50 Must-Know Excel Functions for Data Analysts

This list contains 50 Excel functions that are crucial for data analysts, spanning categories like Text, Logical, Date, Lookup, Statistical, and more.

Simple definitions and practical examples for each function are included. You can use this as a checklist of functions to learn and/or a quick reference guide.

Text Functions

1. **CONCAT** - Joins two or more text strings into one string.
   Example: `CONCAT("Hello ", "World!")` ➜ "Hello World!"
   Alternatives: **TEXTJOIN**, **CONCATENATE**
2. **MID** - Extracts a substring from the middle of a text string.
   Example: `MID("Excel", 2, 3)` ➜ "xce"
   Alternatives: **LEFT**, **RIGHT**
3. **LOWER** - Converts all characters in a text string to lowercase.
   Example: `LOWER("Excel")` ➜ "excel"
   Alternatives: **UPPER**
4. **PROPER** - Capitalizes the first letter of each word in a text string.
   Example: `PROPER("excel function")` ➜ "Excel Function"
5. **TRIM** - Removes extra spaces from text except for single spaces between words.
   Example: `TRIM(" Excel ")` ➜ "Excel"
   Alternatives: **CLEAN**
6. **VALUE** - Converts a text string that represents a number to a number.
   Example: `VALUE("123")` ➜ 123
7. **TEXT** - Formats a number and converts it to text.
   Example: `TEXT(1234, "$#,##0")` ➜ "$1,234"
8. **SEARCH** - Finds one text string within another, not case-sensitive, and returns the position of the found text.
   Example: `SEARCH("night", "Goodnight")` ➜ 5
   Alternatives: **FIND**
9. **TEXTSPLIT** - Splits text into rows or columns according to delimiters you specify.
   Example: `TEXTSPLIT("one, two, three", ",", TRUE, TRUE)` ➜ {one; two; three}
10. **TEXTBEFORE** - Extracts the part of a string before a specified delimiter.
    Example: `TEXTBEFORE("123-456-789", "-")` ➜ "123"
    Alternatives: **TEXTAFTER**
11. **SUBSTITUTE** - Replaces existing text with new text in a string.
    Example: `SUBSTITUTE("Hello World", "World", "Excel")` ➜ "Hello Excel"

Logical Functions

12. **IF** - Checks whether a condition is met, and returns one value if TRUE, and another if FALSE.

    Example: `IF(1+1=2, "Yes", "No")` ➜ "Yes"

13. **AND** - Returns TRUE if all of the arguments are TRUE.

    Example: `AND(TRUE, FALSE)` ➜ FALSE

14. **OR** - Returns TRUE if any argument is TRUE.

    Example: `OR(TRUE, FALSE)` ➜ TRUE

15. **NOT** - Reverses the logic of its argument.

    Example: `NOT(TRUE)` ➜ FALSE

16. **IFERROR** - Returns a value you specify if a formula evaluates to an error; otherwise, returns the result of the formula.

    Example: `IFERROR(1/0, "Error")` ➜ "Error"

17. **IFS** - Checks whether one or more conditions are met and returns a value that corresponds to the first TRUE condition.

    Example: `IFS(1>2, "No", 2>1, "Yes")` ➜ "Yes"

Date and Time Functions

18. **TODAY** - Returns the current date.
    Example: `TODAY()` ➜ [Current date]
    Alternatives: **NOW**

19. **DATE** - Returns the serial number of a particular date.
    Example: `DATE(2025, 1, 31)` ➜ 1/31/2025

20. **DATEDIF** - Calculates the difference between two dates.
    Example: `DATEDIF("1/1/2025","12/31/2025","d")` ➜ 364

21. **YEAR** - Converts a serial number to a year.
    Example: `YEAR("1/1/2025")` ➜ 2025
    Alternatives: **MONTH**, **DAY**

22. **EOMONTH** - Returns the serial number for the last day of the month that is the indicated number of months before (negative) or after (positive) the start date. Example: `EOMONTH("1/1/2025", 1)` ➜ 2/28/2025

23. **WEEKNUM** - Converts a date into a week number (1-52) that represents its position in the year.
    Example: `WEEKNUM("2025-03-17")` ➜ 12

24. **NETWORKDAYS** - Calculates the number of whole working days between two dates, excluding weekends and any identified holidays.
    Example: `NETWORKDAYS("2025-01-01", "2025-01-31")` ➜ 23

25. **YEARFRAC** - Returns the year fraction representing the number of whole days between two dates.
    Example: `YEARFRAC("2025-01-01", "2026-01-01")` ➜ 1.0

Financial Functions

26. **NPV** - Calculates the net present value of an investment based on a series of periodic cash flows and a discount rate.
    Example: `NPV(0.1, -100, 120)` ➜ 90.91
27. **IRR** - Calculates the internal rate of return for a series of cash flows.
    Example: `IRR({-100, 30, 35, 40})` ➜ 12%
28. **PMT** - Calculates the payment for a loan based on constant payments and a constant interest rate.
    Example: `PMT(0.05/12, 12, 1000)` ➜ -$85.61
29. **FV** - Calculates the future value of an investment.
    Example: `FV(0.05/12, 12, -100)` ➜ $1,230.16
    Alternatives: **PV**

Statistical & Engineering Functions

30. **AVERAGE** - Returns the average of the arguments.
    Example: `AVERAGE(1, 2, 3, 4)` ➜ 2.5
    Alternatives: **MEDIAN**, **MODE**
31. **COUNT** - Counts the number of cells that contain numbers, and counts numbers within the list of arguments.
    Example: `COUNT(1, 2, "text", TRUE, 5)` ➜ 3
32. **COUNTA** - Counts the number of cells that are not empty in a range.
    Example: `COUNTA(1, 2, "text", TRUE, 5)` ➜ 5
    Alternatives: **COUNTBLANK**
33. **STDEV.P / STDEV.S** - Calculates standard deviation based on the entire population or a sample.
    Example: `STDEV.S(1, 2, 3, 4, 5)` ➜ 1.58
    Alternatives: **VAR.P**, **VAR.S**
34. **PERCENTILE.INC** - Calculates the specified percentile of a dataset, inclusive of both endpoints.
    Example: `PERCENTILE.INC(A1:A10, 0.9)` ➜ [90th percentile value of A1:A10]
    Alternatives: **PERCENTILE.EXC**
35. **CONVERT** - Converts a number from one measurement system to another.
    Example: `CONVERT(100, "cm", "m")` ➜ 1

Lookup & Reference Functions

36. **XLOOKUP** - Searches a range or an array, and returns an item corresponding to the first match it finds.
    Example: `XLOOKUP("apple", A1:A10, B1:B10)` ➜ [Matching value from B1:B10]
    Alternatives: **VLOOKUP**, **HLOOKUP**

37. **INDEX** - Returns a value or the reference to a value from within a table or range.
    Example: `INDEX(A1:C3, 2, 3)` ➜ [Value in B2]
    Alternatives: **CHOOSECOLS**, **CHOOSEROWS**

38. **MATCH** - Searches for a specified item in a range of cells, and then returns the relative position (row or column number) of that item in the range.
    Example: `MATCH("apple", A1:A10, 0)` ➜ [Position of "apple" in A1:A10]
    Alternatives: **XMATCH**

39. **FILTER** - Returns an array that meets specified criteria from a given range or array, allowing for dynamic extraction of matching records.
    Example: `FILTER(A1:B10, B1:B10>10)` ➜ [Subset of rows from A1:B10 where the corresponding values in B1:B10 are greater than 10]

40. **UNIQUE** - Returns a list of unique values from a specified range or array.
    Example: `UNIQUE(A1:A10)` ➜ [List of unique values from A1:A10]

41. **SORT** - Sorts the contents of a range or array in ascending or descending order.
    Example: `SORT(A1:A10)` ➜ [Sorted values from A1:A10 in ascending order]

42. **TRANSPOSE** - Switches the rows and columns of an array or range.
    Example: `TRANSPOSE(A1:C3)` ➜ [Contents of A1:C3 with rows and columns switched]

Math Functions

43. **SUM** - Adds its arguments.
    Example: `SUM(1, 2, 3, 4)` ➜ 10

44. **SUBTOTAL** - Returns a subtotal in a list or database. It is primarily used for filtering and provides options to ignore hidden rows in the calculation.
    Example: Formula: `SUBTOTAL(9, A1:A10)` ➜ [Sum of A1:A10 excluding hidden rows]
    Alternatives: **AGGREGATE**

45. **ROUND** - Rounds a number to a specified number of digits.
    Example: `ROUND(2.555, 2)` ➜ 2.56
    Alternatives: **ROUNDUP**, **ROUNDDOWN**

46. **RANDBETWEEN** - Returns a random number between the numbers you specify.
    Example: `RANDBETWEEN(1, 10)` ➜ [Random number between 1 and 10] 48.
    Alternatives: **RAND**, **RANDARRAY**

47. **SUMPRODUCT** - Returns the sum of the products of corresponding ranges or arrays.
    Example: `SUMPRODUCT({1, 2}, {3, 4})` ➜ 3 * 8 ➜ 11
    Alternatives: **PRODUCT**

48. **MOD** - Returns the remainder from division.
    Example: MOD(10, 3) ➜ 10 / 3 ➜ 3 ⅓ ➜ 1
49. **ABS** - Returns the absolute value of a number.
    Example: ABS(-10) ➜ 10
50. **SEQUENCE** - Generates a list or matrix of sequential numbers, based on the arguments for rows, columns, start, and step.
    Example: SEQUENCE(4) ➜ {1; 2; 3; 4}

Additional Resources

YouTube Channel - Our YouTube channel contains tons of videos that will help you learn formulas and all other aspects of Excel.

Excel Campus Blog - If you prefer written articles, our award winning blog contains in-depth tutorials with annotated images in an easy to read format.

Week Excel Pro Tips Newsletter - If you received this list from a friend or coworker, join our free newsletter to receive Excel tips, techniques and news right to your inbox.

Modern Formulas Course - Learn how to write both simple and advanced formulas in the modern era of Excel with dynamic array formulas, spill ranges, error handling, dynamic formatting, interactive reporting, best practices, and more.

Elevate Excel Training Program - Our comprehensive, beginner to advanced training program covers all areas of Excel, data analysis, and data automation. Get access to our entire course library and our community forum where you can ask questions and connect with other professionals like yourself.

Notes:

_____

_____

_____

_____

_____

_____