

ADVANCED
EXCEL

COMPREHENSIVE STUDY NOTES

Complete Reference Guide for Excel Power Users

Formulas • Functions • Pivot Tables • Charts • Macros & VBA • Data Tools • Power Query

	Formulas & Functions	Charts & Visualization
	Macros & VBA	Data Tools

This guide covers 30+ pages of advanced Excel techniques, formulas, data analysis tools, and automation with real-world examples and diagrams.



TABLE OF CONTENTS

#	Topic	Page
1	Excel Interface & Navigation	3
2	Essential Formulas & Functions	4
3	Logical & Conditional Functions	6
4	Lookup & Reference Functions	8
5	Text Functions	10
6	Date & Time Functions	12
7	Statistical & Math Functions	13
8	Pivot Tables — Complete Guide	15
9	Charts & Data Visualization	18
10	Conditional Formatting	20
11	Data Validation & Protection	22
12	Advanced Sorting & Filtering	23
13	Named Ranges & Dynamic Arrays	24
14	Power Query — Data Transformation	25
15	Macros & VBA Automation	27
16	Excel Tables (ListObjects)	29
17	Financial Functions	30
18	Array Formulas & LAMBDA	31
19	Best Practices & Keyboard Shortcuts	32

1

Excel Interface & Navigation

Master the workspace before diving deep

Understanding the Excel Interface

Microsoft Excel's interface is organized into several key components. Mastering navigation and interface elements dramatically improves productivity and allows you to access powerful features quickly.

Component	Description	Shortcut / Tip
Ribbon	Tabbed toolbar with all commands organized by category	Alt to activate key tips
Name Box	Shows cell reference; type address to navigate quickly	Ctrl+G or F5 for Go To
Formula Bar	Displays/edits cell content and formulas	F2 to enter edit mode
Worksheet Tabs	Navigate between sheets in a workbook	Ctrl+PgUp/PgDn
Quick Access Toolbar	Customizable toolbar for frequent commands	Alt+1,2,3... to invoke
Status Bar	Shows SUM, AVERAGE, COUNT of selected cells	Right-click to customize
Freeze Panes	Lock rows/cols while scrolling	View → Freeze Panes
Split View	View different parts of a large sheet simultaneously	View → Split

Essential Navigation Shortcuts

Action	Shortcut
Go to cell A1	Ctrl + Home
Go to last used cell	Ctrl + End
Move to edge of data	Ctrl + Arrow Keys
Select entire column	Ctrl + Space
Select entire row	Shift + Space
Select all cells	Ctrl + A
Insert new worksheet	Shift + F11
Switch between workbooks	Ctrl + Tab
Find & Replace	Ctrl + H
Open Name Manager	Ctrl + F3

■ **Pro Tip:** Press Ctrl+; to insert today's date instantly, and Ctrl+Shift+; to insert the current time. Use Ctrl+` to toggle between showing formulas and values in all cells.

2

Essential Formulas & Functions

The backbone of every Excel spreadsheet

Formula Fundamentals

Every formula in Excel begins with the equals sign (=). Excel evaluates formulas using a specific order of operations. Understanding cell referencing types — relative, absolute, and mixed — is critical for building scalable models.

Cell Reference Types

Type	Syntax	Behavior	When to Use
Relative	A1	Adjusts when copied in any direction	General calculations
Absolute	\$A\$1	Never changes when copied (F4 to toggle)	Fixed constants, tax rates
Mixed (Row)	A\$1	Row locked, column free	Horizontal copy, fixed row
Mixed (Col)	\$A1	Column locked, row free	Vertical copy, fixed column

Operator Precedence

Excel follows PEMDAS/BODMAS order: Parentheses first, then Exponents (^), then Multiplication/Division, then Addition/Subtraction, then text concatenation (&), then comparison operators (<, >, =, <=, >=, <>).



Core Mathematical Functions

Function	Syntax	Description	Example
SUM	=SUM(range)	Adds all numbers in range	=SUM(A1:A100)
AVERAGE	=AVERAGE(range)	Arithmetic mean of values	=AVERAGE(B2:B50)
COUNT	=COUNT(range)	Counts cells with numbers	=COUNT(C1:C20)
COUNTA	=COUNTA(range)	Counts non-empty cells	=COUNTA(A:A)
COUNTBLANK	=COUNTBLANK(range)	Counts empty cells	=COUNTBLANK(D1:D10)
MAX	=MAX(range)	Returns largest value	=MAX(E2:E100)
MIN	=MIN(range)	Returns smallest value	=MIN(E2:E100)
PRODUCT	=PRODUCT(range)	Multiplies all numbers	=PRODUCT(F1:F5)
ABS	=ABS(number)	Returns absolute value	=ABS(-42)
ROUND	=ROUND(number, digits)	Rounds to specified decimal places	=ROUND(3.14159,2)
ROUNDUP	=ROUNDUP(number, digits)	Always rounds up	=ROUNDUP(2.1,0)
ROUNDDOWN	=ROUNDDOWN(number, digits)	Always rounds down	=ROUNDDOWN(9.9,0)

MOD	=MOD(number, divisor)	Returns remainder of division	=MOD(10,3) → 1
INT	=INT(number)	Rounds down to nearest integer	=INT(5.9) → 5
SQRT	=SQRT(number)	Square root	=SQRT(144) → 12
POWER	=POWER(base, exponent)	Raises to a power	=POWER(2,10) → 1024

■ SUMPRODUCT is one of the most powerful functions: it multiplies corresponding elements of arrays and returns the sum — essentially doing array math without Ctrl+Shift+Enter. Example:
 =SUMPRODUCT((A2:A100="East")*(B2:B100>1000)*C2:C100) sums column C where region is East AND sales > 1000.

3

Logical & Conditional Functions

Decision-making power in your spreadsheets

IF Function Family

The IF function is the foundation of conditional logic in Excel. It tests a condition and returns one value if TRUE, another if FALSE. Nested IFs and the newer IFS function handle multiple conditions elegantly.

=IF(logical_test, value_if_true, value_if_false)

Basic conditional — returns one of two values based on a test

=IFS(cond1,val1, cond2,val2, ..., TRUE,default)

Tests multiple conditions — cleaner alternative to nested IFs

=SWITCH(expr, val1,result1, val2,result2, ..., default)

Evaluates an expression against a list of values

Nested IF Example — Grade Calculator

Score Range	Grade	Nested IF Formula
90-100	A	=IF(A2>=90,"A",IF(A2>=80,"B",IF(A2>=70,"C",IF(A2>=60,"D","F"))))
80-89	B	(same formula evaluates progressively)
70-79	C	Better approach: =IFS(A2>=90,"A",A2>=80,"B",A2>=70,"C",A2>=60,"D",TRUE,"F")
60-69	D	
<60	F	

AND, OR, NOT Functions

Combine multiple conditions using AND (all must be true), OR (at least one must be true), and NOT (reverses the logic).

Function	Syntax	Returns TRUE when...	Example
AND	=AND(cond1, cond2, ...)	ALL conditions are TRUE	=AND(A1>0, A1<100)
OR	=OR(cond1, cond2, ...)	ANY condition is TRUE	=OR(B1="Yes",B1="y")
NOT	=NOT(condition)	Condition is FALSE	=NOT(ISBLANK(C1))
XOR	=XOR(cond1, cond2)	ODD number of TRUES	=XOR(A1,B1)

SUMIF, COUNTIF, AVERAGEIF — Conditional Aggregation

These functions apply aggregation only to cells that meet a specified criterion. The plural variants (SUMIFS, COUNTIFS, AVERAGEIFS) support multiple criteria.

Function	Syntax	Description
SUMIF	=SUMIF(range, criteria, sum_range)	Sum values where one condition is met

SUMIFS	=SUMIFS(sum_range, range1, crit1, range2, crit2, ...)	Sum values where multiple conditions are met
COUNTIF	=COUNTIF(range, criteria)	Count cells matching one condition
COUNTIFS	=COUNTIFS(range1, crit1, range2, crit2, ...)	Count cells matching multiple conditions
AVERAGEIF	=AVERAGEIF(range, criteria, avg_range)	Average values where condition is met

Wildcard Characters in Criteria

Wildcard	Meaning	Example Criteria	Matches
*	Any sequence of characters	"East*"	East, Eastern, Eastside
?	Any single character	"J?n"	Jan, Jun, Jon
~	Escape wildcard literal	"~*"	Literal asterisk *

■■ IFERROR and IFNA: Wrap formulas to handle errors gracefully.

=IFERROR(VLOOKUP(A1,Table,2,0),"Not Found") returns 'Not Found' instead of #N/A when a lookup fails.

IFNA only catches #N/A errors — useful when you want other error types to still surface.

4

Lookup & Reference Functions

Find, match, and retrieve data dynamically

VLOOKUP — Vertical Lookup

VLOOKUP searches the first column of a range for a value and returns a value from a specified column in the same row. It is one of the most commonly used functions but has several important limitations.

=VLOOKUP(lookup_value, table_array, col_index_num, [range_lookup])

range_lookup: FALSE = exact match (recommended), TRUE = approximate match

Parameter	Description	Common Mistake
lookup_value	Value to search for in the first column	Using text when column has numbers
table_array	The table range to search	Not locking range with \$ for copying
col_index_num	Column number to return (1=first col)	Hardcoding column numbers (breaks on insert)
range_lookup	FALSE for exact, TRUE for approximate	Omitting this — defaults to TRUE (approximate)!

■■ VLOOKUP Limitation: It can only look RIGHT — cannot return values to the left of the lookup column. Also, inserting columns breaks the col_index_num. Modern Excel users prefer INDEX/MATCH or XLOOKUP for these reasons.

INDEX & MATCH — The Power Combo

INDEX returns a value from a specific position in a range. MATCH finds the position of a value within a range. Combined, they form a flexible, bi-directional lookup that outperforms VLOOKUP.

=INDEX(return_range, MATCH(lookup_value, lookup_range, 0))

0 in MATCH = exact match | Replace VLOOKUP with this for flexibility

Function	Purpose	Key Argument
INDEX	Returns cell value at row/col intersection	=INDEX(A1:D100, row_num, col_num)
MATCH	Returns relative position of a value	=MATCH(value, range, 0) — 0 for exact
Combined	2-way lookup in any direction	=INDEX(cols, MATCH(row_val, rows, 0), MATCH(col_val, cols_h, 0))

XLOOKUP — The Modern Replacement (Excel 365/2021)

XLOOKUP replaces VLOOKUP, HLOOKUP, and INDEX/MATCH with a simpler, more powerful syntax. It looks left/right/up/down, handles errors natively, and can return entire arrays.

=XLOOKUP(lookup, lookup_array, return_array, [not_found], [match_mode], [search_mode])

match_mode: 0=exact, -1=exact/smaller, 1=exact/larger, 2=wildcard

Feature	VLOOKUP	INDEX/MATCH	XLOOKUP
Lookup Direction	Right only	Any direction	Any direction

Return Multiple Columns	No	With array	Yes, natively
Built-in Error Handling	No (use IFERROR)	No	Yes ([not_found])
Wildcard Search	With wildcards	With wildcards	Yes (match_mode=2)
Approximate Match	Yes	Yes	Yes
Reverse/Last Match	No	No	Yes (search_mode=-1)
Excel Version	2003+	2003+	365/2021+

Other Reference Functions

Function	Syntax	Description
HLOOKUP	=HLOOKUP(val,table,row,match)	Horizontal lookup — searches across top row
OFFSET	=OFFSET(ref,rows,cols,h,w)	Returns range offset from reference point
INDIRECT	=INDIRECT(text_ref)	Returns reference from a text string
CHOOSE	=CHOOSE(index, val1, val2,...)	Returns value from a list by index number
ADDRESS	=ADDRESS(row,col,type)	Builds a cell reference as text
ROW/COLUMN	=ROW(), =COLUMN()	Returns row or column number of a cell
ROWS/COLUMNS	=ROWS(range)	Returns count of rows or columns in range

5

Text Functions

Clean, manipulate, and transform text data

Text Manipulation Functions

Function	Syntax	Description	Example & Result
LEFT	=LEFT(text, n)	First n characters	=LEFT("Excel",3) → Exc
RIGHT	=RIGHT(text, n)	Last n characters	=RIGHT("Excel",3) → cel
MID	=MID(text, start, n)	n chars starting at position	=MID("Excel",2,3) → xce
LEN	=LEN(text)	Length of text string	=LEN("Hello") → 5
FIND	=FIND(find_text, text)	Position (case-sensitive)	=FIND("@",email)
SEARCH	=SEARCH(find_text, text)	Position (case-insensitive)	=SEARCH("excel",A1)
SUBSTITUTE	=SUBSTITUTE(text,old,new)	Replace text (all occurrences)	=SUBSTITUTE(A1," ","_")
REPLACE	=REPLACE(text,start,n,new)	Replace by position	=REPLACE(A1,1,3,"NEW")
TRIM	=TRIM(text)	Remove extra spaces	=TRIM(" hello ")
CLEAN	=CLEAN(text)	Remove non-printable characters	=CLEAN(A1)
UPPER	=UPPER(text)	Convert to UPPERCASE	=UPPER("hello")
LOWER	=LOWER(text)	Convert to lowercase	=LOWER("HELLO")
PROPER	=PROPER(text)	Capitalise Each Word	=PROPER("john doe")
CONCAT	=CONCAT(text1, text2,...)	Join text (modern)	=CONCAT(A1," ",B1)
TEXTJOIN	=TEXTJOIN(delim,ignore,rng)	Join with delimiter	=TEXTJOIN(" ",TRUE,A1:A5)
TEXT	=TEXT(value, format)	Format number as text	=TEXT(TODAY(),"DD/MM/YYYY")
VALUE	=VALUE(text)	Convert text number to number	=VALUE("1,234") → 1234
REPT	=REPT(text, n)	Repeat text n times	=REPT("★",5)
NUMBERVALUE	=NUMBERVALUE(text)	Text to number with locale	=NUMBERVALUE("1.234,5",".",",":")

■ **Flash Fill (Ctrl+E):** Excel 2013+ can detect patterns from examples you type and fill a column automatically. Type the desired result in the first row, then press Ctrl+E — Excel fills the rest! Great for splitting names, reformatting phone numbers, or extracting parts of text without formulas.

6

Date & Time Functions

Work with dates, times, and time intelligence

Date & Time Functions Reference

Function	Syntax	Description	Example
TODAY	=TODAY()	Current date (updates daily)	=TODAY()
NOW	=NOW()	Current date and time	=NOW()
DATE	=DATE(year, month, day)	Create a date from parts	=DATE(2024,12,25)
TIME	=TIME(hour, min, sec)	Create a time value	=TIME(14,30,0)
YEAR	=YEAR(date)	Extract year	=YEAR(A1) → 2024
MONTH	=MONTH(date)	Extract month (1-12)	=MONTH(A1) → 12
DAY	=DAY(date)	Extract day of month	=DAY(A1) → 25
HOUR	=HOUR(time)	Extract hour (0-23)	=HOUR(A1) → 14
MINUTE	=MINUTE(time)	Extract minutes	=MINUTE(A1) → 30
WEEKDAY	=WEEKDAY(date, [type])	Day of week (1-7)	=WEEKDAY(A1,2) → 1=Mon
WEEKNUM	=WEEKNUM(date)	Week number of year	=WEEKNUM(A1)
DATEDIF	=DATEDIF(start,end,unit)	Difference between dates	=DATEDIF(A1,B1,"Y")
NETWORKDAYS	=NETWORKDAYS(start,end,[hols])	Working days between dates	=NETWORKDAYS(A1,B1,Hols)
WORKDAY	=WORKDAY(start, days, [hols])	Date N working days from start	=WORKDAY(A1,10)
EDATE	=EDATE(date, months)	Date N months from date	=EDATE(A1,3)
EOMONTH	=EOMONTH(date, months)	Last day of month N months on	=EOMONTH(A1,0)
TEXT	=TEXT(date,"MMMM")	Format date as text	=TEXT(A1,"dddd") → Monday

■ DATEDIF units: "Y" = complete years, "M" = complete months, "D" = days, "YM" = months ignoring years, "MD" = days ignoring months and years. Example age formula: =DATEDIF(birthdate, TODAY(), "Y") & " years, " & DATEDIF(birthdate, TODAY(), "YM") & " months"

7

Statistical & Math Functions

Analyze data with built-in statistics

Statistical Functions

Function	Syntax	Description
MEDIAN	=MEDIAN(range)	Middle value of a dataset
MODE.SNGL	=MODE.SNGL(range)	Most frequently occurring value
STDEV.S	=STDEV.S(range)	Sample standard deviation
STDEV.P	=STDEV.P(range)	Population standard deviation
VAR.S	=VAR.S(range)	Sample variance
PERCENTILE.INC	=PERCENTILE.INC(range,k)	k-th percentile (0 to 1)
QUARTILE.INC	=QUARTILE.INC(range,quart)	Q1=1, Q2=2, Q3=3, Min=0, Max=4
RANK.EQ	=RANK.EQ(num, range, [order])	Rank of a number in a list
LARGE	=LARGE(range, k)	k-th largest value
SMALL	=SMALL(range, k)	k-th smallest value
CORREL	=CORREL(range1, range2)	Correlation coefficient (-1 to 1)
FORECAST.LINEAR	=FORECAST.LINEAR(x,ys,xs)	Predict future value by linear trend
NORM.DIST	=NORM.DIST(x,mean,sd,TRUE)	Cumulative normal distribution
CONFIDENCE.NORM	=CONFIDENCE.NORM(α , σ ,n)	Confidence interval half-width
FREQUENCY	=FREQUENCY(data,bins)	Array: count by bin (Ctrl+Shift+Enter)

Rounding Function Comparison

Function	Value Input	Result	Notes
ROUND	2.567	2.57	Standard rounding, 2 decimal places
ROUNDUP	2.501	3.00	Always away from zero
ROUNDDOWN	2.999	2.00	Always toward zero
CEILING	2.3	3.00	Up to nearest multiple of significance
FLOOR	2.9	2.00	Down to nearest multiple
MROUND	2.4	2.00	Round to nearest multiple (e.g. 0.5)
INT	2.9	2	Integer part only (rounds down always)
TRUNC	-2.9	-2	Truncate (toward zero, unlike INT)

Monthly Performance Chart (% of Target)

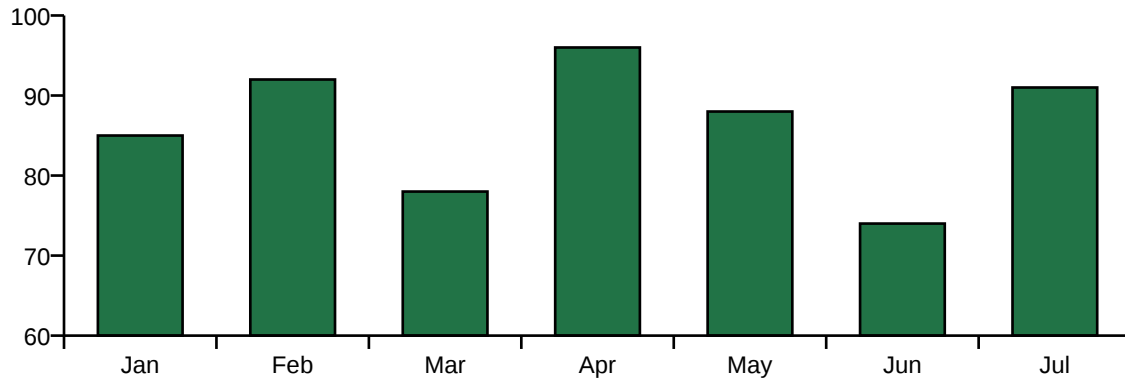


Figure: Monthly Performance Chart — built with Excel's chart engine

8

Pivot Tables — Complete Guide

Summarize millions of rows in seconds

What is a Pivot Table?

A PivotTable is an interactive summary tool that rapidly aggregates, sorts, counts, totals, or averages data stored in a spreadsheet. It allows you to rotate ('pivot') rows and columns to see different summaries of source data without writing a single formula.

RAW DATA			PIVOT TABLE			
Region	Product	Sales	Region	Laptop	Phone	Total
North			East	2800	1500	4300
South			North	5200	4200	9400
North			South	3900	3100	7000
East			Total	11900	8800	20700
South						
East						

Figure: Raw data → Pivot Table transformation

Creating a Pivot Table



The Four Pivot Table Areas

Area	Purpose	Examples
FILTERS (Report Filter)	Filter the entire report by field values	Year, Region, Department
ROWS	Group data into rows	Product, Sales Rep, Month
COLUMNS	Group data into columns (cross-tab)	Quarter, Category, Status
VALUES	Aggregated data (SUM, COUNT, AVG...)	Revenue, Units, Profit Margin

Value Field Settings — Summarize By

Summary Type	Use For
Sum	Total of numeric values (default for numbers)
Count	Number of records (default for text)
Average	Mean of values

Max / Min	Highest or lowest value
Product	Multiplication of all values
StdDev / Var	Statistical dispersion measures
Count Numbers	Only counts numeric cells (ignores text/blanks)

Show Values As — Calculated Display

Instead of raw sums, pivot values can be displayed as % of total, running totals, rank, difference from, and more — without any formulas:

Option	Displays Value As
% of Grand Total	Each cell as a % of the overall total
% of Column Total	% within each column
% of Row Total	% within each row
% of Parent Total	% relative to parent group subtotal
Running Total In	Cumulative sum by a base field
% Running Total In	Cumulative % by a base field
Rank Smallest to Largest	Rank within the field
Index	Relative importance index

Pivot Table Tips & Tricks

- Double-click any value cell to drill down and see the source records
- Right-click → Group to group dates by Year/Quarter/Month automatically
- Use Slicers (Insert → Slicer) for visual, clickable filters
- Use Timelines (Insert → Timeline) for date-based filtering
- Calculated Fields: Analyze → Fields, Items & Sets → Calculated Field
- GETPIVOTDATA function retrieves specific data from a pivot table by reference
- Refresh data with Alt+F5 (single pivot) or Ctrl+Alt+F5 (all pivots)
- Change the pivot table layout: Report Layout (Compact / Outline / Tabular)

■ **Power Tip:** To create a PivotChart alongside your PivotTable, go to PivotTable Analyze → PivotChart. It stays linked to the pivot — filter the pivot and the chart updates instantly. Add a Slicer to control both simultaneously.

9

Charts & Data Visualization

Turn numbers into compelling visual stories

Choosing the Right Chart Type

Chart Type	Best For	Avoid When
Column / Bar	Comparing categories side by side	Too many categories (>8-10)
Line	Trends over time	Non-continuous data
Pie / Doughnut	Part-to-whole proportions (max 5-6 slices)	Too many slices or all similar sizes
Area	Cumulative totals over time	Individual series are important
Scatter	Correlation between two variables	Categorical x-axis
Bubble	3-variable relationships	Exact values need to be read
Waterfall	Visualizing cumulative effect of +/- changes	Simple totals
Histogram	Distribution of a single dataset	Comparing multiple distributions
Box & Whisker	Statistical spread and outliers	Non-technical audiences
Treemap	Hierarchical part-to-whole	When exact values matter most
Funnel	Sales pipeline / process stages	Non-sequential data
Map Chart	Geographic data by country/state	Granular sub-regional data
Sparklines	Trends in a single cell (mini charts)	When detail is required
Combo	Two different data types on one chart	More than 2 chart types

Revenue vs Target Trend (thousands)

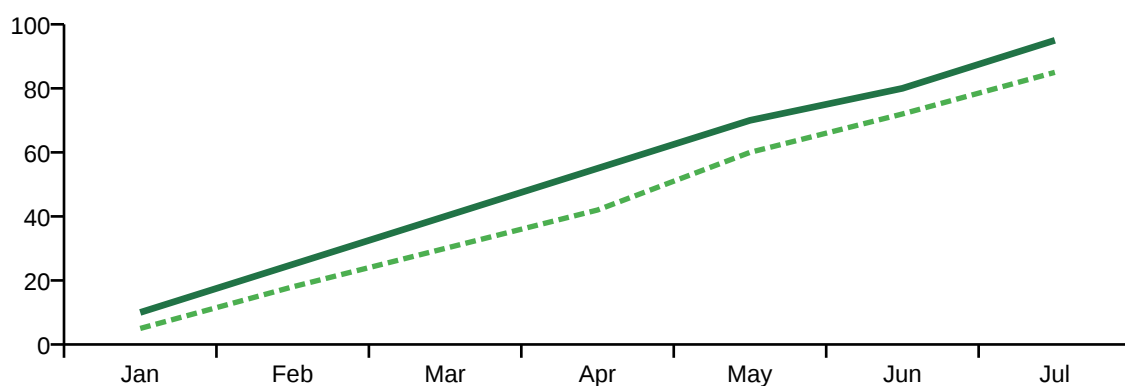


Figure: Line chart comparing Revenue vs Target over time

Chart Formatting Best Practices

- Remove chartjunk: delete gridlines, borders, and 3D effects that don't add information
- Direct labeling: place data labels ON the bars/lines rather than using a legend when possible
- Color with purpose: use one accent color for the key data series; grey for context
- Consistent axis scale: always start the y-axis at zero for bar/column charts
- Chart title should state the insight, not just describe the data
- Annotate: add text boxes to call out key events or inflection points

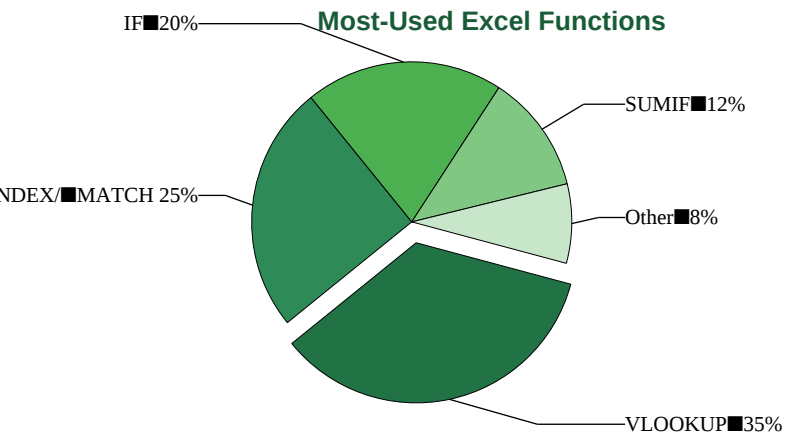


Figure: Distribution of most-used Excel functions among power users

Dynamic Charts with Named Ranges / Tables

Convert data to an Excel Table (Ctrl+T) before creating a chart. Charts based on Tables automatically expand when new rows are added — no manual range updates needed. For complex dynamic charts, use OFFSET or INDEX-based named ranges in the chart's data series formula.

10

Conditional Formatting

Make your data visually intelligent

Conditional Formatting Overview

Conditional Formatting applies automatic formatting (colors, icons, data bars) to cells based on rules. It makes patterns, outliers, and trends immediately visible without changing the underlying data.

Rule Type	What It Does	Best Use Case
Highlight Cell Rules	Format cells matching a condition (>, <, =, between, text, dates)	Flag overdue dates, negative values
Top/Bottom Rules	Highlight top N%, bottom N, above/below average	Identify top performers
Data Bars	Horizontal bars proportional to cell value	Quick in-cell bar chart
Color Scales	Gradient color based on relative value (2 or 3 color)	Heat maps, score grids
Icon Sets	Traffic lights, arrows, stars based on thresholds	Dashboard status indicators
Formula-Based	Apply format based on any formula result (most flexible)	Highlight entire rows, complex logic
Duplicate Values	Flag duplicate or unique values in a range	Data quality checks

Formula-Based Rules — Power Examples

Goal	Formula (applied to row 2)	Effect
Highlight entire row where Sales > 10000	=C2>10000	Row turns green
Highlight every other row (banding)	=MOD(ROW(),2)=0	Alternating row color
Flag rows where deadline has passed	=D2<TODAY()	Overdue rows turn red
Highlight duplicate names in col A	=COUNTIF(\$A\$1:\$A2,A2)>1	Duplicate cells highlighted
Mark rows missing required data	=ISBLANK(\$B2)	Missing cells turn yellow
Highlight max value in a column	=A2=MAX(\$A:\$A)	Maximum cell highlighted

■ Managing Rules: Home → Conditional Formatting → Manage Rules. Rules are evaluated top-to-bottom and you can check 'Stop if True' to prevent lower-priority rules from overriding. Use 'Clear Rules' to remove formatting quickly.

11

Data Validation & Protection

Control input and secure your workbooks

Data Validation

Data Validation (Data → Data Validation) restricts what users can enter into cells, ensuring data integrity. It can create dropdown lists, enforce number ranges, require specific date ranges, and validate text length.

Validation Type	Allow Setting	Example Criteria
Whole Number	Whole number	Between 1 and 100
Decimal	Decimal	Greater than 0
List	List	Source: Yes,No,Pending OR =\$A\$1:\$A\$10
Date	Date	Between =TODAY()-30 and =TODAY()
Time	Time	Between 09:00 and 17:00
Text Length	Text length	Less than or equal to 50
Custom Formula	Custom	=ISNUMBER(A1), =LEN(A1)=10, =COUNTIF(\$A:\$A,A1)=1

Workbook & Worksheet Protection

Protection Level	How To	What It Protects
Worksheet	Review → Protect Sheet	Prevents editing locked cells; can allow specific actions
Workbook Structure	Review → Protect Workbook	Prevents adding/deleting/renaming sheets
Cell Locking	Format Cells → Protection → Locked	Mark cells as locked (active when sheet is protected)
File Password	File → Info → Protect Workbook → Encrypt	Requires password to open the file
VBA Project	VBA Editor → Tools → VBA Project Properties	Hides and locks VBA code

12

Advanced Sorting & Filtering

Find exactly the data you need

Advanced Sort Options

Sort Feature	Description
Multi-Level Sort	Sort by multiple columns — Data → Sort → Add Level (e.g. Region then Sales)
Sort by Color	Sort rows by cell fill color or font color
Sort by Icon	Sort by conditional formatting icon set values
Custom Sort Order	Define custom order (e.g. Mon,Tue,Wed or Low,Med,High) via Custom List
Left to Right	Sort columns instead of rows — Sort → Options → Left to Right
Case Sensitive	Case-sensitive text sort — Sort → Options → Case Sensitive

Advanced Filter

Unlike AutoFilter, Advanced Filter (Data → Advanced) lets you specify complex criteria in a separate range on the worksheet — including OR conditions across different columns and formula-based criteria. You can also extract filtered results to a different location.

Advanced Filter Criteria Range Example

Region	Product	Min Sales
North		
	Laptop	
East		5000

Rows in criteria range = OR conditions; columns = AND conditions

FILTER Function (Excel 365)

=FILTER(array, include, [if_empty])

Returns rows/cols from array where include is TRUE — spills results dynamically

■ FILTER example: =FILTER(A2:D100, (B2:B100="East")*(C2:C100>5000), "No results") returns all rows where Region=East AND Sales>5000. Combine with SORT:
=SORT(FILTER(A2:D100,B2:B100="East"),3,-1) to sort results by column 3 descending.

13

Named Ranges & Dynamic Arrays

Build cleaner, more powerful formulas

Named Ranges

Named ranges assign a meaningful name to a cell or range, making formulas readable and easier to maintain. Manage them via Formulas → Name Manager.

How to Define	Method
Name Box (quick)	Click Name Box → type name → press Enter
Define Name dialog	Formulas → Define Name → enter name & reference
Name Manager	Formulas → Name Manager → New (full control)
From Selection	Select range with labels → Formulas → Create from Selection
Dynamic Named Range	=OFFSET(\$A\$1,0,0,COUNTA(\$A:\$A),1) — expands automatically
Table Structured Ref	Excel Tables auto-create names: Table1[Sales], Table1[#Totals]

Dynamic Array Functions (Excel 365)

Function	Syntax	Description
FILTER	=FILTER(array, include)	Filter array by condition — spills results
SORT	=SORT(array, [col], [order])	Sort an array; 1=asc, -1=desc
SORTBY	=SORTBY(array, by_array, order)	Sort by one or more separate arrays
UNIQUE	=UNIQUE(array, [by_col])	Return unique values from a range
SEQUENCE	=SEQUENCE(rows,[cols],[start],[step])	Generate number sequence array
RANDARRAY	=RANDARRAY(rows,[cols],[min],[max])	Fill array with random numbers
TOCOL	=TOCOL(array)	Stack array into a single column
TOROW	=TOROW(array)	Stack array into a single row
HSTACK	=HSTACK(arr1, arr2, ...)	Stack arrays horizontally
VSTACK	=VSTACK(arr1, arr2, ...)	Stack arrays vertically
TEXTSPLIT	=TEXTSPLIT(text, col_delim)	Split text into array by delimiter
TEXTBEFORE	=TEXTBEFORE(text, delimiter)	Text before a delimiter
TEXTAFTER	=TEXTAFTER(text, delimiter)	Text after a delimiter
CHOOSEROWS	=CHOOSEROWS(array, row1, row2)	Pick specific rows from an array
CHOOSECOLS	=CHOOSECOLS(array, col1, col2)	Pick specific columns from an array

■ **Spill Range Reference:** Dynamic array results 'spill' into adjacent cells. Reference the entire spill range with the # operator: =SUM(A1#) sums all spilled values from the formula in A1. Use @A1# to reference only the first value (implicit intersection — legacy behavior).

14

Power Query — Data Transformation

ETL powerhouse built into Excel

What is Power Query?

Power Query (Get & Transform Data) is Excel's built-in ETL (Extract, Transform, Load) engine. It connects to virtually any data source, applies a series of transformation steps, and loads clean data into Excel — all repeatable with one click of 'Refresh'.



Data Sources Power Query Connects To

Category	Sources
Files	Excel, CSV, Text, JSON, XML, PDF, Folder (bulk import)
Databases	SQL Server, Access, Oracle, MySQL, PostgreSQL, SAP HANA
Online	SharePoint, OneDrive, Salesforce, Dynamics 365, OData feeds
Web	Web pages (HTML tables), REST APIs
Azure	Azure SQL, Azure Data Lake, Azure Analysis Services, Blob Storage
Other	Python, R (via connector), ODBC, OLEDB, Hadoop HDFS

Key Transformation Operations

Operation	Description	Common Use
Remove Columns	Delete irrelevant columns	Strip out unnecessary fields from source
Filter Rows	Keep/exclude rows matching criteria	Remove blanks, filter by value/date
Change Data Type	Set proper type (Text, Number, Date...)	Fix auto-detected wrong types
Split Column	By delimiter or character position	Split 'First Last' into two columns
Merge Columns	Combine with separator	Combine date+time, name fields
Replace Values	Find & replace in a column	Standardize text (N→North)
Pivot / Unpivot	Reshape data structure	Unpivot month columns to rows
Group By	Aggregate data by category	Sum sales by region
Merge Queries	JOIN two tables (like SQL JOIN)	Lookup tables, combine sources
Append Queries	Stack tables vertically (like UNION)	Combine multiple file imports
Add Custom Column	Calculate new column with M formula	Concatenate, if/then logic
Conditional Column	IF/THEN/ELSE without formulas	Categorize values

Fill Down/Up	Fill merged/blank cells	Fix merged cell exports
Trim / Clean	Remove whitespace, non-printable chars	Clean dirty text data

■ **M Language:** Every Power Query step is recorded in the M formula language. View it in Advanced Editor (Home → Advanced Editor). You can write M directly for complex transformations not available in the GUI.
 Example: `Table.SelectRows(Source, each [Sales] > 1000 and [Region] = "East")`

15

Macros & VBA Automation

Automate repetitive tasks with code

Recording Macros

A macro is a recorded sequence of actions. Use View → Macros → Record Macro or Developer → Record Macro. Excel stores the actions as VBA code which you can play back, edit, and enhance. Always save as .xlsm (macro-enabled) format.



VBA Fundamentals

VBA (Visual Basic for Applications) is the programming language built into Excel. Access the editor with Alt+F11. Code is organized into Modules, and actions are written as Sub procedures or Function procedures.

Concept	Syntax / Example	Description
Sub Procedure	Sub MyMacro() 'code here End Sub	A macro that performs actions
Function	Function Cube(n) As Double Cube = n^3 End Function	Returns a value (use in cells)
Variable	Dim x As Integer x = 10	Store data in memory
Comment	'This is a comment	Non-executing documentation
String	Dim name As String name = "Excel"	Text data type
Range Object	Range("A1").Value = "Hello"	Interact with cells
Cells Object	Cells(1,1).Value = "Hello"	Cells(row, column) syntax
ActiveCell	ActiveCell.Value = "test"	Currently selected cell
Selection	Selection.Font.Bold = True	Currently selected range
Worksheets	Worksheets("Sheet1").Activate	Reference a sheet by name
MsgBox	MsgBox "Done!"	Display a message to user
InputBox	x = InputBox("Enter value:")	Prompt user for input
If Statement	If x > 10 Then '... End If	Conditional logic

For Loop	For i = 1 To 10 '... Next i	Repeat N times
For Each	For Each cell In Range("A1:A10") Next	Loop over collection
Do While	Do While x < 100 x = x*2 Loop	Loop while condition is true
With Block	With Range("A1") .Bold=True End With	Avoid repeating object ref
Error Handling	On Error Resume Next On Error GoTo Err1	Handle runtime errors

Practical VBA Examples

Task	VBA Code Snippet
Format all sheets the same	<pre>Sub FormatAllSheets() Dim ws As Worksheet For Each ws In ThisWorkbook.Worksheets ws.Tab.ColorIndex = 4 Next ws End Sub</pre>
Auto-sort on data change	<pre>Private Sub Worksheet_Change(ByVal Target As Range) If Not Intersect(Target, Columns("A")) Is Nothing Then Range("A1").CurrentRegion.Sort Key1:=Range("A1") End If End Sub</pre>
Create a report with timestamp	<pre>Sub CreateReport() Dim ws As Worksheet Set ws = Worksheets.Add ws.Name = "Report_" & Format(Now, "YYYYMMDD") ws.Range("A1") = "Generated: " & Now End Sub</pre>

■ Security: Macros are disabled by default. Enable via File → Options → Trust Center → Trust Center Settings → Macro Settings. Never enable macros in files from unknown sources — malicious macros can harm your system.

16

Excel Tables (ListObjects)

Structure your data for maximum power

Why Use Excel Tables?

Converting a data range into an Excel Table (Ctrl+T) unlocks automatic features: structured references, auto-expanding formulas, built-in filter dropdowns, total rows, and seamless integration with PivotTables, Power Query, and charts.

Feature	Plain Range	Excel Table
Auto-expand formulas	Must copy manually	Formulas copy to new rows automatically
Structured references	=A\$1:\$A\$100	=Table1[Sales] — readable and dynamic
Filter dropdowns	Add manually	Built-in on every column header
Total row	Write formula	Toggle on with checkbox — multiple aggregations
Styles & banding	Manual format	One-click professional formatting + auto-alternating rows
Chart/Pivot dynamic	Must update range	Auto-include new data in connected objects
Duplicate check	Use formulas	Use Table tools with conditional formatting
VLOOKUP range	Must be absolute	Table column references never break

Structured Reference Syntax

Specifier	Reference	Meaning
[Column Name]	Table1[Sales]	All data cells in Sales column
[#Headers]	Table1[#Headers],[Name]]	Header cell of Name column
[#Totals]	Table1[#Totals],[Sales]]	Total row cell of Sales column
[#All]	Table1[#All]	Entire table including headers and totals
[#Data]	Table1[#Data],[Name]]	Data cells only (no headers/totals)
[@Column]	[@Sales]	Same row intersection — for row formulas
Multi-column	Table1[[Col1]:[Col3]]	Range spanning Col1 through Col3

17

Financial Functions

Built-in tools for finance and analysis

Time Value of Money Functions

Function	Syntax	Description	Example
PV	=PV(rate,nper,pmt,[fv],[type])	Present Value of future cash flows	=PV(5%/12,60,-500)
FV	=FV(rate,nper,pmt,[pv],[type])	Future Value of an investment	=FV(8%/12,120,-200)
PMT	=PMT(rate,nper,pv,[fv],[type])	Periodic payment amount	=PMT(6%/12,360,200000)
NPER	=NPER(rate,pmt,pv,[fv],[type])	Number of periods needed	=NPER(5%/12,-500,20000)
RATE	=RATE(nper,pmt,pv,[fv],[type])	Interest rate per period	=RATE(60,-400,18000)*12
NPV	=NPV(rate,value1,value2,...)	Net Present Value of cash flows	=NPV(10%,B2:B6)-B1
IRR	=IRR(values,[guess])	Internal Rate of Return	=IRR(B1:B6)
MIRR	=MIRR(values,finance_rate,reinvest)	Modified IRR	=MIRR(B1:B6,10%,12%)
XNPV	=XNPV(rate,values,dates)	NPV with irregular dates	=XNPV(10%,C2:C7,D2:D7)
XIRR	=XIRR(values,dates,[guess])	IRR with irregular dates	=XIRR(C2:C7,D2:D7)
SLN	=SLN(cost,salvage,life)	Straight-line depreciation	=SLN(50000,5000,5)
DB	=DB(cost,salvage,life,period)	Declining balance depreciation	=DB(50000,5000,5,1)
PRICE	=PRICE(settle,mature,rate,yld,redemp,freq)	Bond price per 100 face	=PRICE(...)
YIELD	=YIELD(settle,mature,rate,price,redemp,freq)	Bond yield to maturity	=YIELD(...)

18

Array Formulas & LAMBDA

Next-level formula techniques

Traditional Array Formulas (CSE)

Before dynamic arrays, Ctrl+Shift+Enter (CSE) was required to evaluate formulas across arrays. These formulas appear in {curly braces} in the formula bar. They are still valid but largely replaced by dynamic array functions in Excel 365.

Goal	CSE Array Formula	Modern Equivalent
Count cells matching multiple criteria	<code>{=SUM((A2:A10="East")*(B2:B10>500))}</code>	<code>COUNTIFS(A2:A10,"East",B2:B10,">500")</code>
Average of top 3 values	<code>{=AVERAGE(LARGE(A2:A20,{1,2,3}))}</code>	<code>=AVERAGE(LARGE(A2:A20,{1,2,3}))</code>
Sum of n largest values	<code>{=SUM(LARGE(A2:A20,ROW(INDIRECT("1:5"))))}</code>	<code>=SUM(LARGE(A2:A20,SEQUENCE(5)))</code>
Two-way lookup	<code>=INDEX(B2:D10,MATCH(G1,A2:A10,0),MATCH(G2,B1:D1,0))</code>	<code>=XLOOKUP(G2,B1:D1,A2:A10,XLOOKUP(G1,A2:A10,B2:D10))</code>

LAMBDA Function — Custom Functions Without VBA

LAMBDA (Excel 365) lets you define reusable custom functions using Excel formula syntax — no VBA required. Define once in the Name Manager, use everywhere in the workbook.

=LAMBDA(param1, param2, ..., formula_body)

Define: Name Manager → New → Name: MyFunc → Refers To: =LAMBDA(x, y, x^2 + y^2)

Function Name	Definition	How to Call
CircleArea	<code>=LAMBDA(r, PI()*r^2)</code>	<code>=CircleArea(5) → 78.54</code>
GrowthRate	<code>=LAMBDA(old,new, (new-old)/old)</code>	<code>=GrowthRate(100,120) → 0.2</code>
TitleCase	<code>=LAMBDA(t, PROPER(TRIM(t)))</code>	<code>=TitleCase(" hello world ")</code>
AgeYears	<code>=LAMBDA(dob, DATEDIF(dob,TODAY(),"Y"))</code>	<code>=AgeYears(A1) → age in years</code>

■ **LET Function:** `=LET(name1, value1, name2, value2, ..., formula)` defines named variables within a formula for readability and performance (calculated once, not repeatedly). Example: `=LET(x, A1*1.1, y, B1*0.9, x+y)` is cleaner than `=(A1*1.1)+(B1*0.9)` and calculates each sub-expression only once.

19

Best Practices & Keyboard Shortcuts

Work faster and build better spreadsheets

Excel Best Practices

- One row header rule: keep exactly ONE row of column headers — no merged header rows
- No blank rows or columns inside your data range — they break Table logic and CTRL+SHIFT+END
- Separate data, calculations, and presentation: raw data on one sheet, formulas on another, output/dashboard on another
- Document assumptions: use a dedicated 'Assumptions' or 'Parameters' sheet with named ranges
- Avoid hardcoding values in formulas — put constants in named cells and reference them
- Consistent data types: never mix numbers and text in the same column
- Use tables (Ctrl+T) for all structured data — they auto-expand and use structured references
- Color-code your workbook: inputs (blue), calculated (black/formula), outputs (green), warnings (red)
- Protect formula cells: lock cells with formulas (Format Cells → Protection) before sharing
- Version control: include date in file names and keep a changelog tab for shared workbooks

Essential Keyboard Shortcuts Reference

Action	Windows	Mac
Undo / Redo	Ctrl+Z / Ctrl+Y	Cmd+Z / Cmd+Y
Save / Save As	Ctrl+S / F12	Cmd+S / Cmd+Shift+S
New workbook	Ctrl+N	Cmd+N
Bold / Italic / Underline	Ctrl+B / I / U	Cmd+B / I / U
Format Cells dialog	Ctrl+1	Cmd+1
Insert row / column	Ctrl+Shift++ (plus)	Ctrl+Shift++
Delete row / column	Ctrl+- (minus)	Ctrl+-
AutoSum selected range	Alt+=	Cmd+Shift+T
Fill Down / Fill Right	Ctrl+D / Ctrl+R	Ctrl+D / Ctrl+R
Enter array formula	Ctrl+Shift+Enter	Ctrl+Shift+Return
Absolute reference (F4)	F4 (toggles \$ signs)	Cmd+T
Select current region	Ctrl+Shift+*	Cmd+Shift+Space
Open Find & Replace	Ctrl+H	Cmd+H
Paste Special	Ctrl+Alt+V	Ctrl+Cmd+V
Copy cell above	Ctrl+'	Ctrl+'
Insert current date	Ctrl+;	Ctrl+;
Insert current time	Ctrl+Shift+;	Cmd+;
Show all formulas	Ctrl+` (backtick)	Ctrl+`

Create chart (in sheet)	Alt+F1	Fn+Alt+F1
Create chart (new sheet)	F11	Fn+F11
Add filter to headers	Ctrl+Shift+L	Cmd+Shift+F
Flash Fill	Ctrl+E	Cmd+E
Open Power Query editor	Data → Get Data	Data → Get Data
Open VBA editor	Alt+F11	Opt+F11
Run macro	Alt+F8	Opt+F8
Freeze/unfreeze panes	View → Freeze Panes	View → Freeze Panes
Group rows/columns	Alt+Shift+→	Cmd+Shift+K
Ungroup rows/columns	Alt+Shift+←	Cmd+Shift+J

Common Excel Errors & Fixes

Error	Meaning	Common Causes & Fixes
#DIV/0!	Division by zero	Denominator is 0 or empty → use IFERROR or IF(B1=0,0,A1/B1)
#VALUE!	Wrong data type	Text in numeric formula → check cell types, use VALUE()
#REF!	Invalid reference	Deleted row/column → check formula, fix reference
#NAME?	Unrecognized name	Typo in function/named range → check spelling
#N/A	Value not found	VLOOKUP/MATCH can't find value → use IFERROR or XLOOKUP
#NULL!	Invalid intersection	Space instead of comma between ranges → fix syntax
#NUM!	Invalid number	SQRT of negative, out-of-range RATE iteration → check inputs
#####	Column too narrow	Widen the column → double-click column border
Circular!	Circular reference	Formula refers to its own cell → trace dependents (Formulas tab)

■ **Formula Auditing Tools:** Formulas → Trace Precedents shows arrows to cells that feed into the selected cell. Trace Dependents shows where the selected cell is used. Evaluate Formula (Alt+TUF) steps through a formula's calculation one element at a time — invaluable for debugging complex nested formulas.

Quick Reference: Function Categories Summary

Category	Key Functions	Primary Use
Math & Stats	SUM, AVERAGE, COUNT, ROUND, SUMPRODUCT	Numerical computation
Logical	IF, IFS, AND, OR, NOT, IFERROR, SWITCH	Decision making
Lookup	VLOOKUP, INDEX, MATCH, XLOOKUP, OFFSET	Data retrieval
Text	LEFT, MID, TRIM, SUBSTITUTE, TEXTJOIN	String manipulation
Date & Time	TODAY, DATEDIF, NETWORKDAYS, EOMONTH	Date calculations
Financial	PMT, NPV, IRR, FV, PV, XIRR	Financial modelling

Dynamic Array	FILTER, SORT, UNIQUE, SEQUENCE, VSTACK	Modern data operations
Information	ISBLANK, ISNUMBER, ISTEXT, ISERROR, TYPE	Data type checking

■ You are now an Advanced Excel Power User!

Practice daily • Build real projects • Explore Power BI next