

---

# Ms Excel All Formulas With Examples

---

*Ms Excel All  
Formulas  
With  
Examples*

*Downloaded from  
[template-dev.mobaptist.org](http://template-dev.mobaptist.org)  
by Rivera & Ballard*

---

## INTRODUCTION: UNLOCKING THE POWER OF EXCEL FORMULAS

---

Microsoft Excel is more than just a spreadsheet application; it is a powerful tool for data analysis, financial modeling, and business intelligence. At the heart of Excel's functionality lie its formulas—expressions that perform calculations, manipulate text, and derive insights from

raw data. Whether you are a beginner looking to automate simple additions or an advanced user building complex dashboards, mastering **Ms Excel all formulas with examples** is essential for efficiency and accuracy.

Formulas in Excel always begin with an equals sign (=) and can combine cell references, operators, functions, and constants. The real magic happens when you combine multiple functions to solve real-world problems. This

article provides a detailed guide to the most important Excel formulas, each accompanied by practical examples and clear explanations. By the end, you will have a solid foundation to handle data tasks confidently.

---

## CORE MATHEMATICAL AND STATISTICAL FORMULAS

---

# The SUM Formula

One of the most frequently used formulas in Excel is **SUM**. It adds all numbers in a selected range of cells.

### Syntax:

`=SUM(number1, [number2], ...)`

**Example:** You have

monthly sales in cells A2 to A13. To calculate total sales, enter `=SUM(A2:A13)`. Excel returns the sum of all values in that range. If you have non-adjacent cells (e.g., A2, A5, A9), use `=SUM(A2, A5, A9)`.

# The AVERAGE Formula

To find the mean of a dataset, use **AVERAGE**.

### Syntax:

`=AVERAGE(number1, [number2], ...)`

**Example:** To calculate the average exam score from cells B2 to B30, write `=AVERAGE(B2:B30)`. This formula ignores empty cells and text, making it ideal for

numeric datasets.

## The COUNT and COUNTA Formulas

**COUNT** counts cells containing numbers, while **COUNTA** counts all non-empty cells, including text.

### Syntax:

**=COUNT**(value1, [value2], ...)

### Syntax:

**=COUNTA**(value1, [value2], ...)

**Example:** In a column of student grades (some blank, some text like "Absent"), **=COUNT**(C2:C50) returns the number of numeric entries.

**=COUNTA**(C2:C50) returns all filled cells, including "Absent".

## The MIN and MAX Formulas

To find the smallest or largest value in a range, use **MIN** and **MAX**.

### Syntax:

**=MIN**(number1, [number2], ...)  
**=MAX**(number1, [number2], ...)

**Example:** For daily temperatures in D2:D365, **=MIN**(D2:D365) gives the lowest temperature, and **=MAX**(D2:D365) gives the highest.

---

## LOGICAL FORMULAS FOR

## DECISION MAKING

# The IF Function

The **IF** formula performs a logical test and returns one value if true, another if false. It is the cornerstone of conditional logic in Excel.

### **Syntax:**

```
=IF(logical_test,  
value_if_true,  
value_if_false)
```

**Example:** Suppose you have sales amounts in column E. To label them as "Target Met" if  $\geq 10000$  and "Below Target" if not, use `=IF(E2>=10000, "Target Met", "Below Target")`. You can nest IF functions for multiple conditions, but newer alternatives like IFS or

SWITCH are often clearer.

# The AND, OR, and NOT Functions

These logical functions combine multiple conditions.

### **Syntax:**

```
=AND(condition1,  
[condition2],  
...)  
=OR(condition1,  
[condition2],  
...)  
=NOT(condition)
```

**Example:** To check if a student passed both Math ( $\geq 50$ ) and English ( $\geq 60$ ), use `=IF(AND(F2>=50, G2>=60), "Pass", "Fail")`. OR returns true if at least one

condition is met.

## IFERROR and IFNA

These handle errors gracefully. **IFERROR** catches any error, while **IFNA** specifically catches #N/A errors.

### Syntax:

```
=IFERROR(value,  
value_if_error)  
=IFNA(value,  
value_if_na)
```

**Example:** If you divide A1 by B1 and B1 might be empty, `=IFERROR(A1/B1, "No value")` prevents ugly error messages.

---

## LOOKUP AND REFERENCE FORMULAS

---

## VLOOKUP (Vertical Lookup)

**VLOOKUP** searches for a value in the first column of a table and returns a value in the same row from a specified column. It is essential for pulling data from large datasets.

### Syntax:

```
=VLOOKUP(lookup_v  
alue,  
table_array,  
col_index_num,  
[range_lookup])
```

**Example:** You have a product list in A2:C100 with Product ID, Name, and Price. To find the price of product with ID "P123", use `=VLOOKUP("P123", A2:C100, 3, FALSE)`. The FALSE

ensures an exact match.

# INDEX and MATCH (Advanced Lookup)

Combining **INDEX** and **MATCH** offers more flexibility than VLOOKUP, especially when your lookup column is not the leftmost one.

## Syntax:

```
=INDEX(array,
row_num,
[column_num])
=MATCH(lookup_value, lookup_array,
[match_type])
```

**Example:** To find the price of a product from column C where the

Product Name is in column B and the Price is in column D, use =INDEX(D2:D100, MATCH("Widget", B2:B100, 0)). This is more robust and faster for large datasets.

# XLOOKUP (Modern Replacement)

Available in newer Excel versions, **XLOOKUP** simplifies lookups and eliminates many limitations of VLOOKUP and INDEX-MATCH.

## Syntax:

```
=XLOOKUP(lookup_value,
lookup_array,
return_array,
[if_not_found],
[match_mode],
```

[search\_mode])

### Example:

=XLOOKUP("Widget",  
B2:B100,  
D2:D100, "Not  
found") returns the  
price directly. No need  
to specify column  
index number.

## TEXT AND STRING FORMULAS

# LEFT, RIGHT, and MID

These extract parts of  
a text string.

### Syntax:

=LEFT(text,  
[num\_chars])  
=RIGHT(text,  
[num\_chars])  
=MID(text,  
start\_num,  
num\_chars)

**Example:** If A1  
contains "Excel-2024",

=LEFT(A1,5) returns  
"Excel",  
=RIGHT(A1,4)  
returns "2024", and  
=MID(A1,7,4) also  
returns "2024".

# CONCATENATE and TEXTJOIN

Join text from multiple  
cells. **TEXTJOIN** is  
newer and offers a  
delimiter.

### Syntax:

=CONCATENATE(text  
1, [text2], ...)  
or simply =text1 & "  
& text2  
=TEXTJOIN(delimit  
er, ignore\_empty,  
text1, [text2],  
...)

**Example:** First name  
in B2, last name in C2:  
=B2 & " " & C2  
gives "John Doe". For a  
list: =TEXTJOIN(", "

", TRUE, A1:A10) combines all non-empty cells with a comma and space.

## LEN, TRIM, and UPPER/LO WER

**LEN** returns the character count, **TRIM** removes extra spaces, and **UPPER/LOWER** change case.

**Example:** =LEN(A1) counts characters. =TRIM(A1) cleans up messy data. =UPPER(A1) converts to uppercase.

---

### DATE AND TIME FORMULAS

---

## TODAY and NOW

**TODAY** returns the current date, **NOW** returns date and time. They update automatically.

**Syntax:** =TODAY()  
=NOW()

**Example:** To calculate age from birthdate in B2: =DATEDIF(B2, TODAY(), "Y").

## DATEDIF and EDATE

**DATEDIF** calculates the difference between two dates in days, months, or years.

**EDATE** adds months to a date.

**Example:**

**FORMULAS** A1, A2, "m") gives full months between two dates. =EDATE(A1, 6) returns the date six months later.

**WEEKDAY**  
and  
**NETWORKDAYS**  
**KDAYS**

**WEEKDAY** returns the day of the week (1=Sunday, etc.).  
**NETWORKDAYS** counts working days between two dates excluding weekends.

**Example:**  
=NETWORKDAYS(A1, A2) excludes Saturday and Sunday by default.

**ROUND,**  
**ROUNDU**  
**P,**  
**ROUND**  
**DD**  
**OWN**

These round numbers to a specified number of digits.

**Syntax:**  
=ROUND(number, num\_digits)

**Example:**  
=ROUND(3.14159, 2) returns 3.14. For currency, use =ROUND(A1,2) to avoid rounding errors in calculations.

**MATH AND**  
**TRIGONOMETRY**

# POWER and Sqrt

FINANCIAL

**POWER** raises a number to a power; **SQRT** finds square root.

**Example:** =POWER(5, 3) returns 125.  
=SQRT(144) returns 12.

# MOD (Modulo)

Returns the remainder after division. Useful for checking even/odd or cyclic patterns.

**Example:** =MOD(10, 3) returns 1. To check if a number in A1 is even:  
=IF(MOD(A1, 2)=0,

"Even", "Odd").

## STATISTICAL AND

## FORMULAS

# SUMIF and SUMIFS

**SUMIF** adds values based on one condition; **SUMIFS** handles multiple criteria.

### Syntax:

=SUMIF(range, criteria, [sum\_range])  
=SUMIFS(sum\_range, criteria\_range1, criterial, ...)

### Example: