Microsoft Excel – Part 6

Creating advanced formulas

IF

IF Function: Returns one value if a condition you specify evaluates to TRUE and another value if it evaluates to FALSE. Use IF to conduct conditional tests on values and formulas.

Syntax

IF(logical_test,value_if_true,value_if_false)

1 2 3

Formula with the IF function

1 logical_test: The condition that you want to check. This argument can use any comparison calculation operator (=, >, <, =>, =<, <>).

2 value_if_true: The value to return if the condition is true. Value_if_true can be another formula.

3 value_if_false: The value to return if the condition is false. Value_if_false can be another formula.

Use *Advanced Excel Skills 2.xlsx* to learn this IF logic.
Open IFCondition worksheet to practice the followings:

- Select cell H2. Enter an IF condition formula to achieve the result – If the value in cell B2 is greater than 500, give me the excess amount over 500, or else just give me the text "OK".
=IF(B2>500,B2-500,"OK")
- You may copy the formula in H2 to all the way to cell H17 by using the Fill Handle.

Note: Up to 64 IF functions can be nested as value_if_true and value_if_false arguments to construct more elaborate tests.

If you have more than one criteria, you may nest IF functions as below.

Use *Nested Ifs* worksheet to figure out two IF conditions.

1. If Cell A2 has "UK", commission in Cell F2 will be 20% of Cell E2.
2. If Cell A2 has "USA", commission in Cell F2 will be 25% of Cell E2.

We can achieve these two IFS by nesting one IF inside the other. The single IF Function is below.

IF(logical_test,value_if_true,value_if_false)

In place of "value_if_false", we will nest another IF function.

=IF(A2="UK",E2*0.2,IF(A2="USA",E2*.25))

A2=”UK” is our first logical test

E2*0.2 is our value if true.

IF(A2=”USA”, E2*.25) is the another IF function to replace value if false.

See more examples of more than 2 nested functions in nestedconditional worksheet.

COUNTIF

Use *More* worksheet to learn this concept. The **COUNTIF** function counts the number of cells within a range that meet a single criterion that you specify. For example, you can find out how many times a particular text or a number value occurs.

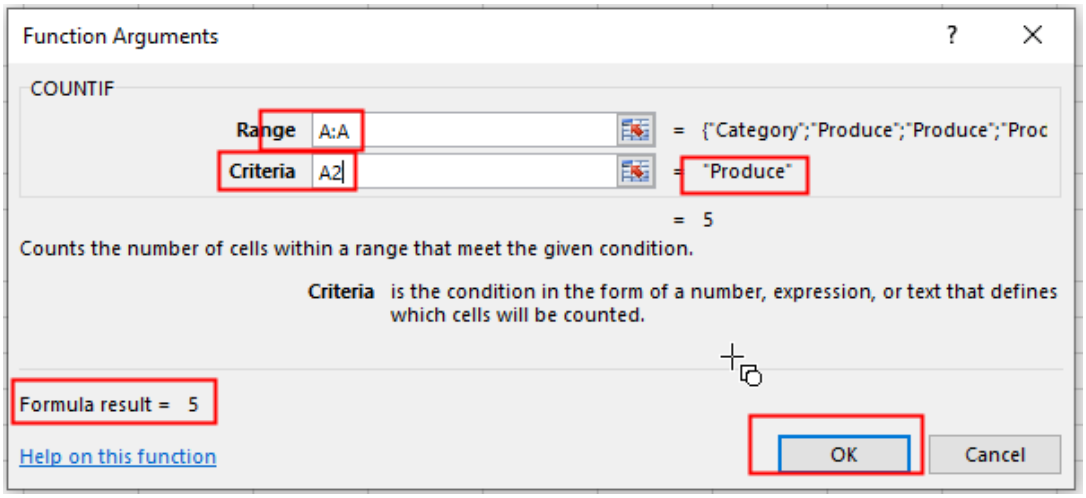
Syntax: =COUNTIF(range, criteria)

Range: Required. One or more cells to count, including numbers or names, arrays, or references that contain numbers. Blank and text values are ignored.

Criteria: Required. A number, expression, cell reference, or text string that defines which cells will be counted. For example, criteria can be expressed as 32, ">32", B4, "apples", or "32".

We want to count how many cells in Column A (range) includes the word “Produce” (criteria). (If we want more explanation and see the examples, we can click on **Help on this Function** box and a new **Help** window will pop up with more detailed explanation and examples.) Follow the steps below.

- Select **Cell H4**.
- Click on **Formulas>Insert Function.**
- Select **COUNTIF** from the suggested functions.
- Click on **OK** button.
- **Function Arguments** dialog box will come up. In the **Range** box, you want to count all cells in Column A. So, simply select the entire column A.
- In the **Criteria** box, you want to include all cells in column A with the word “Produce” in it. You can type in the word “Produce” yourself or simply select one of the cells that include “Produce” in it, in this example, point and click on cell **A4.** You can see the word “Produce” shows up on the right of the **Criteria** box and the result shows “5” already.
- Click on **OK** button.
- Click on **OK**. The result should be 5.



SUMIF

Use *More* worksheet in *IfCondition.xlsx* workbook to practice this concept.

You use the SUMIF function to sum the values in a range that meet criteria that you specify. For example, suppose that in a column that contains numbers, you want to sum only the values that are larger than 14. You can use the following formula: =SUMIF(F4:F30,">14") or =SUMIF(F:F,”>14”). Enter the formula in Cell H4 to find out the sum of Total cost that is over $14.00.

If you want, you can apply the criteria to one range and sum the corresponding values in a different range. For example, the formula =SUMIF(A:A,"DAIRY",F:F) Or SUMIF(A4:A30,”DAIRY”,F4:430) sums only the values in the range F:F or F4:F30, where the corresponding cells in the range A:A or A4:A30 equal “Dairy”. Try one of the formulas in cell H5.

IFERROR

IFERROR is a function in Excel that returns an alternate result, like a blank cell or text, when a formula evaluates an error. It's categorized as a "logical function" that can also work as a worksheet function. The syntax for IFERROR is:

=IFERROR(value,value_if_error)

The calculation formula fills the "value" part, and the desired return fills the "value_if_error" part. When using IFERROR(formula,"error display"), the function evaluates the parts of the formula. If the formula encounters no errors, IFERROR returns the expected result of the formula. If the formula encounters an error, IFERROR returns the chosen display value.

Errors that IFERROR identifies

IFERROR monitors these error returns in a spreadsheet:

- #DIV/0!: #DIV/0! appears when applying a division or percentage formula and entering a zero value in the cell range of the denominator argument.

- #N/A: An #N/A returns when a LOOKUP function or MATCH function of a formula can't find a reference value, like when a lookup value isn't in the source data.
- #VALUE!: #VALUE! results when the formula encounters an unexpected type, like a blank cell, text input instead of numbers or vice versa or dates treated as text.
- #REF!: #REF! develops when a formula refers to a cell that isn't valid, which may happen when you delete, paste over or move referenced cells in the spreadsheet.
- #NUM!: A #NUM! error happens when a formula or function contains numeric values that aren't valid for the formula, like when an entered numeric value uses a different data type or number format that the argument doesn't support.
- #NAME?: #NAME? appears when Excel doesn't recognize something in the formula, like a misspelling in a formula or function name or a mistyped formula argument.
- #NULL!: #NULL! populates when you use an incorrect range operator in a formula or use an intersection operator (space character) to specify two ranges that don't intersect.

Let's practice using *More* worksheet in Cell I4 enter the formula =F4/C4. You should get the correct result 3.99. Now, change the C4 value from 1 to 0 (zero). The result in I4 now shows #DIV/0! As you have entered a zero value in denominator argument. You may still prefer to use 0 value instead of leaving the cell blank and expect the desired result value as 0 (zero). Let's fix the formula in Cell I4 by selecting that cell again and in formula bar area, click in front of F4/C4 and fix to look like this =iferror(F4/C4,0). By doing so, your result in cell I4 now should see a zero value instead of DIV error.

VLOOKUP

Let's say you want to look up an employee's phone extension by using their badge number or the correct rate of a commission for a sales amount. You look up data to quickly and efficiently **find specific data in a list** and to automatically verify that you are using correct data. After you look up the data, you can perform calculations or display results with the values returned. There are several ways to look up values in a list of data and to display the results. There are two Lookup functions: **VLOOKUP** and **HLOOKUP.**

When to use what:

- VLOOKUP looks at a value in one column, and finds its corresponding value on the same row in another column. Use VLOOKUP when your comparison values are located in a column (*Vertical)* to the left of the data you want to find
- HLOOKUP searches for a value in the top row of a table or an array of values, and then returns a value in the same column from a row you specify in the table or array. Use HLOOKUP when your comparison values are located in a row across the top of a table of data, and you want to look down a specified number of rows (*Horizontal)*.

VLOOKUP

Syntax - VLOOKUP(lookup_value,table_array,col_index_num,range_lookup)

Sample - =VLOOKUP(1,A2:C10,2,True) OR =VLOOKUP(1,A2:C10,2,False)

Meaning: Look up value 1 in cell ranges A2:C10; if found give me the data in column #2 from the same row where 1 was found. *True* means give me exact or approximate match; *False*, the exact match.

Lookup_value (Required): The value to search in the first column of a table_array, meaning the value you want to look up must be in the first column of the range of cells you specify in table-array. It can be a value (either a number or text) or a reference cell such as A23. If the lookup_value is a text, place it in double quotes. If lookup_value is a number and smaller than the smallest value in the first column of table_array, VLOOKUP returns the #N/A error value.

Table_array (Required): One or more columns of data. Use a reference to a range or a range name. The values in the first column of table_array are the values searched by lookup_value. These values can be text, numbers, or logical values. Uppercase and lowercase texts are equivalent. You **cannot** have duplicate values in the leftmost column of the lookup range.

Col_index_num (Required): It is the column number (starting with 1 for the left-most column of table-array) that contains the return value. A col_index_num of 1 returns the value in the first column in table_array (in the same row); a col_index_num of 2 returns the value in the second column in table_array (in the same row), and so on. If col_index_num is:

Less than 1, VLOOKUP returns the #VALUE! error value.

Greater than the number of columns in table_array, VLOOKUP returns the #REF! error value.

Range_lookup (Optional): A logical value that specifies whether you want VLOOKUP to find an exact match or an approximate match:

If TRUE or omitted, an exact or approximate match is returned. If an exact match is not found, the next largest value that is less than lookup_value is returned. **The values in the first column of table_array must be placed in ascending sort order**; otherwise, VLOOKUP may not give the correct value. You can put the values in ascending order by choosing the Sort command from the Data menu and selecting Ascending.

If FALSE, VLOOKUP will only find an exact match. In this case, the values in the first column of table_array **do not need to be sorted**. If there are two or more values in the first column of table_array that match the lookup_value, the first value found is used. If an exact match is not found, the error value #N/A is returned.

Remarks

When searching text values in the first column of table_array, ensure that the data in the first column of table_array does not have leading spaces, trailing spaces, inconsistent use of straight (' or ") and curly (‘ or “) quotation marks, or nonprinting characters. In these cases, VLOOKUP may give an incorrect or unexpected value. (Use =TRIM(Cell address) function to get rid of leading or trailing spaces.)

When searching number or date values, ensure that the data in the first column of table_array is not stored as text values. In this case, VLOOKUP may give an incorrect or unexpected value.

If range_lookup is FALSE and lookup_value is text, then you can use the wildcard characters, question mark (?) and asterisk (*), in lookup_value. A question mark matches any single character; an asterisk matches any sequence of characters. If you want to find an actual question mark or asterisk, type a tilde (~) preceding the character.

Open the *1000 Records* worksheet in the same workbook.

- In Cell R2, enter the VLookup formula to find the social security number of someone with the employee ID number 330816. Note the employee numbers are in column A and social security in Column K or 11th column. (answer: =VLOOKUP(330816,A:K,11,FALSE)
- In Cell R3, enter the VLookUp formula to find the social security number of someone with the last name "freeman". (answer: =VLOOKUP("freeman", E:K,7,FALSE)

But VLOOKUP has many limitations. It can only look to the right from the first column. It only finds the first match in a data set. What if you only know the last name "freeman" and you want to find his employee ID number? Your data is not set accommodate your search to the right of your lookup value; in fact, the column index number that holds your answer is on the left of the Lookup value column. That is where INDEX and MATCH functions comes in handy.

INDEX MATCH

=INDEX finds a cell's value in a table using its column and row number

=MATCH finds the position of a cell in a row or column

Together, they can find and give you a cell's value in a table by looking up both up and down, and left and right. This is known as the Index and Match formula.[XELPLUS]

This is the quote from the XELPlUS site.

The syntax for INDEX is:

=INDEX(array, row_num, [column_num])

- array: The range where the return value resides. You're basically pointing to a block of cells and saying, "Hey, the information I want is somewhere in here."
- row_num: You're telling INDEX how many rows down from the top of your selected area to go to find the information. If you say "2", INDEX will move two rows down from the top of the area you pointed out.
- [column_num]: This is like the row number, but instead, you're telling INDEX how many columns to the right to go. If you say "3", it moves three columns to the right. This part is optional because sometimes you're only interested in a single column.

In short, you're giving INDEX directions like, "In this block of cells, go down this many rows and over this many columns, and show me what's there." That's how the INDEX formula knows what specific piece of information to grab and show you. On its own, the INDEX formula is not that useful. **We hardly know exactly which row or column location that our information exist in a spreadsheet.**

Here's how you set up the MATCH function:

=MATCH(lookup_value, lookup_array, [match_type])

- lookup_value (what you're searching for): This is the piece of information you want to find. You can either point to a cell that has this information or type it directly into the formula.
- lookup_array (where you're searching): Think of this as the list or table in your spreadsheet where MATCH is going to look for your item. It's the area where you expect to find what you're searching for.
- match_type (how precise you want to be): When you use "0" here, it means you want an exact match. That tells MATCH you're looking for something that matches your lookup value perfectly, without any differences.

So to practice with what we have, let's look for the employee id number for the last name "freeman". All we know is the last name. We want to find his employee ID (the column with employee ID info is left of the last name column). So we cannot use VLookUp in this case. We will use Index and Match functions together. Select Cell R5.

We will give the array (block of cells or even entire columns) that include the information somewhere in there. In this case, we know the employee ID is somewhere in column A and we will select entire column A.

=INDEX(A:K,

This will followed immediately by MATCH function like this.

=INDEX(A:K,MATCH(

Now, we have to provide the lookup value as it required by MATCH function

=MATCH(lookup_value, lookup_array, [match_type])

=INDEX(A:K,MATCH("freeman",

What we are looking for is text so we place that in double quotes. This has to be followed by the lookup_array, which in this case is Column E where employee last name exists. Then the match type is 0 to have an exact match. That completes the Match function.

=INDEX(A:K,MATCH("freeman",E:E,0)

Now, back to finish the INDEX function. The requirement is

=INDEX(array, row_num, [column_num])

We replaced "row_num" entirely with MATCH("freeman",E:E,0). Now we have to come up with the [Column_num].

As what we wanted in column A which is column number 1, we will write

=INDEX(A:K,MATCH("freeman",E:E,0),1) as we are pointing out the information we want (employee ID) is in the column 1.

The answer should be “330816”, the employee number in Column A that match the same row in Column E, where the employee last name is our lookup value “freeman”.

Try to change column number to 7 to find freeman’s email; or column number to 10 to find his salary, to practice.

Power Query

Follow Kevin Stratvert on his Power Query episode on YouTube here at https://www.youtube.com/watch?v=0aeZX1l4JT4

The data used here is taken from his sample data.

Power Query is a data transformation and preparation tool. It comes with a graphical interface for getting data from various sources and a Power Query Editor will let you transform the data before uploading the data onto your Excel sheet. You do not have to know any coding. Power Query Editing tool will record the steps you transform your data into steps like macro and you can see which steps you took to get the clean data. You can make all kinds of adjustments to your data such as:

1. combining more than one file
2. Splitting columns
3. Adding columns with formulas
4. Changing formats
5. Analyzing with Pivot table and charts -- to name a few.

Let’s start a new Excel spreadsheet.

1. Click on Data tap
2. Under New Query, select From File then From folder (this is to pull more than one Excel file from a folder)
3. Click on Browse and direct to the appropriate folder where your files are; you will see all files (I have 3 in the list) in the new window
4. Here, you want to combine and edit the files. It allows you to transform or edit the data before it pulls into Excel sheet. Click the arrow next to Combine and select “Combine and Edit”.
5. In next window, you will see the example file is the default first file of 3 files you are combining. Keep it as default. (you can select any other file if you like; this is to display as your sample as you edit your data before loading into Excel sheet)
6. Under File Parameter, you will see “Order Data” as all of 3 files have the same sheet named “Order Data”.
7. Select that and you will see the sample data on the right pane. Click on OK. Now you will get the Power Query Editor window. Here you can change all adjustments as you desire before loading into Excel sheet.
8. Let’s transform this data by taking the following steps. As you take each step, pay attention to the Query Settings pane on the right under Applied Steps.
    a. Let’s remove the Source_Name column by clicking on the Remove Columns on the ribbon or right click on the header row and select Remove.

b. Let's hide the undesired data marked as "xxxxxx" under Order ID column by filtering and unchecking that (bottom of the list). We do not know how many of them will be marked with that data but now that particular data will not come along into Excel sheet.
c. Let's split the Customer ID column into two since ID and customer names are in a single column. Select the entire Customer ID column. Click on Transform>Split Column>By Delimiter as we have hyphen in the middle of two different data. We can see a space before and space after the hyphen. So simply place a space before and after in the box shown. Click on OK. You will see the Customer ID.1 for customer IDs and Customer ID.2 for customer names. Simply rename them appropriately such as Customer ID and Customer Name by double-clicking on the header sections.
d. Let's add a new column that would show the profit (Revenue – Cost). Click on Revenue column first since it is the correct order in the formula. Hold down Control button and click on the Cost column. Click on Add Column>Under Standard, select Subtract. A new column will be added at the end of all columns. Simply rename as Profit and drag (by clicking in the header name) and move next to the Cost column.
e. Also want to apply currency format to all three columns of Revenue, Cost, and Profits. Select all there columns by using Control button. Go to Transform>Data Type>select Currency. Now you will see $ on header rows to make the correct data type.
f. Finally, let's add a column to find out the length from order to ship date. Here we want to have Ship Date – Order Date to figure out days in between. Make sure you click on Ship date column first then hold down Control and select Order date column. Go to Add Column>Date>Subtract Days. A new column with the name Subtraction will show up at the end. Rename as Days and move it next to Ship Date column.
g. Let's practice removing the Order Status column (right-click on the header column); you will see that step as Removed Columns1 in applied steps. If you want to reverse what you have just done, simply click on the X button on that step and the column will be back in the editor window.
h. Now, it seems all data is transformed and ready to be loaded. Click on Home tab>Close & Load.
i. Let's say now you have a new data file just received in your folder where you have previous 3 files. Point your mouse on the Workbook Queries pane on the right where it says Power Query. You will see where the data is coming from. By adding a new file to the same folder will allow you to refresh data under Query tab under Query Tools. You will see the data lines increase from 2,483 to 3,185 rows in this example.