formula).
- return_type → optional; determines which day is considered day 1.
- Return Type Options (most common):
  - 1 or omitted → Sunday = 1, Monday = 2, … Saturday = 7.
  - 2 → Monday = 1, Tuesday = 2, … Sunday = 7.
  - 3 → Monday = 0, Tuesday = 1, … Sunday = 6.

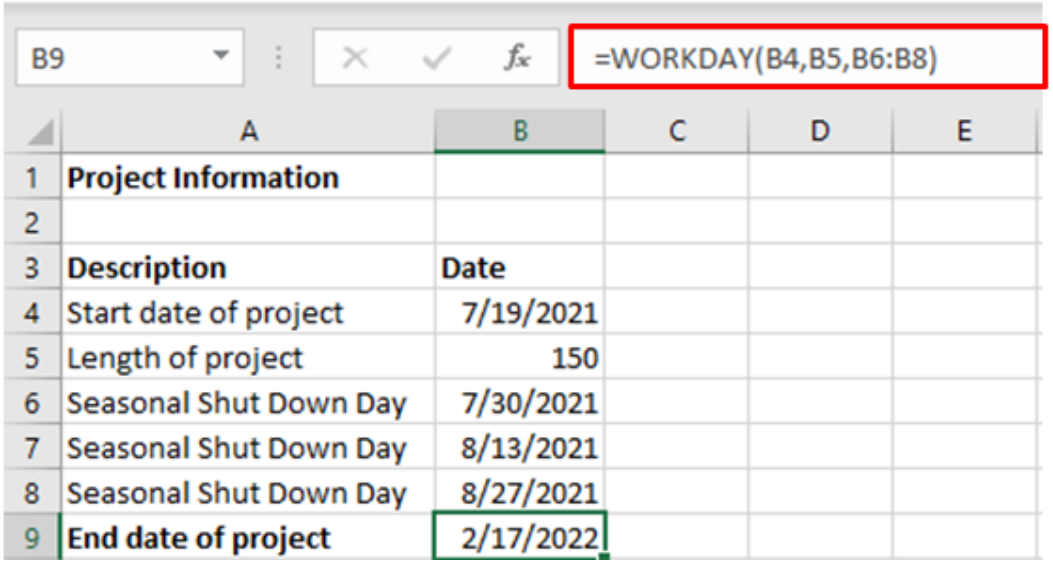


The WORKDAY Function

The **WORKDAY function** is designed to calculate a future (or past) date by adding a specified number of *working days* to a start date, automatically excluding weekends and any holidays you define.

Syntax: =WORKDAY(start_date, days, [holidays])

- **start_date** → the beginning date.
- **days** → the number of working days to add (positive for future, negative for past).
- **holidays** → optional; a range or array of dates to exclude (e.g., public holidays, company shutdowns).



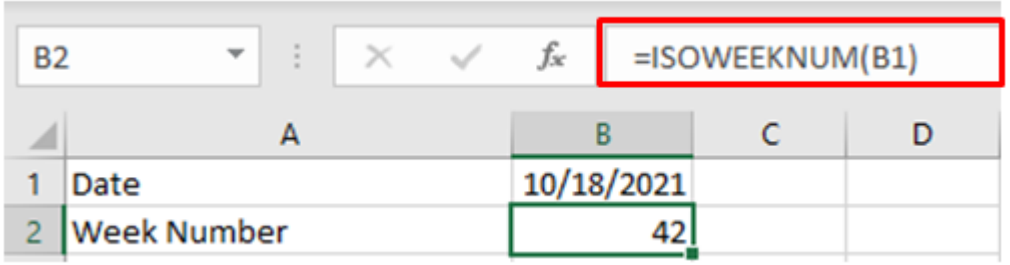
	A	B	C	D	E
1	**Project Information**				
2					
3	**Description**	**Date**			
4	Start date of project	7/19/2021			
5	Length of project	150			
6	Seasonal Shut Down Day	7/30/2021			
7	Seasonal Shut Down Day	8/13/2021			
8	Seasonal Shut Down Day	8/27/2021			
9	**End date of project**	2/17/2022			

The ISOWEEKNUM Function

The ISOWEEKNUM function is a specialized date function in Excel that returns the ISO week number for a given date. Unlike the standard WEEKNUM function, it follows the ISO 8601 standard, which defines weeks as starting on Monday, and designates Week 1 as the first week of the year that contains a Thursday.

Syntax: =ISOWEEKNUM(date)

- date → the date you want to evaluate (entered with the DATE function or another date-returning formula).
- Key Points:
- Weeks always begin on Monday.
- Week 1 is the week containing January 4th (since it's the first week with a Thursday).
- This makes ISO week numbering consistent across international standards, often used in manufacturing, logistics, and project planning.



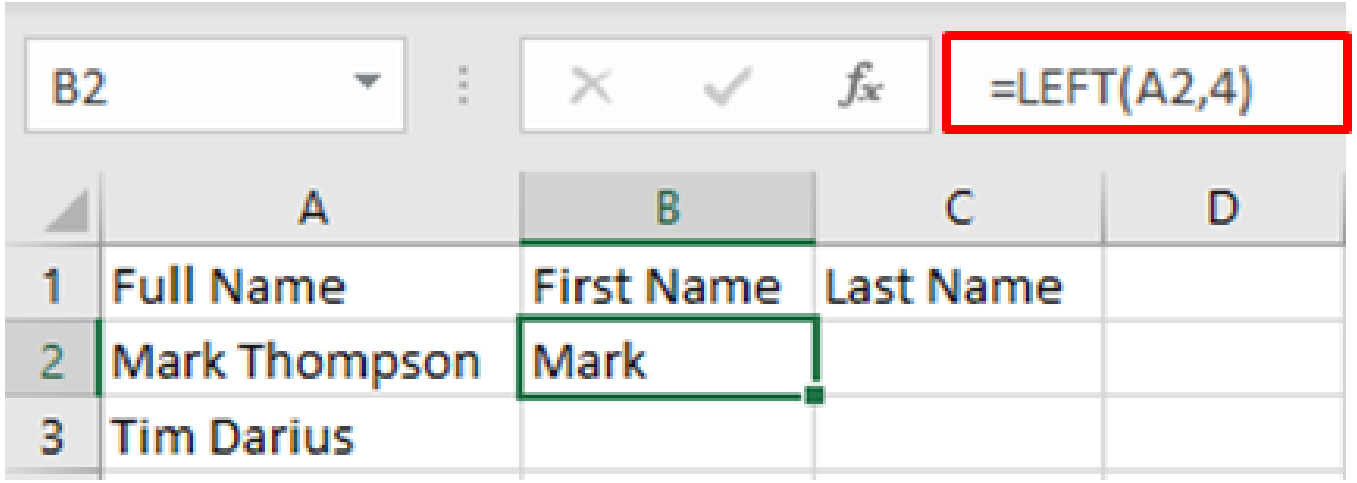
Activity 11. Working with Date and Time Functions

Work with Text Functions

The LEFT Function

Syntax: =LEFT(text,[num_chars])

The LEFT function returns the first character or characters in a text string, based on the number of characters you specify. For example, if the full name Mark Thompson was in cell **A2**, you could use the LEFT function in cell **B2** to extract the first four characters of that text, resulting in a cell with the first name Mark.

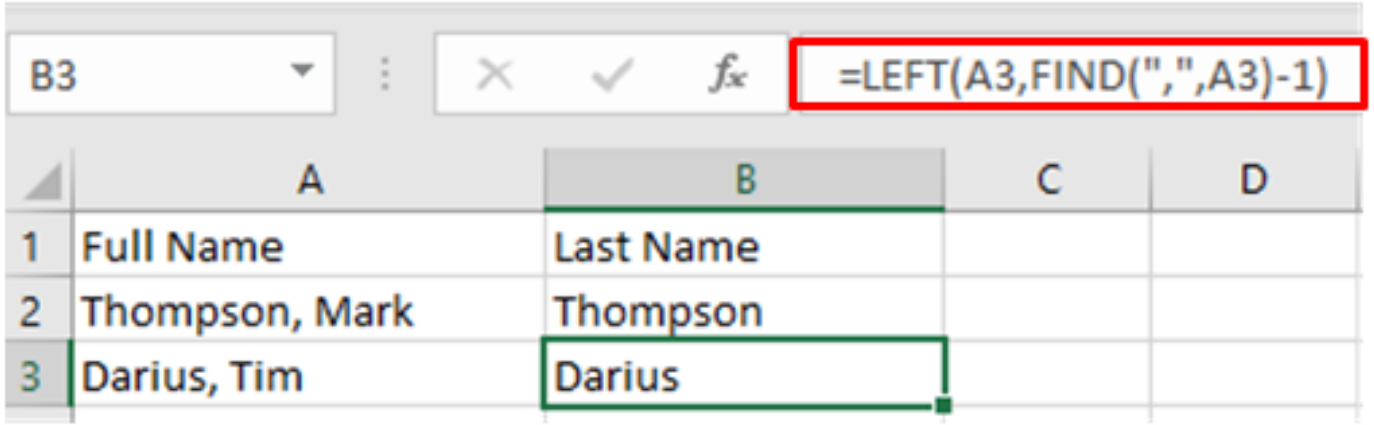


The FIND Function

The FIND function in Excel is used to locate the position of one text string within another, returning the starting character number of the found text. It's case-sensitive and doesn't allow wildcards, making it precise for text extraction tasks.

Syntax: =FIND(find_text, within_text, [start_num])

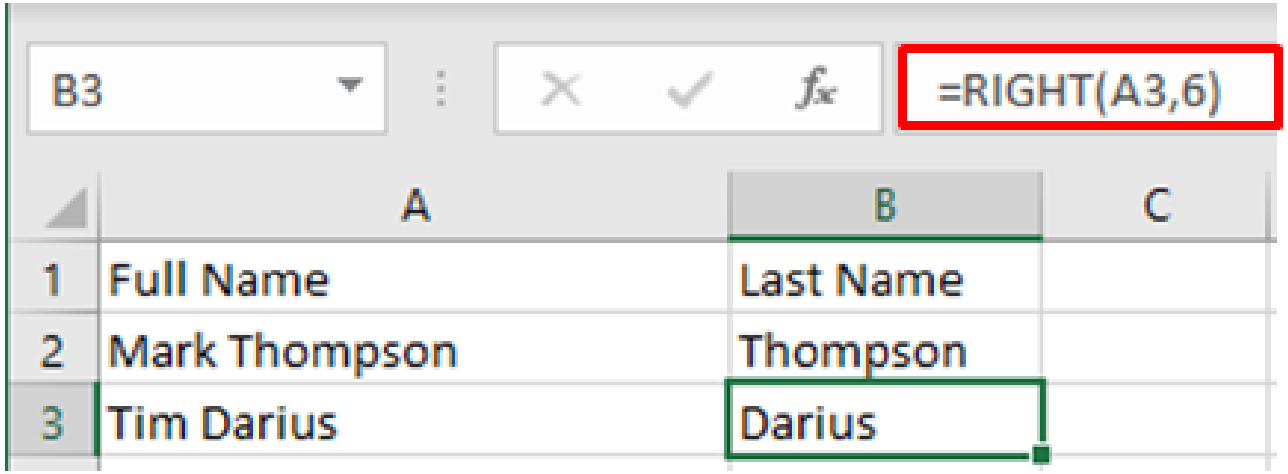
- find_text → the text you want to locate (e.g., a comma ",").
- within_text → the text string or cell containing the text to search.
- start_num → optional; the character position to begin searching (default is 1).



The RIGHT Function

Syntax: =RIGHT(text,[num_chars])

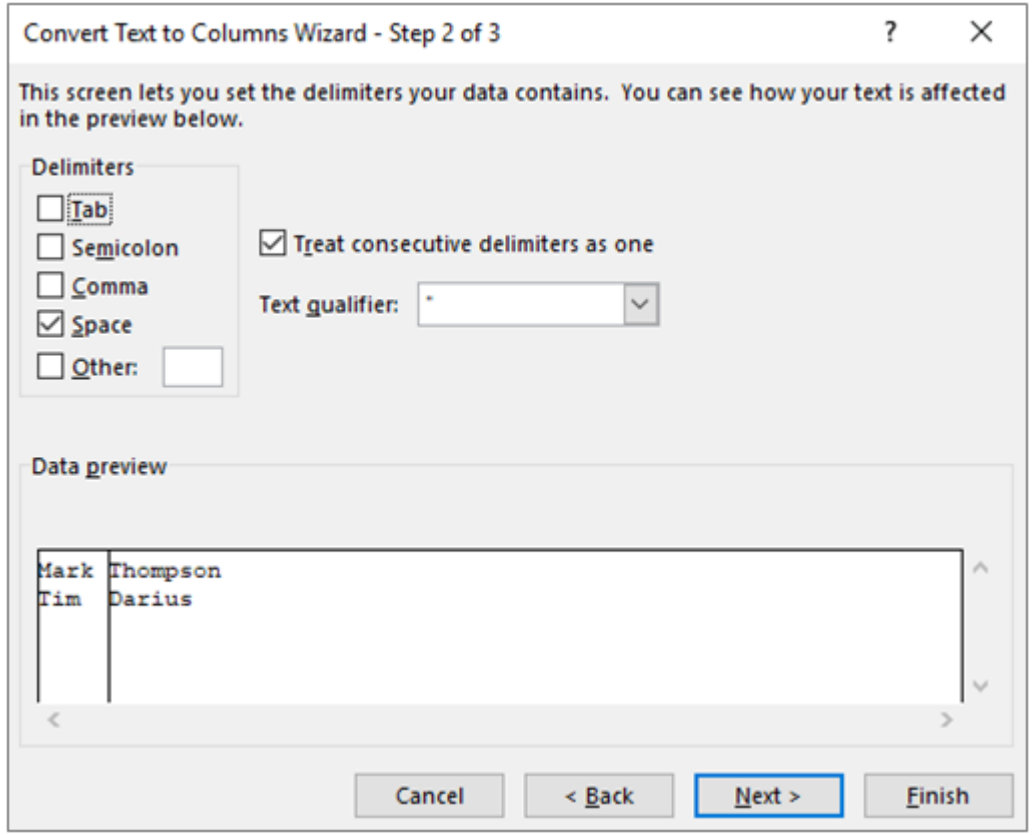
The RIGHT function returns the first character or characters in a text string, based on the number of characters you specify. For example, if the full name Timothy Darius was in cell **A3**, you could use the RIGHT function in cell **C3** to extract six characters from the right of that text, resulting in a cell with the last name Darius.



The Text to Columns Feature

The **Text to Columns** command in the **Data Tools** group on the **Data** tab of the ribbon is another method of splitting text. This feature splits a single column of text into multiple columns.

For example, you can create first name and last name columns from one column of full names. When used, this feature starts the Convert Text to Columns Wizard. The wizard's three steps let you select how the text should be treated as a group, what character separates the text, and the destination of the text in other columns.

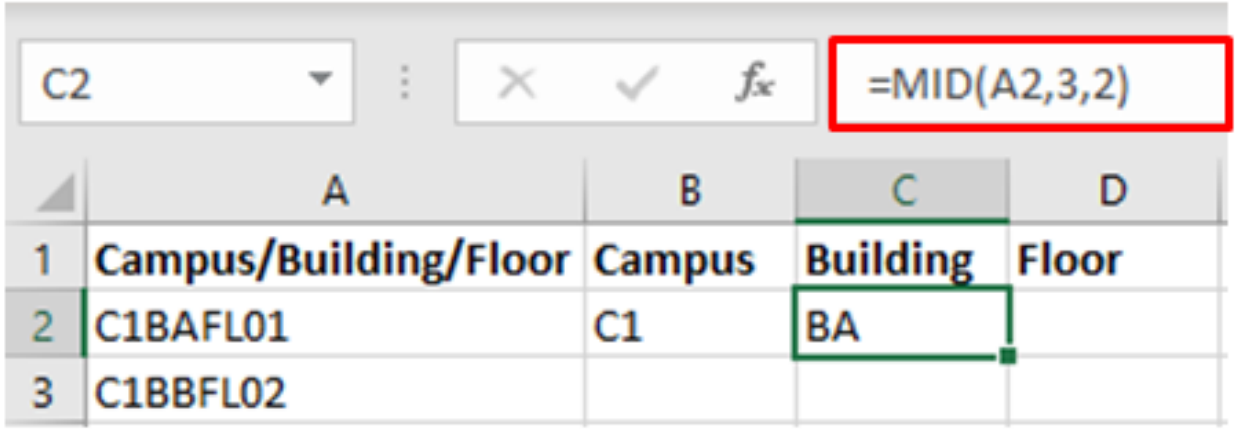


The MID Function

The **MID function** is perfect for extracting a specific portion of text from the middle of a string, based on the position you specify and the number of characters you want returned.

Syntax: =MID(text, start_num, num_chars)

- **text** → the cell or string containing the text.
- **start_num** → the position of the first character you want to extract (counting from the left).
- **num_chars** → the number of characters to return.

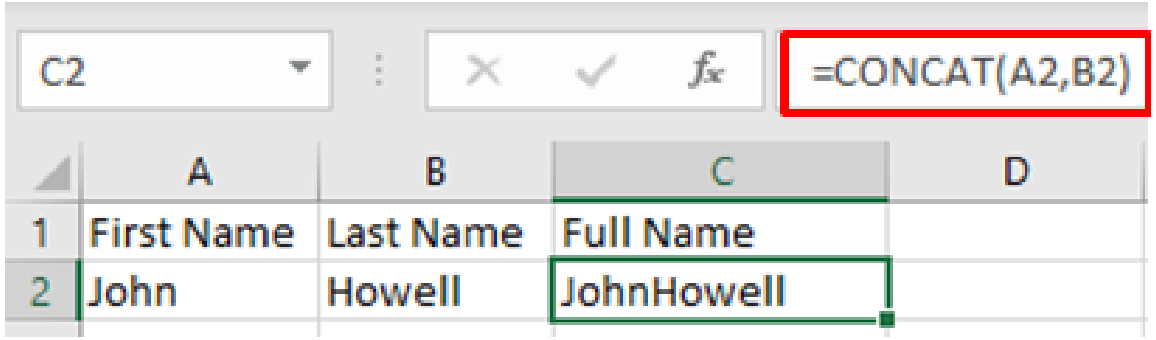


The CONCAT Function

The CONCAT function is Excel's modern way to join text strings together, replacing the older CONCATENATE function (still available for compatibility). It's especially useful when you need to merge data from multiple cells into one.

Syntax: =CONCAT(text1,[text2],...)

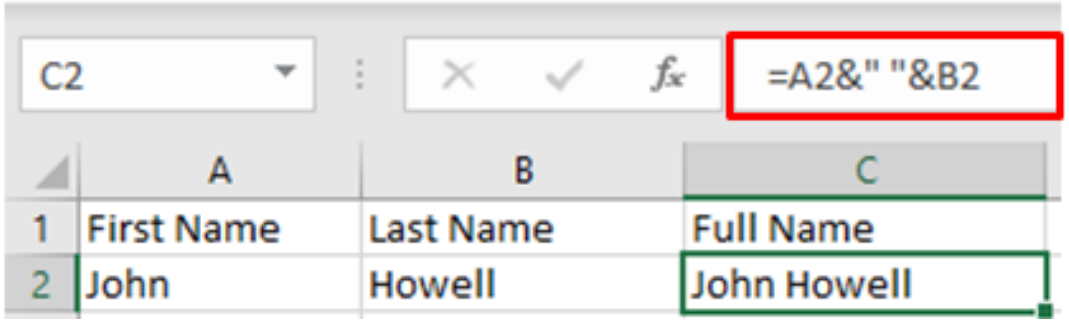
- text1 → required; the first text string or cell reference.
- text2...text255 → optional; additional text strings or cell references (up to 255 total).
- You can include spaces manually by adding " " as an argument.
- CONCAT preserves any spaces already present in the source cells.

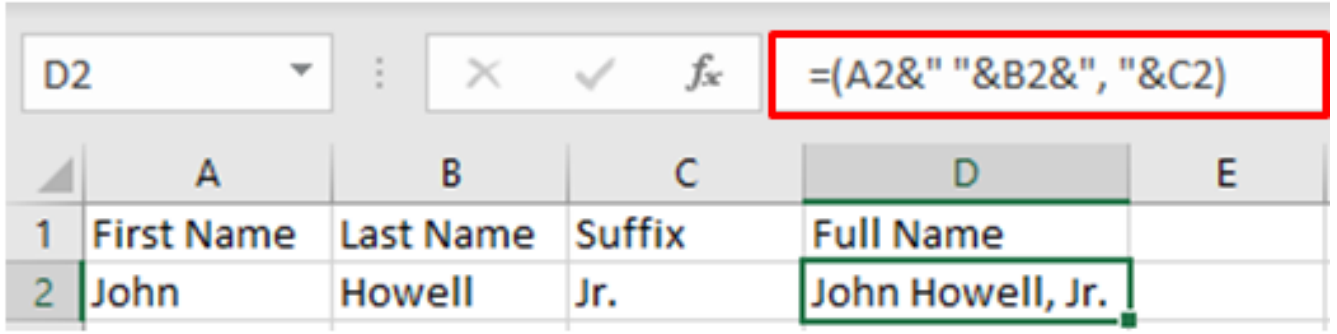


The CONCAT Function

Text Concatenation with the Ampersand

Remember, another way to concatenate text strings and numeric values from worksheet cells is to use the ampersand (&) operator in formulas. By using this method, you can include a mix of either cell references or text and values entered in double quotation marks to join text strings. This is how the aforementioned first and third examples would work if you used the ampersand operator in formulas, instead of the CONCAT function.





Other Text Functions

The UPPER Function

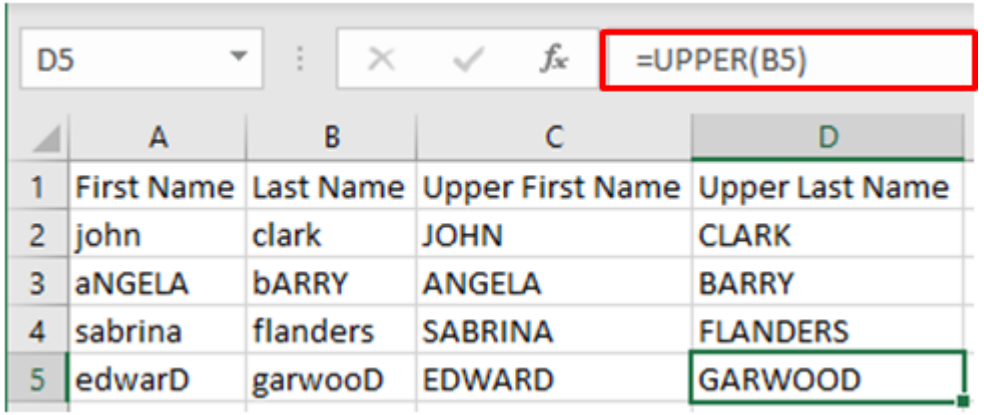
- Syntax: =UPPER(**text1**)
- Description: The UPPER function converts a text string to uppercase or capitalizes all the text in a string.

The LOWER Function

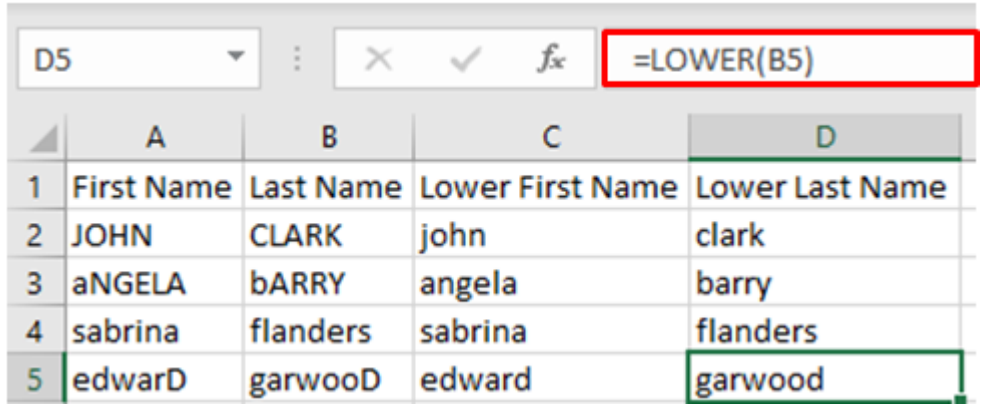
- Syntax: =LOWER(**text1**)
- Description: The LOWER function converts a text string to lowercase or makes all the text in a string lowercase. The LOWER function does not change characters in text that are not letters.

The PROPER Function

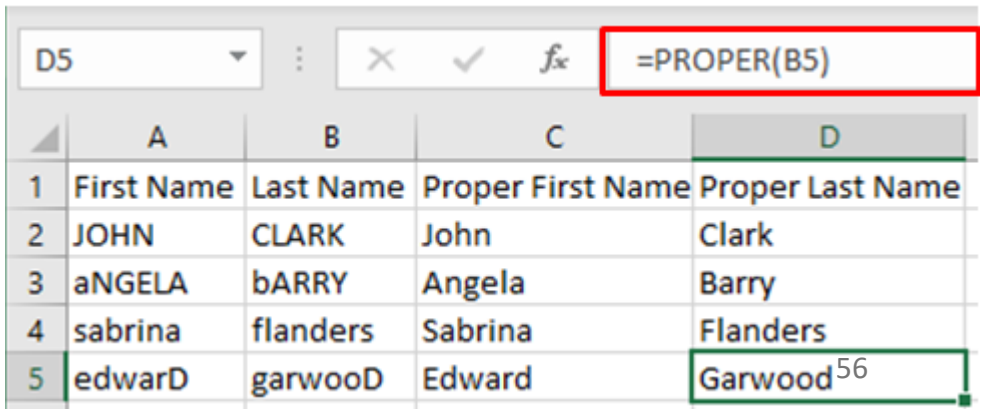
- Syntax: =PROPER(**text1**)
- Description: The PROPER function converts a text string to proper case, meaning that the function capitalizes the first letter in a text string and converts the other letters to lowercase.



	A	B	C	D
1	First Name	Last Name	Upper First Name	Upper Last Name
2	john	clark	JOHN	CLARK
3	aNGELA	bARRY	ANGELA	BARRY
4	sabrina	flanders	SABRINA	FLANDERS
5	edwarD	garwooD	EDWARD	GARWOOD



	A	B	C	D
1	First Name	Last Name	Lower First Name	Lower Last Name
2	JOHN	CLARK	john	clark
3	aNGELA	bARRY	angela	barry
4	sabrina	flanders	sabrina	flanders
5	edwarD	garwooD	edward	garwood



	A	B	C	D
1	First Name	Last Name	Proper First Name	Proper Last Name
2	JOHN	CLARK	John	Clark
3	aNGELA	bARRY	Angela	Barry
4	sabrina	flanders	Sabrina	Flanders
5	edwarD	garwooD	Edward	Garwood

Activity 12. Working with Text Functions