Logical Functions

**How to Use the IF Function**

The IF function is the main logical function in Excel and is, therefore, the one to understand first. It will appear numerous times throughout this article.

Let's have a look at the structure of the IF function, and then see some examples of its use.

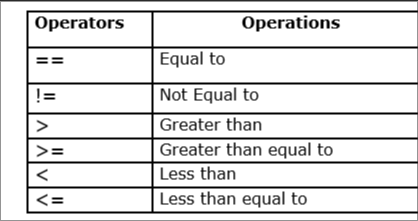
The IF function accepts 3 bits of information:

=IF(logical\_test, [value\_if\_true], [value\_if\_false])

* **logical\_test:** This is the condition for the function to check.
* **value\_if\_true:** The action to perform if the condition is met, or is true.
* **value\_if\_false:** The action to perform if the condition is not met, or is false.

**Comparison Operators to Use with Logical Functions**

When performing the logical test with cell values, you need to be familiar with the comparison operators. You can see a breakdown of these in the table below.



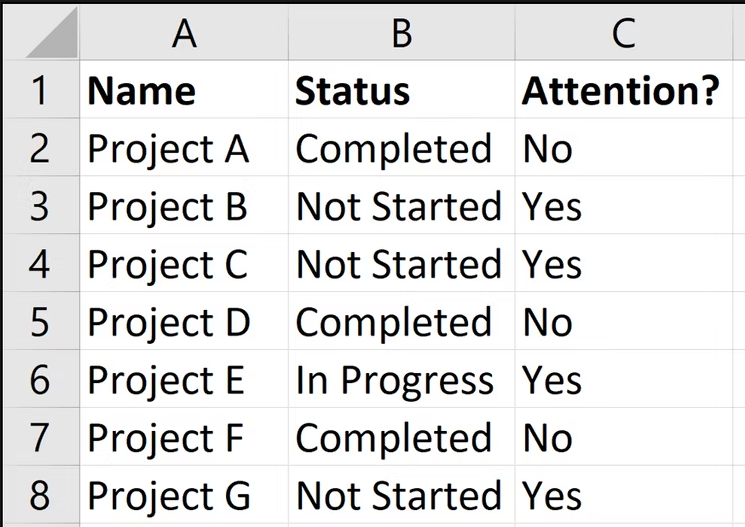
### IF Function Example 1: Text Values

In this example, we want to test if a cell is equal to a specific phrase. The IF function is not case-sensitive so does not take upper and lower case letters into account.

The following formula is used in column C to display "No" if column B contains the text "Completed" and "Yes" if it contains anything else.

=IF(B2="Completed","No","Yes")

Although the IF function is not case sensitive, the text must be an exact match.

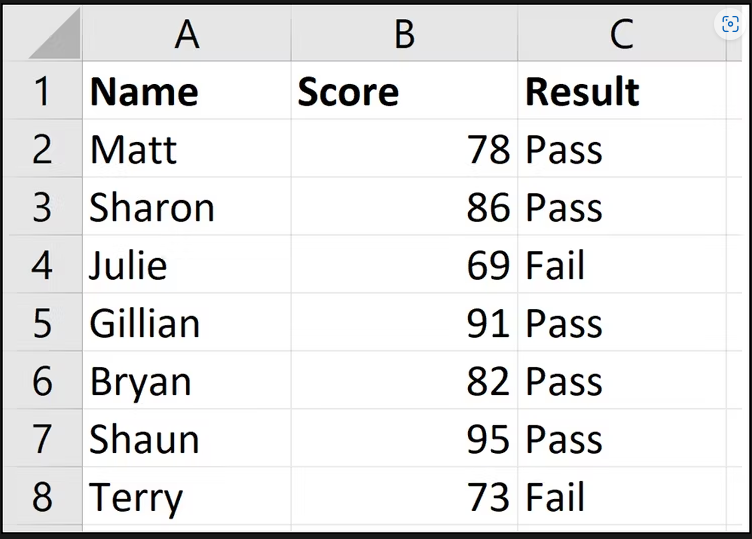


### IF Function Example 2: Numeric Values

The IF function is also great for comparing numeric values.

In the formula below we test if cell B2 contains a number greater than or equal to 75. If it does, then we display the word "Pass," and if not the word "Fail."

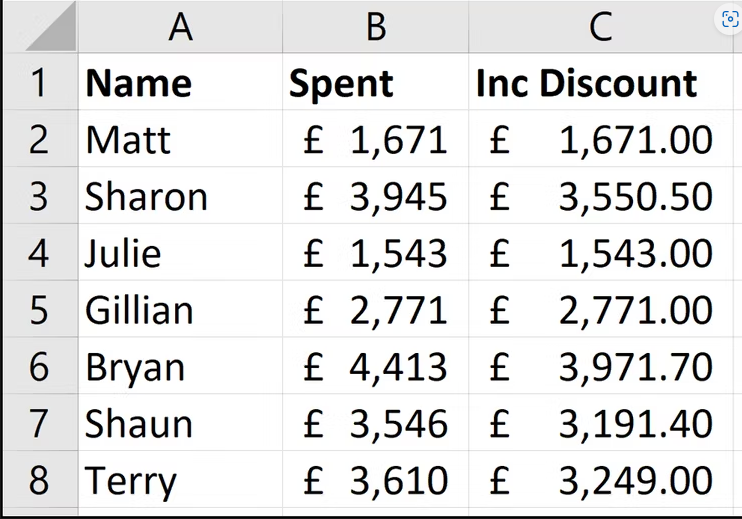
=IF(B2>=75,"Pass","Fail")



The IF function is a lot more than just displaying different text on the result of a test. We can also use it to run different calculations.

In this example, we want to give a 10% discount if the customer spends a certain amount of money. We will use £3,000 as an example.

=IF(B2>=3000,B2\*90%,B2)

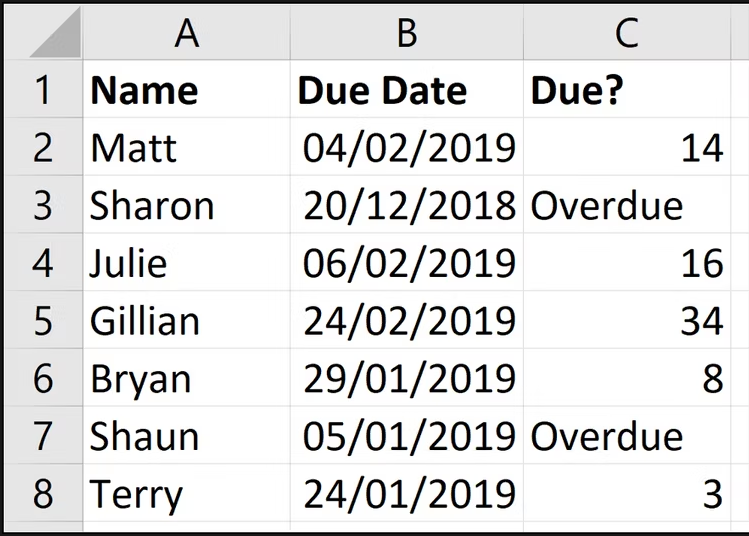


### IF Function Example 3: Date Values

In this third example, we use the IF function to track a list of due dates. We want to display the word "Overdue" if the date in column B is in the past. But if the date is in the future, calculate the number of days until the due date.

The formula below is used in column C. We check if the due date in cell B2 is less than today's date (The TODAY function returns today's date from the computer's clock).

=IF(B2<TODAY(),"Overdue",B2-TODAY())



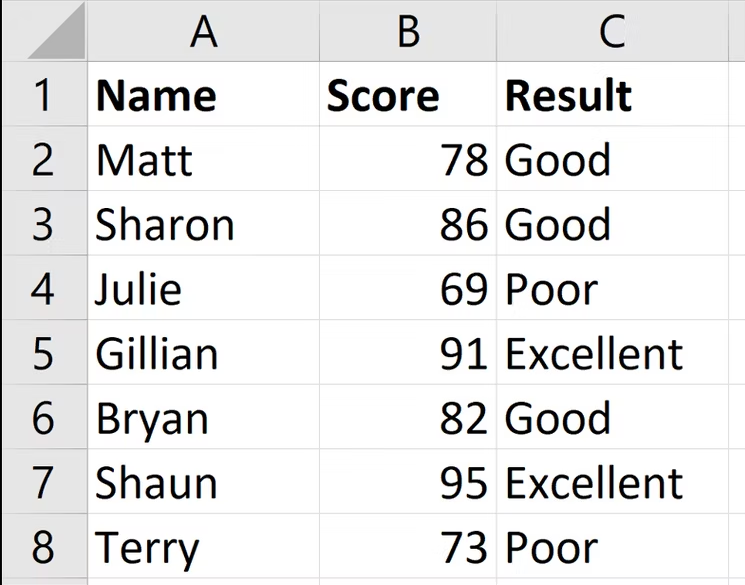
### What are Nested IF Formulas?

You may have heard of the term nested IFs before. This means that we can write an IF function within another IF function. We may want to do this if we have more than two actions to perform.

One IF function is capable of performing two actions (the value\_if\_true and value\_if\_false ). But if we embed (or nest) another IF function in the value\_if\_false section, then we can perform another action.

Take this example where we want to display the word "Excellent" if the value in cell B2 is greater than or equal to 90, display "Good" if the value is greater than or equal to 75, and display "Poor" if anything else.

=IF(B2>=90,"Excellent",IF(B2>=75,"Good","Poor"))



We have now extended our formula to beyond what just one IF function can do. And you can nest more IF functions if necessary.

Notice the two closing brackets on the end of the formula---one for each IF function.

There are alternative formulas that can be cleaner than this nested IF approach. One very useful alternative is [the SWITCH function in Excel](https://support.office.com/en-us/article/switch-function-47ab33c0-28ce-4530-8a45-d532ec4aa25e).

## The AND and OR Logical Functions

The AND and OR functions are used when you want to perform more than one comparison in your formula. The IF function alone can only handle one condition, or comparison.

Take an example where we discount a value by 10% dependent upon the amount a customer spends and how many years they have been a customer.

On their own, the AND and OR functions will return the value of TRUE or FALSE.

The AND function returns TRUE only if every condition is met, and otherwise returns FALSE. The OR function returns TRUE if one or all of the conditions are met, and returns FALSE only if no conditions are met.

These functions can test up to 255 conditions, so are certainly not limited to just two conditions like is demonstrated here.

Below is the structure of the AND and OR functions. They are written the same. Just substitute the name AND for OR. It is just their logic which is different.

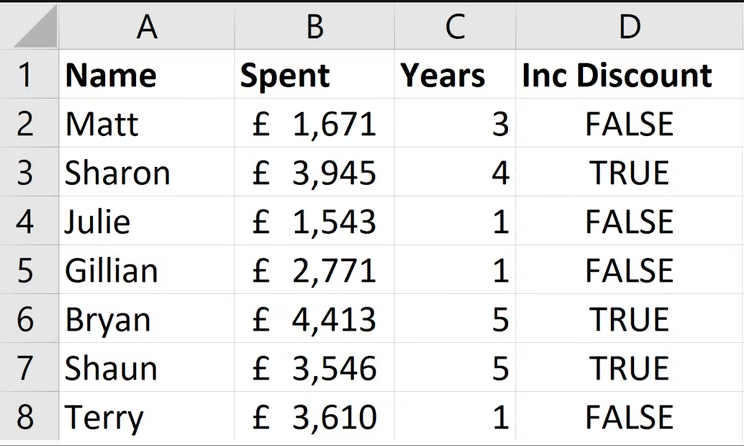
=AND(logical1, [logical2] ...)

Let's see an example of both of them evaluating two conditions.

### AND Function example

The AND function is used below to test if the customer spends at least £3,000 and has been a customer for at least three years.

=AND(B2>=3000,C2>=3)

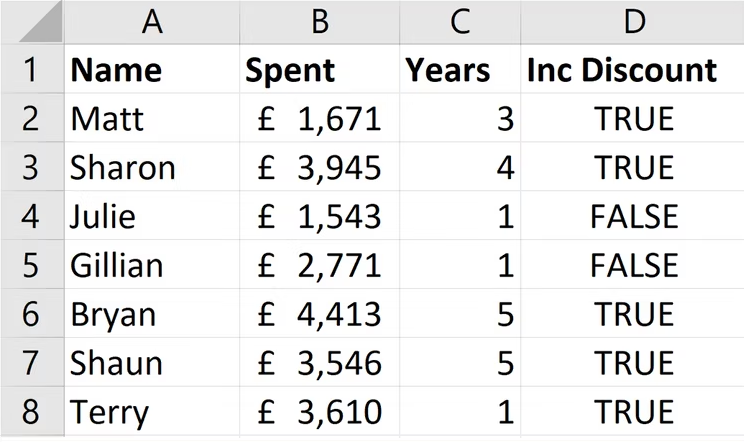


You can see that it returns FALSE for Matt and Terry because although they both meet one of the criteria, they need to meet both with the AND function.

### OR Function Example

The OR function is used below to test if the customer spends at least £3,000 or has been a customer for at least three years.

=OR(B2>=3000,C2>=3)



In this example, the formula returns TRUE for Matt and Terry. Only Julie and Gillian fail both conditions and return the value of FALSE.

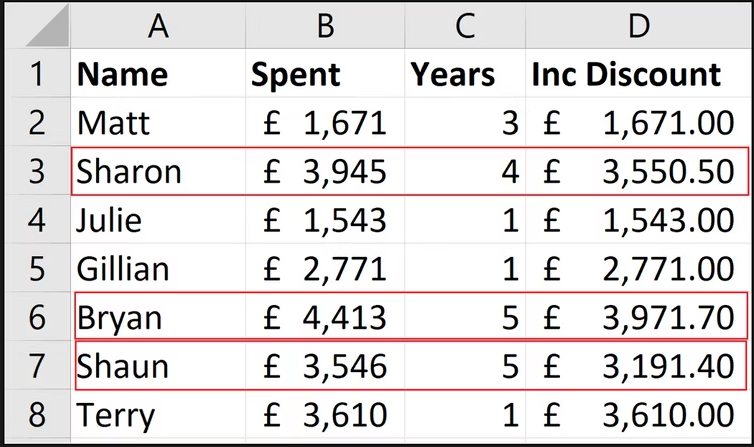
### Using AND and OR with the IF Function

Because the AND and OR functions return the value of TRUE or FALSE when used alone, it's rare to use them by themselves.

Instead, you'll typically use them with the IF function, or within an Excel feature such as Conditional Formatting or Data Validation to perform some retrospective action if the formula evaluates to TRUE.

In the formula below, the AND function is nested inside the IF function's logical test. If the AND function returns TRUE then 10% is discounted from the amount in column B; otherwise, no discount is given and the value in column B is repeated in column D.

=IF(AND(B2>=3000,C2>=3),B2\*90%,B2)



IFNA function in Excel

The Excel IFNA function is purposed for catching and handling #N/A errors. If a formula evaluates to #N/A, IFNA traps that error and replaces it with a custom value you specify; otherwise returns a normal result of the formula.

IFNA syntax

The syntax of the IFNA function is as follows:

**IFNA(value, value\_if\_na)**

Where:

**Value** (required) - the formula, value, or reference to check for a #N/A error.

**Value\_if\_na** (required) - the value to return if a #N/A error is detected.

How to use IFNA function in Excel

To effectively use IFNA in Excel, follow this generic approach:

1. In the first argument (*value*), put a formula impacted by the #N/A error.
2. In the second argument (*value\_if\_na*), type the text you want to return instead of the standard error notation. To return an empty cell when nothing is found, supply an empty string ('"").

To return **custom text**, the generic formula is:

**IFNA(*formula*(), "*custom text*")**

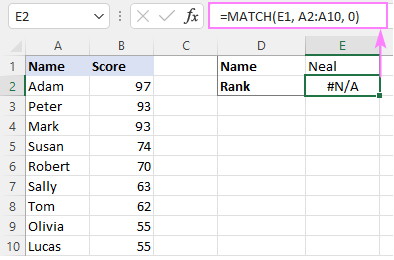
To return a **blank cell**, the generic formula is:

**IFNA(*formula*(), "")**

Let's see how it works on a simple example. In the table below, suppose you want to know how a score of a given student ranks among others. Since the data is sorted by the *Score* column from highest to lowest, the rank will match the relative position of the student in the table. And to get the position, you can use the [MATCH](https://www.ablebits.com/office-addins-blog/excel-match-function/) function in its simplest form:

**=MATCH(E1, A2:A10, 0)**

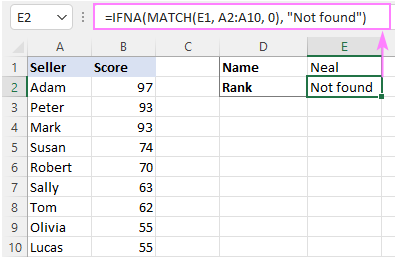
Because the lookup value (Neal) is not available in the lookup array (A2:A10), a #N/A error occurs.



Running into this error, inexperienced users might think there's something wrong with the formula, and you as the workbook creator will receive a lot of questions. To avoid this, you can explicitly indicate that the formula is correct, it just cannot find the value it is asked to look for. So, you nest the MATCH formula in the first argument of IFNA and, in the second argument, type your custom text, "Not found" in our case:

**=IFNA(MATCH(E1, A2:A10, 0), "Not found")**

Now, instead of the standard error notation, your own text is displayed in a cell, informing users that the lookup value is not present in the dataset:



## What's the difference between IFNA and IFERROR?

Depending on the root cause of the problem, an Excel formula can trigger different errors such as #N/A, #NAME, #VALUE, #REF, #DIV/0, #NUM, and others.

The [IFERROR](https://www.ablebits.com/office-addins-blog/excel-iferror-function-formula-examples/) function catches all those errors while IFNA is limited to only #N/A. Which one is better to choose? That depends on the situation.

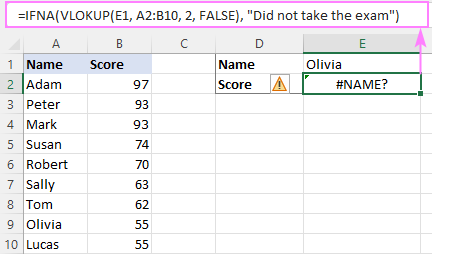
If you want to suppress **any kind of error**, then use the IFERROR function. It is especially useful in complex calculations when a formula includes several functions that can generate different errors.

With **lookup functions**, you'd better use IFNA as it displays a custom result only when a lookup value is not found and does not hide underlying problems with the formula itself.

To illustrate the difference, let's bring back our basic [IFNA VLOOKUP formula](https://www.ablebits.com/office-addins-blog/ifna-function-excel/#example-basic) and "accidentally" misspell the function's name (VLOKUP instead of VLOOKUP).

**=IFNA(VLOKUP(E1, A2:B10, 2, FALSE), "Did not take the exam")**

IFNA does not suppress this error, so you can clearly see that something is wrong with one of the function names:



Now, let's see what happens if you use IFERROR:

**=IFERROR(VLOKUP(E1, A2:B10, 2, FALSE), "Did not take the exam")**

Hmm… it says that Olivia did not take the exam, which is not true! This is because the IFERROR function traps the #NAME? error and returns the custom text instead. In this situation, it not only returns wrong information but also obscures the issue with the formula.

