



EXCEL & ADVANCE EXCEL

CONTENT

Chapter 1 : EXCEL BASICS.....	5
1.1 — Introduction to Excel & Business Applications.....	5
1.2 — Excel Interface	6
1.3 — Workbook & Worksheet Management.....	7
1.4 — Data Entry Techniques.....	8
1.5 — Cell Referencing (Relative, Absolute & Mixed)	9
1.6 — Formatting & Number Formats.....	11
1.7 — AutoFill & Flash Fill	12
1.8 — Sorting & Filtering.....	14
1.9 — Excel Tables	16
1.10 — Basic Charts	19
1.11 — Freeze Panes & Split	20
1.12 — Named Ranges.....	21
1.13 — Page Setup & Printing.....	23
1.14 — Workbook & Worksheet Protection.....	24
Chapter 2 – Advanced Formulas	26
Sample Data Tables — Used Throughout This Guide	27
Table 1 — Employee Master (Sheet: EmpData)	27
Table 2 — Sales Transactions (Sheet: SalesData)	27
Table 3 — Course Fee Table (Sheet: Courses)	27
Section 1 — Lookup & Reference Functions.....	28
1.2 HLOOKUP — Horizontal Lookup	29
1.3 INDEX + MATCH — The Professional Alternative	30
VLOOKUP vs INDEX-MATCH Comparison	31
1.4 XLOOKUP — Modern Replacement (Excel 2021+).....	31
Section 2 — Logical Functions	33
2.1 IF — Basic Decision Making.....	33
2.2 Nested IF — Multiple Conditions.....	33
2.3 IFS — Cleaner Alternative to Nested IF.....	34
2.4 AND / OR — Combine Multiple Conditions	34
Section 3 — Text Functions.....	36
3.1 LEFT, RIGHT, MID — Extract Parts of Text	36
3.2 LEN — Count Characters.....	36

3.3 TRIM — Remove Extra Spaces	36
3.4 CONCAT and TEXTJOIN — Combine Text.....	37
Section 4 — Date & Time Functions	38
4.1 TODAY and NOW.....	38
4.2 DATEDIF — Calculate Age / Tenure / Duration.....	38
4.3 EOMONTH — End of Month Date.....	39
4.4 NETWORKDAYS — Working Days Excluding Weekends	39
Industry Use Cases — Date Functions	40
Section 5 — Statistical & Math Functions.....	41
5.1 COUNTIF and COUNTIFS.....	41
5.2 SUMIF and SUMIFS.....	41
5.3 AVERAGEIF and AVERAGEIFS	42
Section 6 — Array & Dynamic Array Functions	43
6.1 UNIQUE — Extract Distinct Values.....	43
6.2 SORT — Sort a Range Dynamically.....	43
6.3 FILTER — Extract Rows Meeting a Condition.....	44
6.4 SEQUENCE — Generate Number Series.....	44
Section 7 — Financial Functions	45
7.1 PMT — Calculate Loan EMI	45
7.2 PV — Present Value (What is a Future Amount Worth Today?)	45
7.3 FV — Future Value (How Much Will Savings Grow?).....	46
7.4 NPV — Net Present Value (Is This Investment Worth It?)	46
7.5 IRR — Internal Rate of Return.....	47
Section 8 — Error Handling Functions	48
8.1 IFERROR — Catch ANY Error	48
8.2 IFNA — Catch Only #N/A Errors.....	49
8.3 ISERROR / ISERR / ISNA — Test for Errors.....	49
8.4 IS Functions — Data Validation.....	49
Master Quick Reference — All Functions	51
Chapter 3 : DATA CLEANING	54
3.1 — Remove Duplicates.....	54
3.2 — Text to Columns.....	55
3.3 — Flash Fill (Advanced Data Cleaning).....	56
3.4 — TRIM & CLEAN Functions.....	57

3.5 — Find & Replace.....	59
3.6 — Data Validation.....	61
3.7 — Conditional Formatting	62
3.8 — Case Studies & Assignments.....	64
PRACTICE DATASETS — FILLED EXAMPLES	67
Chapter 4 : PIVOT TABLE.....	68
4.1 — Pivot Tables	68
4.2 — Pivot Charts	70
4.3 — Grouping.....	71
4.4 — Slicers.....	72
4.5 — Timelines	73
4.6 — Complete PivotTable Workflow	73
4.7 — Practice Dataset.....	74
Chapter 5 : DASHBOARD.....	75
5.1 — KPI Cards.....	75
5.2 — Interactive Dashboards	76
5.4 — Layout Design	77
5.5 — Dynamic Dashboards.....	78
5.6 — Project Work / Assignments.....	78
Chapter 6 : DATA ANALYSIS	79
6.1 — Comparison Analysis	79
6.2 — Trend Analysis	79
6.3 — Percentage Analysis.....	80
6.4 — Variance Analysis.....	80
6.5 — Top/Bottom Analysis	81
6.6 — Descriptive Statistics.....	81
6.7 — Case Study & Assignment.....	82
Chapter 7 : FORECASTING.....	82
7.1 — Forecast Sheet.....	83
7.2 — FORECAST Function	84
7.3 — FORECAST.LINEAR	84
7.4 — TREND.....	85
7.5 — Moving Average.....	85
7.6 — Sales Forecasting	86

7.7 — Project Work / Assignments	87
Chapter 8 : FORMS IN EXCEL.....	88
8.1 — Data Form	88
8.2 — Developer Tab	88
8.3 — Form Controls.....	89
8.4 — Check Box	90
8.5 — Option Button.....	90
8.7 — Combo Box	91
8.8 — Spin Button	92
8.9 — Scroll Bar.....	92
8.10 — Interactive Forms.....	93
8.11 — Case Studies / Assignments.....	93
Chapter 9 : MACROS	95
9.1. Prerequisites for Using Excel Macros.....	95
9.2. Enabling Macros on Windows	96
9.3. Enabling Macros on Mac.....	96
9.4. How to Record a Macro in Excel	96
9.5. How to Run the Macro in Excel.....	97
9.6. How to Run Macro By Clicking a Button	100
9.7. How to Create Macro in VBA Editor	102
Chapter 10 : What If Analysis.....	107
10.1 — Goal Seek	107
10.2 — Scenario Manager.....	108
10.3 — One-Variable Data Table	108
10.4 — Two-Variable Data Table	109
10.5 — Sensitivity Analysis	109
10.6 — What-If Analysis Comparison	109
10.7 — Case Studies / Assignments.....	110
Case Study 4 — Pricing Sensitivity	110



CHAPTER 1 — EXCEL BASICS

Foundation skills for every Excel user in business

1.1 — Introduction to Excel & Business Applications

What Is Microsoft Excel?

Microsoft Excel is a spreadsheet application that organises data into rows and columns inside a grid called a worksheet. It is part of the Microsoft Office suite and is the most widely used business software in the world — used daily by accountants, analysts, HR professionals, sales managers, and students.

Why Excel in Business?

Business Area	How Excel Is Used
Finance & Accounts	Budgeting, P&L statements, expense tracking, loan EMI calculators
Sales & Marketing	Sales reports, target vs achievement tracking, customer lists
HR	Payroll calculations, attendance tracking, employee master data
Operations	Inventory management, production schedules, logistics tracking
Data Analysis	Pivot tables, dashboards, trend analysis, forecasting
Education	Gradebooks, mark sheets, result analysis, fee tracking
Banking & Finance	Loan amortisation, investment calculators, reconciliation

Excel File Formats

Extension	Type	When to Use
.xlsx	Standard Excel Workbook	Default — use for all regular work
.xlsm	Macro-Enabled Workbook	When the file contains VBA macros
.csv	Comma-Separated Values	Sharing plain data with other systems (no formatting)
.pdf	PDF Export	Final reports shared with clients — not editable

Industry Use Cases

- **MNC Finance Teams:** Build monthly MIS reports in Excel shared across departments.
- **Startups:** Track revenue, costs, and runway in an Excel financial model.
- **HR Departments:** Maintain employee master data with salary and attendance.
- **Freelancers:** Create invoices, expense sheets, and income statements.

1.2 — Excel Interface

Theory

The Excel interface is the visual layout you see when you open Excel. Knowing where each element is located saves time and makes you faster at navigating the application.

Key Interface Elements

Element	Location	Purpose
Title Bar	Very top	Shows the filename and application name
Ribbon	Top strip with tabs	All commands organised into tabs (Home, Insert, Data, etc.)
Quick Access Toolbar	Top-left, above Ribbon	Pinned shortcuts: Save, Undo, Redo — customisable
Name Box	Left of formula bar	Shows the active cell address (e.g. A1) — also for navigation
Formula Bar	Below Ribbon	Shows and edits the content / formula of the active cell
Column Headers	Lettered A, B, C...	Click to select the entire column
Row Headers	Numbered 1, 2, 3...	Click to select the entire row
Cell	Intersection of row & column	The basic unit of data entry
Active Cell	Highlighted with green border	The cell currently selected for input
Sheet Tabs	Bottom of window	Navigate between worksheets in the workbook
Status Bar	Very bottom	Shows Sum, Average, Count of selected cells instantly
Scroll Bars	Right and bottom edges	Navigate large worksheets
View Buttons	Bottom-right	Normal, Page Layout, Page Break Preview views
Zoom Slider	Bottom-right corner	Adjust the display zoom level (50% – 400%)

Ribbon Tabs — What Each Contains

Tab	Key Contents
Home	Clipboard, Font, Alignment, Number format, Styles, Cells, Editing (most used tab)
Insert	Tables, Charts, PivotTable, Pictures, Shapes, Sparklines, Links
Page Layout	Margins, Orientation, Print Area, Scale, Gridlines, Headers/Footers
Formulas	Function Library, Name Manager, Formula Auditing (Trace, Watch, Evaluate)
Data	Get Data, Sort, Filter, Text to Columns, Remove Duplicates, Data Validation
Review	Spelling, Comments, Track Changes, Protect Sheet/Workbook
View	Freeze Panes, Split, Zoom, Macros, New Window, Arrange All

Essential Navigation Shortcuts

Ctrl + Home	→ Go to cell A1 (start of sheet)
Ctrl + End	→ Go to last used cell in sheet
Ctrl + Arrow Key	→ Jump to last filled cell in direction
Ctrl + G or F5	→ Go To – jump to any cell address
Ctrl + F	→ Find (Ctrl + H = Find & Replace)
Ctrl + Page Down/Up	→ Switch between worksheets
F2	→ Edit the active cell
Escape	→ Cancel editing / exit a dialog
Alt + Tab	→ Switch between open Excel windows

1.3 — Workbook & Worksheet Management

Theory

Workbook: The entire Excel file (.xlsx). One workbook can contain multiple worksheets.

Worksheet: A single sheet (tab) inside the workbook. A worksheet contains up to 1,048,576 rows and 16,384 columns (A to XFD).

Workbook Operations

Action	How to Do It	Keyboard Shortcut
Create New Workbook	File → New → Blank Workbook	Ctrl + N
Open Existing File	File → Open → Browse	Ctrl + O
Save Workbook	File → Save	Ctrl + S
Save As (new name)	File → Save As → Choose location	F12
Close Workbook	File → Close or click X on tab	Ctrl + W
Print	File → Print	Ctrl + P
Undo / Redo	Quick Access Toolbar arrows	Ctrl+Z / Ctrl+Y

Worksheet (Tab) Operations

INSERT new sheet	→ Right-click tab → Insert OR click ⊕ icon next to tabs
RENAME sheet	→ Double-click the tab name → type new name → Enter
DELETE sheet	→ Right-click tab → Delete (WARNING: cannot be undone!)
MOVE / COPY sheet	→ Right-click tab → Move or Copy → choose destination
CHANGE TAB COLOUR	→ Right-click tab → Tab Color → pick colour
HIDE / UNHIDE sheet	→ Right-click tab → Hide / Unhide
SELECT ALL SHEETS	→ Right-click any tab → Select All Sheets (group mode)

NOTE Deleting a worksheet is permanent — there is NO undo. Always check before deleting. Better to hide it if unsure.

Best Practices for Workbook Organisation

- **Naming:** Use clear, short tab names. E.g. "Jan_Sales", "EmpData", "Summary" — not "Sheet1".
- **Colour coding:** Use consistent tab colours — e.g. green for data sheets, orange for summary sheets.
- **Order:** Keep raw data on the left, summaries and dashboards on the right.
- **File naming:** Include version and date in filename: "SalesReport_v2_Jan2024.xlsx".

1.4 — Data Entry Techniques

Theory

Efficient data entry is the foundation of all Excel work. Poor data entry leads to errors in formulas, lookups, and reports. Knowing the correct techniques saves hours of data cleaning later.

Basic Data Entry Rules

Data Type	How Excel Treats It	Alignment	Example
Numbers	Numeric — can calculate	Right-aligned	45000, 3.14, -500
Text	String — cannot calculate	Left-aligned	"Aarav", "Sales", "Mumbai"
Dates	Serial number internally	Right-aligned	01-Jan-2024
Formulas	Starts with =	Depends on result	=A1+B1
Boolean	TRUE or FALSE	Centered	TRUE, FALSE

Smart Data Entry Tips

Tip 1 — Enter data in multiple cells at once

Select a range (e.g. B2:B10), type the value, press Ctrl + Enter — fills all selected cells with the same value.

Tip 2 — Move after entering data

```
Enter      → move DOWN after entry
Tab        → move RIGHT after entry
Shift+Enter → move UP
Shift+Tab  → move LEFT
```

Tip 3 — Drop-down using existing list values

Press Alt + ↓ while in a cell to show a dropdown of values already typed in that column — great for consistent data entry.

Tip 4 — Numbers stored as text (common data quality issue)

INFO If numbers align LEFT instead of RIGHT, they are stored as text. Excel cannot calculate with them. Fix: select the cells → Data → Text to Columns → Finish, OR multiply by 1: =A1*1

Selecting Ranges Efficiently

Click + Drag	→ select a rectangular range
Ctrl + Click	→ select non-adjacent cells
Shift + Click	→ select from active cell to clicked cell
Ctrl + Shift + End	→ select from active cell to last used cell
Ctrl + Shift + Arrow	→ select to last filled cell in direction
Ctrl + A	→ select entire used range (press again for all cells)
Click Row/Column header	→ select entire row or column

Industry Use Cases

- **HR:** Entering attendance data for 200+ employees — use Tab to move right across columns quickly.
- **Accounts:** Enter 50 transaction amounts — select all 50 cells first, type, Ctrl+Enter fills all.
- **Data import:** When CSV data imports with numbers as text, use Text to Columns to fix all at once.

1.5 — Cell Referencing (Relative, Absolute & Mixed)

Theory

Cell referencing determines how a formula's cell addresses behave when the formula is copied or moved to another cell. This is one of the most important concepts in Excel — getting it wrong causes silent formula errors across entire sheets.

Type	Notation	Behaviour When Copied	Use When...
Relative	A1	Row AND column both change	Formula logic should shift with position
Absolute	\$A\$1	Row AND column stay FIXED	Referring to one fixed cell (tax rate, exchange rate)
Mixed — Fix Column	\$A1	Column fixed, row changes	Reference stays in same column across rows
Mixed — Fix Row	A\$1	Row fixed, column changes	Reference stays in same row across columns

TIP Press F4 while typing a cell reference in the formula bar to cycle through all four reference types: A1 → \$A\$1 → A\$1 → \$A1 → A1

Worked Examples

Setup — Sample Data

Row	A	B	C
1	Qty	Unit Price	Total
2	10	500	=A2*B2
3	25	300	=A3*B3
4	8	750	=A4*B4

Example 1 — Relative Reference (default)

Formula in C2: **=A2*B2**

When you copy C2 down to C3, it automatically becomes =A3*B3. This is correct — each row calculates its own total.

```
C2: =A2*B2    → copied to C3 becomes → =A3*B3  ✓  
C3: =A3*B3    → copied to C4 becomes → =A4*B4  ✓  
Row numbers shift automatically — this is CORRECT behaviour here.
```

Example 2 — Absolute Reference (tax rate on a separate row)

Tax rate is in cell **E1** = 18%. Every product total needs to add 18% GST.

```
D2: =C2*(1+$E$1)  ← $E$1 is LOCKED  
  
When copied to D3: =C3*(1+$E$1)  ← $E$1 stays fixed ✓  
When copied to D4: =C4*(1+$E$1)  ← $E$1 stays fixed ✓  
  
WITHOUT $:  D3 would become =C3*(1+E2) — wrong cell! ✗
```

Example 3 — Mixed Reference (multiplication table)

Build a 5×5 multiplication table where row 1 has numbers 1–5 and column A has numbers 1–5:

```
Formula in B2: =B$1 * $A2  
  
B$1 → row 1 is FIXED (so we always pick the column header)  
$A2 → column A is FIXED (so we always pick the row header)  
  
Copy this formula across the entire 5×5 grid — it works perfectly!
```

Common Mistakes

Mistake	Result	Fix
Copying formula without \$ on tax rate cell	Each row picks a different tax rate	Lock with \$E\$1
Using \$ on both cells when copying down	Every row shows same answer	Use relative for the row that should change
Not using \$ in VLOOKUP table_array	Table reference shifts down when formula	Always lock VLOOKUP table: \$A\$2:\$D\$100

	is copied	
--	-----------	--

Industry Use Cases

- **Finance:** One exchange rate cell (\$B\$1) referenced absolutely in every currency conversion formula.
- **Sales:** Commission % in a fixed cell (\$C\$1), multiplied by each sales rep's revenue using relative row reference.
- **Pricing:** Mixed reference to build price matrix — rows = products, columns = discount slabs.

1.6 — Formatting & Number Formats

Theory

Formatting controls how data LOOKS without changing the underlying value. A salary of 45000 can be displayed as ₹45,000.00 — the number stored is still 45000. Good formatting makes reports readable and professional.

Font & Cell Formatting — Home Tab

Format Option	What It Does	Where to Find
Bold / Italic / Underline	Emphasise text	Home → Font group Ctrl+B / Ctrl+I / Ctrl+U
Font & Font Size	Change typeface and size	Home → Font group
Font Color	Change text color	Home → Font → A with color strip
Fill Color	Background color of cell	Home → Font → Paint bucket icon
Borders	Lines around cells	Home → Font → Border dropdown
Alignment	Left / Centre / Right / Wrap Text	Home → Alignment group
Merge & Centre	Combine cells into one	Home → Alignment → Merge & Centre
Wrap Text	Show full text in the cell	Home → Alignment → Wrap Text
Indent	Add space inside cell from left	Home → Alignment → Indent buttons

Number Formats — Critical for Business Reports

Number Format	How It Appears	When to Use
General	45000	Default — no formatting
Number	45,000.00	Plain numbers with decimal places
Currency	₹45,000.00	Financial amounts with currency symbol
Accounting	₹ 45,000.00	Financial statements — symbol left-aligned
Percentage	18.00%	Rates, growth %, margins (stores 0.18, shows 18%)
Date	01-Jan-2024	Dates — choose your preferred format

Time	09:30 AM	Clock time values
Text	"45000"	Forces number to be stored as text (use rarely)
Custom	₹#,##0;(₹#,##0)	Negatives in brackets — standard finance format

Apply Number Formats

Ctrl + 1 → Open Format Cells dialog (most powerful)
Ctrl + Shift + \$ → Currency format
Ctrl + Shift + % → Percentage format
Ctrl + Shift + # → Date format (DD-MMM-YY)
Ctrl + Shift + ! → Number with comma separator

Home → Number group dropdown → choose any built-in format
Format Cells → Number tab → Custom → type your own format code

Custom Number Format Examples

Format Code	Value Stored	Displayed As	Use Case
₹#,##0	45000	₹45,000	Indian currency without decimals
₹#,##0.00	45000	₹45,000.00	Indian currency with 2 decimals
0.0%	0.085	8.5%	Interest rates
DD-MMM-YYYY	45292	01-Jan-2024	Readable date format
0.0"x"	3.5	3.5x	Valuation multiples
[>0]"▲ "0;[<0]"▼ "0;"-	500	▲ 500	Growth indicators
"INV-"000	42	INV-042	Invoice / reference numbers

Industry Use Cases

- **Finance Reports:** Negative numbers shown in red (▼) or brackets — configure in Format Cells → Custom.
- **HR Payroll:** Salary column formatted as ₹#,##0 — makes large numbers readable instantly.
- **Sales Dashboards:** Growth % formatted as 0.0% with conditional colors — green for positive, red for negative.
- **Invoices:** "INV-"000 format generates INV-001, INV-002 automatically as numbers are entered.

1.7 — AutoFill & Flash Fill

AutoFill — Theory

AutoFill detects patterns in your data and extends them automatically. It works by recognising sequences of numbers, dates, days, months, and custom lists. You use the small green square at the bottom-right corner of a selected cell — called the Fill Handle.

AutoFill Examples

You Type (first 1–2 cells)	AutoFill Continues With...	Type
1, 2	3, 4, 5, 6 ...	Number series
1, 3	5, 7, 9, 11 ... (detects step of 2)	Custom step series
Jan	Feb, Mar, Apr ... Dec	Months
January	February, March ... December	Full months
Mon	Tue, Wed, Thu ... Sun	Weekdays
Monday	Tuesday, Wednesday ...	Full weekdays
01-Jan-2024	02-Jan-2024, 03-Jan-2024 ...	Date series
Q1	Q2, Q3, Q4	Quarter labels
=A1*2	=A2*2, =A3*2 ... (formula copies)	Formulas
EMP-001	EMP-002, EMP-003 ...	Text + number

How to Use AutoFill

- STEP 1 Enter starting values — Type the first 1 or 2 values in adjacent cells
- STEP 2 Select those cells — Highlight all the starting value cells
- STEP 3 Grab the fill handle — Hover over bottom-right corner — cursor changes to a thin + sign
- STEP 4 Drag in direction — Drag down, up, left, or right to extend the series

TIP Double-click the Fill Handle to instantly fill down to the last row of data in the adjacent column — no dragging needed!

Flash Fill — Theory

Flash Fill (introduced in Excel 2013) is a smart pattern recognition tool. You manually transform one or two examples, and Flash Fill recognises the pattern and completes the rest automatically — without any formula.

INFO Flash Fill works best for text manipulation: extracting names, combining fields, reformatting codes, changing capitalisation. It is an instant alternative to complex text formulas.

Flash Fill Examples

Column A (Original)	Column B (Your Example)	Flash Fill Fills...	What It Detected
Aarav Shah	Aarav	Sneha, Rohit, Priya...	Extract first name
Aarav Shah	Shah	Mehta, Verma, Joshi...	Extract last name
aarav shah	Aarav Shah	Sneha Mehta, Rohit Verma...	Proper case
9876543210	98765-43210	99887-76655, 91234-56789...	Format phone number
E001 - Aarav Shah	E001	E002, E003, E004...	Extract ID code

01/15/2024	15-Jan-2024	22-Jan-2024, 03-Feb-2024...	Reformat date
Aarav Sales Pune	Aarav	Sneha, Rohit, Priya...	Extract before delimiter

How to Use Flash Fill

STEP 1 Type your example — In the adjacent column, type exactly what you want for the first row

STEP 2 Press Enter — Move to the next row in the same column

STEP 3 Trigger Flash Fill — Press Ctrl + E — OR — Data tab → Flash Fill

STEP 4 Review the results — Check a few rows to confirm Flash Fill detected the right pattern

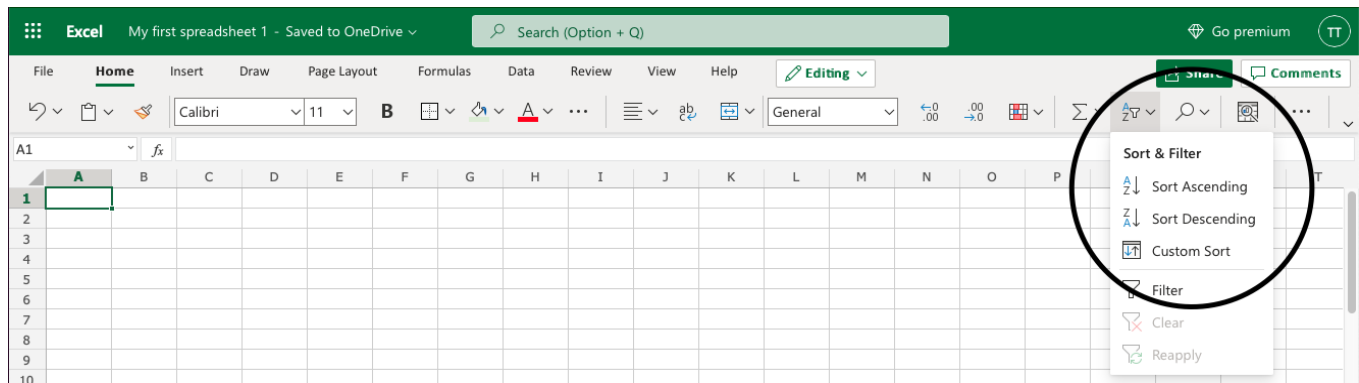
Industry Use Cases

- **HR Data Cleaning:** Names imported as "AARAV SHAH" → Flash Fill to convert to "Aarav Shah" in one step.
- **Phone Numbers:** Database exports numbers as 10 digits → Flash Fill to format as 98765-43210.
- **Report Dates:** System exports dates as YYYYMMDD → Flash Fill reformats to DD-MMM-YYYY for reports.
- **Email Generation:** Flash Fill detects "aarav.shah@company.com" pattern and creates emails for all employees.

1.8 — Sorting & Filtering

Sorting — Theory

Sorting rearranges rows in your data based on the values in one or more columns. It is a non-destructive operation — you can always re-sort or undo.



Quick Sort (Single Column)

STEP 1 Click any cell in the column — Do not select the entire column — just one cell

STEP 2 Go to Data tab — Data → Sort & Filter group

STEP 3 Click A→Z or Z→A — A→Z = ascending (small to large, A to Z) | Z→A = descending

SHORTCUT Keyboard: Alt + A + S + A (ascending) | Alt + A + S + D (descending)

Multi-Level Sort (Sort by multiple columns)

Example: Sort by Department (A→Z), and within each department, sort by Salary (largest first).

STEP 1 Click any cell in data — Do NOT select only one column — Excel needs to see all data

STEP 2 Data → Sort — Opens the Sort dialog box

STEP 3 Add first sort level — Column = dept, Order = A to Z

STEP 4 Click "Add Level" — Add second criterion

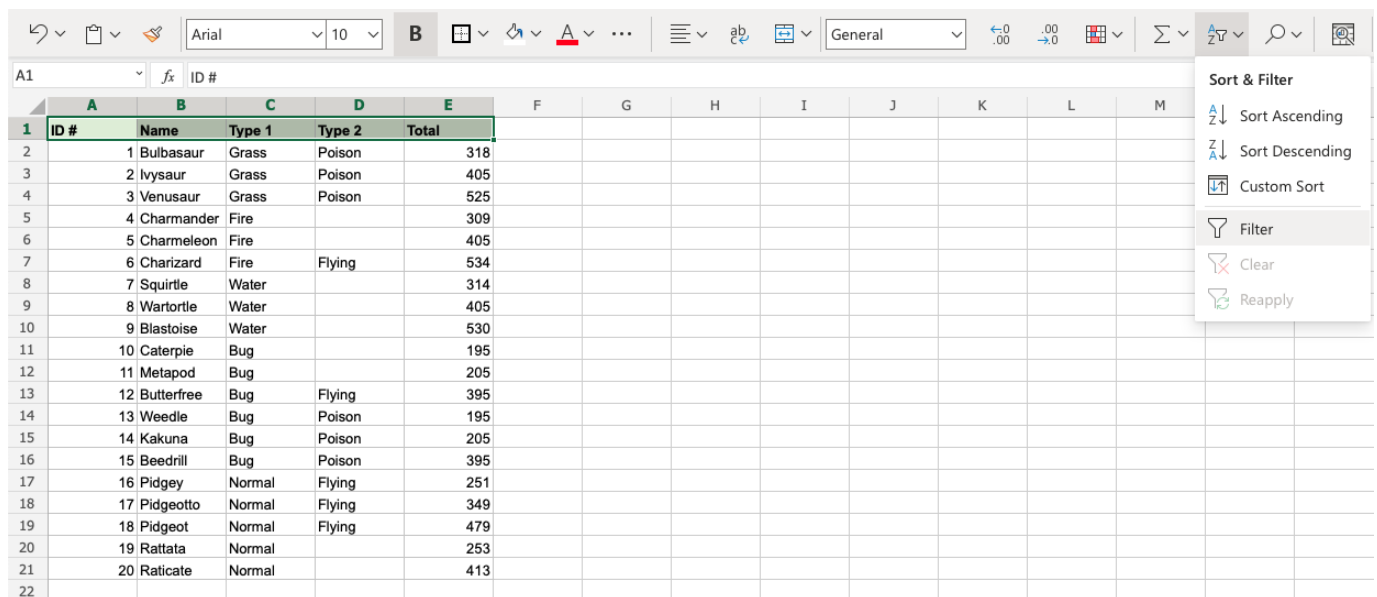
STEP 5 Add second sort level — Column = salary, Order = Largest to Smallest

STEP 6 Click OK — Data is sorted by dept first, then by salary within each dept

NOTE Always ensure your data has a single header row before sorting. If you have merged header cells, sorting will not work correctly.

Filtering — Theory

Filtering hides rows that do not match your criteria. Unlike deletion, filtered rows are still in the worksheet — they are just temporarily hidden. Filtering does not change the data.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	ID #	Name	Type 1	Type 2	Total								
2	1	Bulbasaur	Grass	Poison	318								
3	2	Ivysaur	Grass	Poison	405								
4	3	Venusaur	Grass	Poison	525								
5	4	Charmander	Fire		309								
6	5	Charmeleon	Fire		405								
7	6	Charizard	Fire	Flying	534								
8	7	Squirtle	Water		314								
9	8	Wartortle	Water		405								
10	9	Blastoise	Water		530								
11	10	Caterpie	Bug		195								
12	11	Metapod	Bug		205								
13	12	Butterfree	Bug	Flying	395								
14	13	Weedle	Bug	Poison	195								
15	14	Kakuna	Bug	Poison	205								
16	15	Beedrill	Bug	Poison	395								
17	16	Pidgey	Normal	Flying	251								
18	17	Pidgeotto	Normal	Flying	349								
19	18	Pidgeot	Normal	Flying	479								
20	19	Rattata	Normal		253								
21	20	Raticate	Normal		413								
22													

AutoFilter

STEP 1 Click any cell in the header row — Or select the entire header row

STEP 2 Data → Filter (or Ctrl + Shift + L) — Dropdown arrows appear on each column header

STEP 3 Click the arrow on any column — A filter panel opens with options

STEP 4 Choose filter criteria — Tick/untick values, or use "Text Filters" / "Number Filters"

STEP 5 Click OK — Non-matching rows are hidden. Row numbers turn BLUE to indicate filtering.

Filter Types Available

Filter Type	When to Use	Example
Check/Uncheck values	Pick specific values from a list	Show only "Sales" and "IT" departments
Text Filters	Contains, begins with, ends with	"Contains" → "Shah" → shows all Shahs
Number Filters	Greater than, between, top 10	Salary > 50000, or Top 5 salaries
Date Filters	This month, last week, between	Join date in 2023, or between Jan–Mar 2024
Search Box	Type to filter the list instantly	Type "Pune" to show only Pune entries

Clear Filters

Clear one column filter → click the column arrow → "Clear Filter From [column]"
Clear ALL filters → Data → Clear (in Sort & Filter group)
Turn off AutoFilter → Data → Filter (toggle off) OR Ctrl + Shift + L

Industry Use Cases

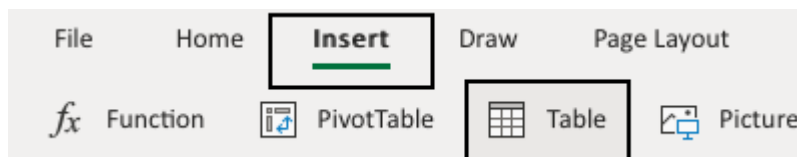
- **HR:** Sort employee list by department then name — clean alphabetical roster per team.
- **Sales:** Filter transactions for "West" region in "March 2024" to build a monthly regional report.
- **Finance:** Filter invoices with status "Unpaid" and amount > ₹50,000 for collections followup.
- **Inventory:** Filter items where stock < reorder level → highlight and send purchase order.

1.9 — Excel Tables

Theory

An Excel Table (created with Ctrl+T) is a structured range with special features built in: automatic formatting, auto-expanding formulas, structured references, and built-in filtering. Tables are the professional way to manage any dataset in Excel.

2. Click **Insert**, then Table ( Table), in the Ribbon.



3. Click OK

Create Table ✕

The data for your table is in the following range: **A1:F22**

☒ My table has headers

OK

Cancel

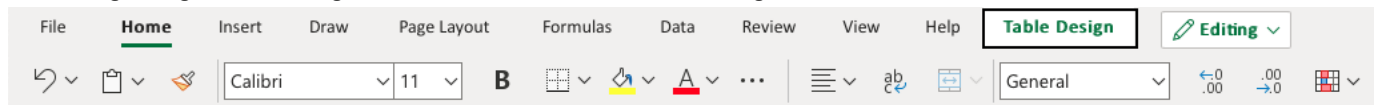
How to Create a Table

- STEP 1** Click any cell in your data — Make sure data has one header row and no blank rows
- STEP 2** Press **Ctrl + T** — Or go to Insert → Table
- STEP 3** Confirm the range — "My table has headers" should be checked
- STEP 4** Click **OK** — Table is created with striped formatting and filter arrows
- STEP 5** Name your table — Table Design tab → Table Name → type a meaningful name (e.g. EmpTable)

How to Design Table

Tables can be customized and styled in a few clicks.

Converting a range into a table gives access to a menu called "Table Design"



The menu appears when selecting a cell in the table's range.

This menu has options and commands such as:

- Resize
- Remove duplicates
- Convert to range
- Style options (Total row, Header row, Banded row etc..)
- Formatting

Table Name

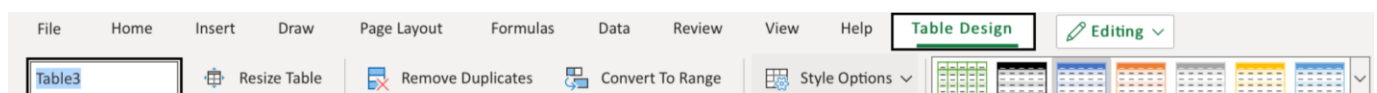
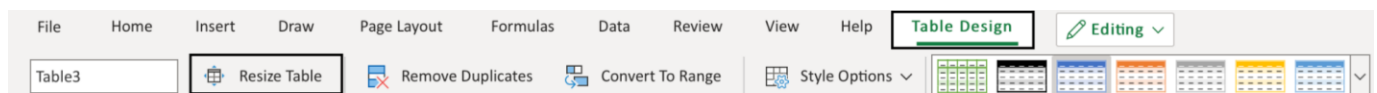


Table Resize



Remove Duplicates

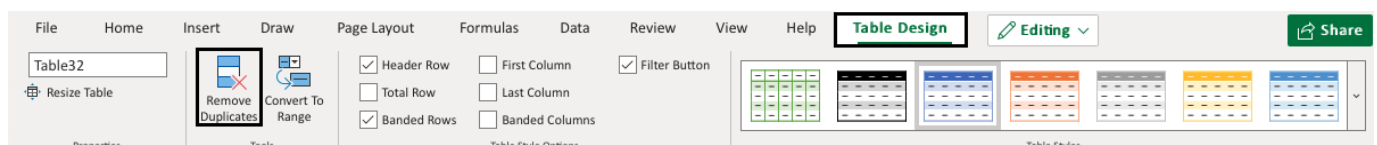


Table Advantages vs Plain Range

Feature	Plain Range	Excel Table
AutoFilter	Must add manually	Built-in automatically
Formula copies	Must drag or copy	Auto-fills when you type in any column
New row references	Must update ranges manually	Range expands automatically
Structured references	Not available	=EmpTable[salary] instead of =\$E\$2:=\$E\$11
Total Row	Must write SUM manually	Toggle with Ctrl+Shift+T
Formatting	Must apply manually	Consistent banded rows automatic
Chart updates	Chart stays static	Chart auto-updates when table grows

Table Features to Know

Structured References

Once a table is named "EmpTable", you can reference columns by name:

=SUM(EmpTable[salary])	→ sum of salary column
=AVERAGE(EmpTable[salary])	→ average salary
=COUNTIF(EmpTable[dept], "Sales")	→ count Sales employees
=EmpTable[@salary]*10%	→ current row salary × 10% (in a table column)

Total Row

Table Design tab → check "Total Row"
OR press Ctrl + Shift + T

Click any cell in the Total Row to choose: Sum, Average, Count, Max, Min...

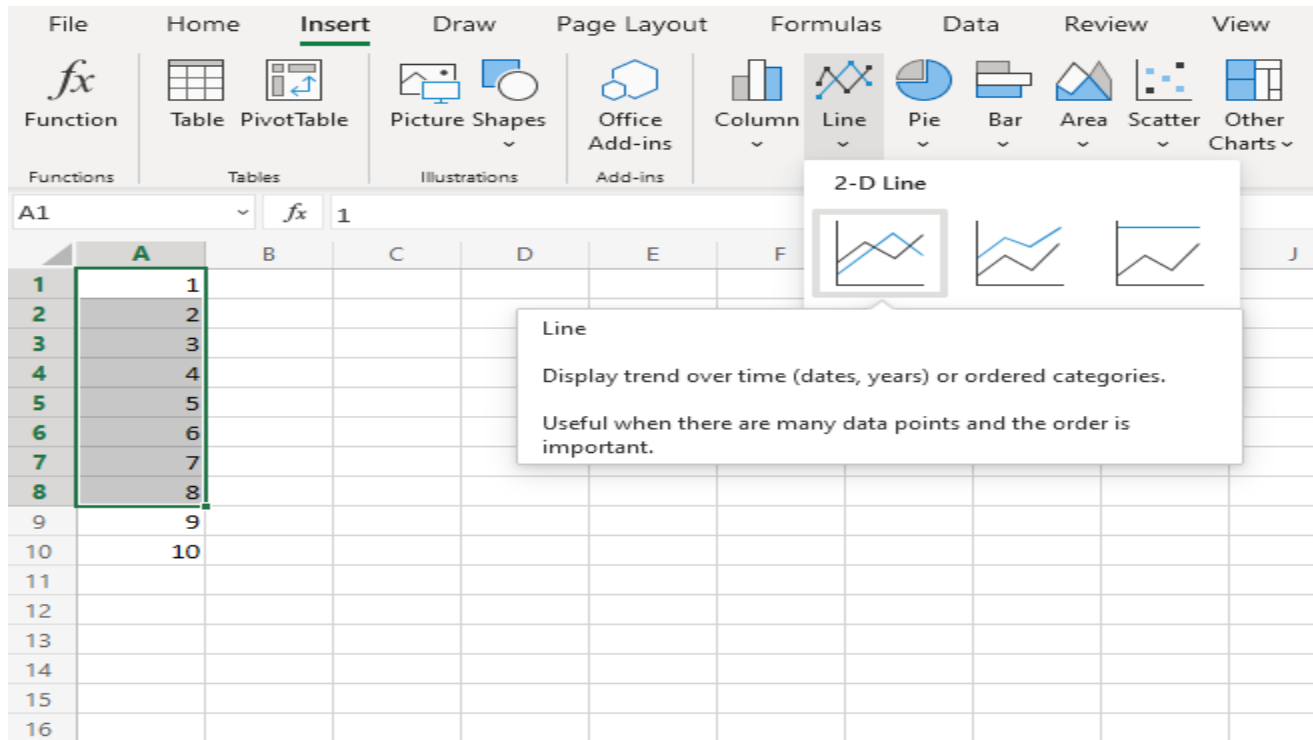
Industry Use Cases

- **HR Databases:** Employee master as a table — adding a new employee row auto-extends all formulas and filters.
- **Sales Tracking:** Monthly transaction table auto-expands — SUMIFS and PivotTables always include new data.
- **Financial Models:** Structured references make formulas self-documenting: =SUM(Revenue[Q1]) is clearer than =SUM(\$B\$2:\$B\$50).

1.10 — Basic Charts

Theory

Charts convert rows and columns of numbers into visual stories. A well-chosen chart makes trends, comparisons, and compositions immediately obvious — something rows of numbers cannot do alone.



Choosing the Right Chart

Chart Type	Best Used For	Example in Business
Column Chart	Comparing values across categories	Sales by department, month-by-month revenue
Bar Chart	Same as column but horizontal (long labels)	Top 10 customers by revenue
Line Chart	Trends over time	Monthly sales trend over a year
Pie Chart	Parts of a whole (max 5–6 slices)	Market share, expense breakdown
Doughnut Chart	Parts of a whole with a hole in centre	Budget allocation
Area Chart	Cumulative trend over time	Running total of sales
Scatter Chart	Relationship between two numeric variables	Salary vs experience, price vs demand
Combo Chart	Two different data series with different scale	Revenue (bars) + Profit % (line)

How to Create a Chart

- STEP 1
- Select your data — Include column headers. Use Ctrl+Click for non-adjacent columns.
- STEP 2
- Insert → Charts group — Choose chart type from the ribbon, OR press Alt + F1 for instant default chart
- STEP 3
- Move and resize — Drag to position. Drag corners to resize.
- STEP 4
- Add Chart Title — Click the chart → Chart Elements (+) → Chart Title
- STEP 5
- Format elements — Right-click any element (bar, axis, legend) → Format to customise

SHORTCUT

Alt + F1 → creates an embedded chart instantly from selected data. F11 → creates a chart on a separate sheet.

Chart Elements to Know

Element	Purpose	How to Access
Chart Title	Describe what the chart shows	Click the title text to edit
Axis Titles	Label the X and Y axes	Chart Elements (+) → Axis Titles
Data Labels	Show exact values on bars/lines	Chart Elements → Data Labels
Legend	Identify each data series by colour	Chart Elements → Legend
Gridlines	Visual guides for reading values	Chart Elements → Gridlines
Data Table	Show underlying data below chart	Chart Elements → Data Table

Industry Use Cases

- Management Dashboards:** Column chart showing monthly revenue vs target — updated automatically from table data.
- Finance:** Line chart showing 12-month expense trend — peaks and dips instantly visible.
- Sales:** Pie chart showing each region's share of total revenue for a quarterly review.
- HR:** Bar chart of headcount by department — horizontal makes long department names readable.

1.11 — Freeze Panes & Split

Freeze Panes — Theory

When you scroll down or right in a large sheet, your column or row headers disappear from view. Freeze Panes locks headers so they stay visible no matter how far you scroll.

Freeze Options

Option	What It Freezes	When to Use
Freeze Top Row	Row 1 stays visible while scrolling down	Simple tables with a single header row
Freeze First Column	Column A stays visible while scrolling right	Wide tables where ID/Name must always

		show
Freeze Panes	Rows above AND columns left of selected cell	Both headers and row labels needed simultaneously

How to Freeze

- STEP 1** Click the cell below and to the right of what you want frozen — E.g. click B2 to freeze Row 1 and Column A together
- STEP 2** View → Freeze Panes → Freeze Panes — A dark line appears marking the frozen boundary
- STEP 3** To unfreeze — View → Freeze Panes → Unfreeze Panes

NOTE Freeze Panes is a display setting only — it does not affect printing. Use Page Setup → Print Titles to repeat headers when printing.

Split — Theory

Split divides the worksheet window into 2 or 4 independent panes. You can scroll each pane separately to compare distant parts of the same sheet side by side.

How to Split

View → Split → splits at the active cell position
 Drag the split bar (thick line at top of scroll bar) to split horizontally
 Double-click the split bar to remove a split

Industry Use Cases

- Large Data:** Freeze the employee ID and name columns while scrolling right through 20+ columns of monthly attendance.
- Financial Models:** Freeze years 2020–2024 on the left, scroll right to see 2025–2030 forecast columns.
- Split for Comparison:** Split screen to compare January data (top pane) with December data (bottom pane).

1.12 — Named Ranges

Theory

A Named Range assigns a memorable name to a cell or range of cells. Instead of using cell addresses like `$E$2:$E$11`, you can write `=SUM(Salary)` — making formulas self-explanatory, easier to audit, and less error-prone.

How to Create a Named Range

Method 1 — Name Box (fastest)

- STEP 1** Select the cells to name — E.g. select E2:E11 (salary column)

STEP 2 Click the Name Box — The box showing the cell address, top-left of formula bar

STEP 3 Type the name and press Enter — Type "Salary" → Enter. Done.

Method 2 — Define Name (more options)

STEP 1 Formulas → Define Name — Opens the New Name dialog

STEP 2 Type the name — E.g. TaxRate

STEP 3 Set the Scope — Workbook (all sheets) or a specific sheet

STEP 4 Check the Refers to range — Confirm the correct cells are selected

STEP 5 Click OK — Name is created and available in the Name Box dropdown

Naming Rules

Rule	Example
No spaces — use underscore	"Tax_Rate" not "Tax Rate"
Cannot start with a number	"Q1_Revenue" not "1Q_Revenue"
No special characters except _ ?	"Salary_2024" OK "Salary-2024" NOT OK
Not the same as a cell address	"C2" is not valid — conflicts with cell C2
Max 255 characters	Keep names short and meaningful

Using Named Ranges in Formulas

Instead of: `=SUMIF($C$2:$C$11,"Sales",$E$2:$E$11)`
Write: `=SUMIF(Dept,"Sales",Salary)`

Instead of: `=VLOOKUP(H2,$A$2:$G$11,5,FALSE)`
Write: `=VLOOKUP(H2,EmpData,5,FALSE)`

Instead of: `=A2*$F$1`
Write: `=A2*TaxRate`

Manage Named Ranges

Formulas → Name Manager (Ctrl + F3) → view, edit, delete all names
F3 while in formula bar → paste a name into a formula
Formulas → Create from Selection → auto-name ranges using header row/column values

Industry Use Cases

- **Financial Models:** "GrowthRate", "Discount", "TaxRate" as named cells — update once, reflects everywhere.
- **Data Validation:** Named range "DeptList" used as the source for a dropdown validation list.
- **VLOOKUP:** "EmpMaster" names the employee table — all lookups reference this name, not a cell address.

1.13 — Page Setup & Printing

Theory

Page Setup controls how your worksheet looks when printed or exported as a PDF. Most Excel users never configure this and end up with poorly formatted printouts — cut-off columns, missing headers, wrong orientation.

Page Setup Options

Setting	Purpose	Where to Find
Orientation	Portrait (tall) or Landscape (wide)	Page Layout → Orientation
Paper Size	A4, Letter, Legal etc.	Page Layout → Size
Margins	Space on each side of the page	Page Layout → Margins
Scale to Fit	Shrink/expand to fit on 1 page or N pages	Page Layout → Scale to Fit
Print Area	Print only selected cells, not entire sheet	Page Layout → Print Area → Set
Print Titles	Repeat header row on every printed page	Page Layout → Print Titles
Gridlines	Print cell borders even without formatting	Page Layout → Sheet Options
Headers & Footers	Page numbers, filename, date on every page	Insert → Header & Footer
Page Breaks	Control where pages break	View → Page Break Preview

Step-by-Step — Preparing to Print

- STEP 1 Set Print Area** — Select only the cells you want printed → Page Layout → Set Print Area
- STEP 2 Choose Orientation** — Wide tables → Landscape. Tall tables → Portrait.
- STEP 3 Set Repeat Rows** — Page Layout → Print Titles → Rows to repeat at top → select header row
- STEP 4 Fit to page** — Page Layout → Scale to Fit → set Width: 1 page
- STEP 5 Preview** — Ctrl + P → see the print preview before printing
- STEP 6 Print or Export PDF** — Print directly, or File → Export → Create PDF/XPS

SHORTCUT Ctrl + P → Print preview and print dialog | Ctrl + F2 → Quick print preview

Industry Use Cases

- Finance:** P&L statement printed on one landscape A4 page — use Scale to Fit and print area.
- HR:** Employee list printed with header row ("Name", "Dept", "Salary") repeated on every page.
- Reports:** Export dashboard as PDF with logo in header and page number in footer for sharing.

1.14 — Workbook & Worksheet Protection

Theory

Protection prevents accidental or unauthorised changes to your data and formulas. Excel has two levels of protection: Worksheet Protection (locks specific cells or the whole sheet) and Workbook Protection (locks the file structure — prevents adding/deleting/rename sheets).

Worksheet Protection — Protect Cells from Editing

Concept: By default, ALL cells are "Locked" but this has no effect until you Protect the Sheet.

STEP 1 **Unlock input cells (cells users CAN edit)** — Select the editable cells → Ctrl+1 → Protection tab → UNCHECK "Locked"

STEP 2 **Lock formula cells (cells users CANNOT edit)** — Leave them locked (default). Or select → Ctrl+1 → Protection → CHECK "Locked"

STEP 3 **Apply protection** — Review tab → Protect Sheet → set a password (optional) → click OK

STEP 4 **To unprotect** — Review → Unprotect Sheet → enter password if set

NOTE If you forget the protection password, recovering it is very difficult. Consider using a simple, memorable password or noting it securely.

Workbook Protection — Lock File Structure

Review → Protect Workbook → Structure (and/or Windows)

Structure protection prevents:

- Adding or deleting sheets
- Renaming sheets
- Moving or copying sheets
- Unhiding hidden sheets

To unprotect: Review → Protect Workbook → enter password

File-Level Password Protection

STEP 1 **File → Info → Protect Workbook** — Click the dropdown

STEP 2 **Encrypt with Password** — Type a password — this locks the file itself (requires password just to open)

STEP 3 **Save the file** — Password takes effect when the file is closed and reopened

Protection Summary

Protection Type	What It Prevents	Password Required?
Cell Lock (Locked checkbox)	Editing that specific cell when sheet is protected	No — sheet password needed

Protect Sheet	Editing cells, inserting rows, formatting changes	Optional
Protect Workbook	Adding, deleting, renaming, moving sheets	Optional
Encrypt with Password	Opening the file at all	Yes — to open file

Industry Use Cases

- **Finance:** Lock all formula cells in a budget template. Leave only input cells (assumptions) editable.
- **HR:** Protect salary sheet so employees cannot see or modify each other's data.
- **Audit:** Lock the data sheet before sharing with auditors — formulas cannot be accidentally changed.
- **Templates:** Distribute blank protected templates to teams — they fill only the designated yellow cells.

MICROSOFT EXCEL

Advanced Formulas — Complete Teaching Guide

Intermediate Level · Industry-Type Examples · All 8 Categories

Topics Covered in This Guide

1. **Lookup & Reference** VLOOKUP · HLOOKUP · INDEX · MATCH · XLOOKUP
 2. **Logical Functions** IF · IFS · AND · OR · NESTED IF
 3. **Text Functions** LEFT · RIGHT · MID · LEN · TRIM · CONCAT · TEXTJOIN
 4. **Date & Time Functions** TODAY · NOW · DATEDIF · EOMONTH · NETWORKDAYS
 5. **Statistical & Math** SUMIF · SUMIFS · COUNTIF · COUNTIFS · AVERAGEIF · AVERAGEIFS
 6. **Array & Dynamic Functions** UNIQUE · SORT · FILTER · SEQUENCE · XLOOKUP Array
 7. **Financial Functions** PMT · NPV · IRR · FV · PV
 8. **Error Handling** IFERROR · IFNA · ISERROR · IS Functions
9. Each section: Theory → Syntax → Example Data → Formulas → Result → Industry Use Cases

Sample Data Tables — Used Throughout This Guide

All formulas in this guide are demonstrated on the same set of realistic business tables. This gives you a consistent reference point across all 8 categories.

Table 1 — Employee Master (Sheet: EmpData)

emp_id	emp_name	dept	city	salary	join_date	status
E001	Aarav Shah	Sales	Pune	45000	01-Mar-2021	Active
E002	Sneha Mehta	HR	Mumbai	52000	15-Jul-2019	Active
E003	Rohit Verma	IT	Delhi	68000	10-Nov-2020	Active
E004	Priya Joshi	Sales	Pune	41000	20-Jan-2022	Inactive
E005	Karan Patel	IT	Mumbai	72000	05-Apr-2018	Active
E006	Meera Nair	HR	Delhi	49000	12-Sep-2021	Active
E007	Arjun Rao	Sales	Mumbai	55000	28-Feb-2020	Active
E008	Divya Singh	IT	Pune	63000	17-Jun-2019	Active
E009	Nikhil Das	Sales	Delhi	38000	03-Aug-2023	Active
E010	Pooja Iyer	HR	Pune	47000	22-Oct-2022	Inactive

Table 2 — Sales Transactions (Sheet: SalesData)

txn_id	emp_id	product	category	qty	unit_price	sale_date	region
T001	E001	Laptop	Electronics	2	55000	05-Jan-2024	West
T002	E003	Chair	Furniture	10	4500	08-Jan-2024	North
T003	E007	Mobile	Electronics	5	22000	12-Jan-2024	West
T004	E001	Laptop	Electronics	1	55000	18-Feb-2024	West
T005	E009	Desk	Furniture	3	8000	22-Feb-2024	North
T006	E007	Tablet	Electronics	4	18000	01-Mar-2024	West
T007	E003	Mobile	Electronics	6	22000	10-Mar-2024	North
T008	E001	Chair	Furniture	8	4500	15-Mar-2024	West

Table 3 — Course Fee Table (Sheet: Courses)

course_id	course_name	duration_hrs	fees	trainer
C001	SQL Server	40	8000	Rahul Sir
C002	Python	50	10000	Anjali Ma'am
C003	Power BI	30	6000	Rahul Sir
C004	Excel Advanced	25	5000	Meera Ma'am

Section 1 — Lookup & Reference Functions

Lookup functions search for a value in a table and return a corresponding value from another column or row. They are the most widely used advanced Excel functions in business.

1.1 VLOOKUP — Vertical Lookup

Theory

VLOOKUP searches for a value in the FIRST column of a range and returns a value from a specified column in the same row. "V" stands for Vertical — it always searches downward.

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

lookup_value → The value you are searching for

table_array → The range containing the data (always lock with \$)

col_index_num → Which column number to return (1 = first column)

[range_lookup] → FALSE = exact match (always use FALSE in business)

LIMITATION VLOOKUP can only look to the RIGHT. The lookup column must always be the leftmost column of the table_array. For left lookups, use INDEX-MATCH or XLOOKUP.

Example Formulas — EmpData Table

Formula 1 — Get Salary by Employee ID

Scenario: You have emp_id E003 in cell H2. Find Rohit's salary.

```
=VLOOKUP(H2, $A$2:$G$11, 5, FALSE)
```

H2 → emp_id to search (E003)

\$A\$2:\$G\$11 → lock the entire EmpData table

5 → salary is the 5th column

FALSE → exact match required

Result: 68000

Formula 2 — Get Department by Employee ID

```
=VLOOKUP(H2, $A$2:$G$11, 3, FALSE)
```

Returns: IT

Formula 3 — VLOOKUP with IFERROR (handle missing IDs gracefully)

```
=IFERROR(VLOOKUP(H2, $A$2:$G$11, 2, FALSE), "Employee Not Found")
```

If emp_id is not found → shows "Employee Not Found" instead of #N/A

Returns: IT

Results Table

Industry Use Cases

- **HR:** Search employee ID and pull name, dept, salary for payslip generation.
- **Finance:** Look up product code and retrieve price from a master price list.
- **Inventory:** Find item code and return stock quantity or warehouse location.

emp_id (H2)	Formula	Result
E003	=VLOOKUP(H2,\$A\$2:\$G\$11,5,FALSE)	68000
E003	=VLOOKUP(H2,\$A\$2:\$G\$11,3,FALSE)	IT
E999	=IFERROR(VLOOKUP(H2,\$A\$2:\$G\$11,2,FALSE),"Not Found")	Employee Not Found

- **Sales:** Match customer ID with customer name and region for reporting.

1.2 HLOOKUP — Horizontal Lookup

Theory

HLOOKUP searches across the FIRST ROW of a range and returns a value from a specified row. "H" stands for Horizontal — it searches across columns instead of down rows.

```
=HLOOKUP(lookup_value, table_array, row_index_num, [range_lookup])  
row_index_num → Which row number to return (1 = header row)
```

INFO HLOOKUP is used when your data is arranged horizontally — months or dates as column headers, categories as row headers. Less common than VLOOKUP in modern Excel.

Example — Monthly Target Table

Imagine a horizontal table where Row 1 has months and Row 2 has targets:

A1	B1 (Jan)	C1 (Feb)	D1 (Mar)	E1 (Apr)
----	----------	----------	----------	----------

A1	B1 (Jan)	C1 (Feb)	D1 (Mar)	E1 (Apr)
Target	420000	380000	510000	460000

```
=HLOOKUP("Mar", $B$1:$E$2, 2, FALSE)
Searches for "Mar" in Row 1 → finds it in column D
Returns the value from Row 2 of that column
Result: 510000
```

1.3 INDEX + MATCH — The Professional Alternative

Theory

INDEX returns the value of a cell at a given row and column position in a range.

MATCH returns the relative position (row number) of a value within a range.

Combined, they overcome every limitation of VLOOKUP — they can look left, look up, and are faster on large datasets.

```
INDEX Syntax: =INDEX(array, row_num, [col_num])
MATCH Syntax: =MATCH(lookup_value, lookup_array, [match_type])
Combined:     =INDEX(return_column, MATCH(lookup_value, search_column, 0))
match_type:  0 = exact match | 1 = less than | -1 = greater than
```

Example Formulas

Formula 1 — Same as VLOOKUP but can look LEFT

Find salary for emp_id E007 (Arjun Rao):

```
=INDEX($E$2:$E$11, MATCH(H2, $A$2:$A$11, 0))
MATCH(H2, $A$2:$A$11, 0) → finds "E007" in emp_id column → returns row 7
INDEX($E$2:$E$11, 7)     → returns value from row 7 of salary column
Result: 55000
```

Formula 2 — Two-way lookup (row AND column)

Find the value at a specific row AND column — like a cross-reference lookup:

```
=INDEX($A$2:$G$11, MATCH("E005", $A$2:$A$11, 0), MATCH("salary", $A$1:$G$1, 0))
First MATCH → finds row position of E005
Second MATCH → finds column position of "salary" header
INDEX       → returns the cell at that intersection
Result: 72000
```

VLOOKUP vs INDEX-MATCH Comparison

Feature	VLOOKUP	INDEX + MATCH
Look LEFT	✗ Cannot	✓ Yes
Look UP (horizontal)	✗ No	✓ Yes
Column reference	Number (breaks if columns added)	Dynamic (uses header name)
Speed on large data	Slower	Faster
Difficulty	Easy	Intermediate
Recommended for	Quick lookups	Professional / complex lookups

1.4 XLOOKUP — Modern Replacement (Excel 2021+)

Theory

XLOOKUP is the newest and most powerful lookup function. It replaces VLOOKUP, HLOOKUP, and INDEX-MATCH with a single, simpler syntax. It can search in any direction and handles errors natively.

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

lookup_value → value to search for
lookup_array → column/row to search in
return_array → column/row to return from
[if_not_found] → text to show if not found (replaces IFERROR)
[match_mode] → 0=exact, -1=exact/smaller, 1=exact/larger, 2=wildcard
[search_mode] → 1=first-to-last, -1=last-to-first, 2=binary ascending

Example Formulas

Formula 1 — Basic XLOOKUP

```
=XLOOKUP(H2, $A$2:$A$11, $E$2:$E$11, "Not Found")
```

H2 → emp_id to find
\$A\$2:\$A\$11 → search in emp_id column
\$E\$2:\$E\$11 → return from salary column
"Not Found" → show this if emp_id missing (no IFERROR needed)
Result: Returns salary directly

Formula 2 — XLOOKUP returning multiple columns at once

```
=XLOOKUP(H2, $A$2:$A$11, $B$2:$E$11, "Not Found")
```

return_array = \$B\$2:\$E\$11 (4 columns: name, dept, city, salary)
Returns ALL four values spilling into adjacent cells automatically
Result: Rohit Verma | IT | Delhi | 68000

VLOOKUP vs XLOOKUP Quick Reference

Capability	VLOOKUP	XLOOKUP
Look left	✗	✓
Built-in error message	✗ (need IFERROR)	✓
Return multiple columns	✗	✓
Search last-to-first	✗	✓
Wildcard search	✗	✓ (match_mode=2)
Available in	All versions	Excel 2021+ / Microsoft 365

Industry Use Cases — Lookup Functions

- **Payroll:** Pull employee name, grade, and salary by ID for salary slips.
- **Procurement:** Look up vendor code to retrieve contact, payment terms, and rating.
- **Student Reports:** Search roll number to auto-fill name, marks, and grade in report card.
- **Banking:** Match account number with customer name and branch for statement generation.

Section 2 — Logical Functions

Logical functions test conditions and return different results based on whether the condition is TRUE or FALSE. They are the foundation of all decision-making in Excel.

2.1 IF — Basic Decision Making

Theory

IF evaluates a condition and returns one value if TRUE, another if FALSE. It is the most fundamental logical function.

```
=IF(logical_test, value_if_true, value_if_false)
logical_test   → any expression that evaluates to TRUE or FALSE
value_if_true  → returned when condition is TRUE
value_if_false → returned when condition is FALSE
```

Example Formulas — EmpData Table

Formula 1 — Check if Employee is Active

```
=IF(G2="Active", "✓ Working", "✗ Left")
G2 = status column
If status is "Active"   → shows "✓ Working"
If status is "Inactive" → shows "✗ Left"
```

Formula 2 — Salary Category

```
=IF(E2>=60000, "Senior", "Junior")
Salary ≥ 60000 → "Senior"
Salary < 60000 → "Junior"
```

Formula 3 — Bonus Calculation (10% if Active, else 0)

```
=IF(G2="Active", E2*10%, 0)
Active employees get 10% bonus
Inactive employees get 0
```

2.2 Nested IF — Multiple Conditions

Theory

A Nested IF places another IF inside the value_if_false argument to test multiple conditions in sequence. Think of it as "IF this... ELSE IF that... ELSE something else."

```
=IF(condition1, result1,
    IF(condition2, result2,
        IF(condition3, result3, default_result)))
Excel evaluates from left to right and stops at the first TRUE condition.
```

BEST PRACTICE Avoid nesting more than 3-4 levels deep. Use IFS() for multiple conditions — it is cleaner and easier to read.

Example — Grade Students by Marks

```
=IF(B2>=90, "A+ Distinction",
  IF(B2>=75, "A First Class",
    IF(B2>=60, "B Second Class",
      IF(B2>=40, "C Pass",
        "F Fail"))))
```

B2 = marks out of 100

Evaluates top condition first → stops at first match

Result Table

Marks (B2)	Result
95	A+ Distinction
78	A First Class
62	B Second Class
45	C Pass
35	F Fail

2.3 IFS — Cleaner Alternative to Nested IF

Theory

IFS tests multiple conditions in order and returns the value for the first TRUE condition. Available in Excel 2019+.

```
=IFS(condition1, result1, condition2, result2, condition3, result3, ...)
```

No nesting required — all conditions listed in one flat formula

Add TRUE, "default" at the end as a catch-all

Example — Same Grade Logic with IFS

```
=IFS(B2>=90, "A+ Distinction",
  B2>=75, "A First Class",
  B2>=60, "B Second Class",
  B2>=40, "C Pass",
  TRUE, "F Fail")
```

Much easier to read and maintain than nested IF

2.4 AND / OR — Combine Multiple Conditions

Theory

AND: Returns TRUE only if ALL conditions are TRUE.

OR: Returns TRUE if ANY ONE condition is TRUE.

```
=AND(condition1, condition2, condition3, ...)
```

```
=OR(condition1, condition2, condition3, ...)
```

Example Formulas

Formula 1 — AND: Active employee earning above 50000

```
=IF(AND(G2="Active", E2>50000), "High Value Active", "Other")
```

BOTH conditions must be true to show "High Value Active"

E003 (Rohit, IT, 68000, Active) → High Value Active

E004 (Priya, Sales, 41000, Inactive) → Other

Formula 2 — OR: Sales dept OR Mumbai city

```
=IF(OR(C2="Sales", D2="Mumbai"), "Target Group", "Others")
```

Employee is in Sales OR is based in Mumbai → "Target Group"

E001 (Sales, Pune) → Target Group (Sales condition true)

E002 (HR, Mumbai) → Target Group (Mumbai condition true)

E003 (IT, Delhi) → Others (neither condition true)

Industry Use Cases — Logical Functions

- **HR:** Flag employees eligible for promotion: Active AND experience > 3 years AND rating > 4.
- **Sales:** Calculate tiered commission: IF sales > 5L then 10%, IF > 3L then 7%, else 5%.
- **Finance:** Mark invoices as overdue IF payment_date > due_date AND status ≠ "Paid".
- **Academics:** Assign grades to students using IFS based on marks ranges.

Section 3 — Text Functions

Text functions manipulate, extract, and combine text strings. They are essential for data cleaning, report formatting, and building dynamic labels.

3.1 LEFT, RIGHT, MID — Extract Parts of Text

```
=LEFT(text, num_chars)      → extract from the LEFT
=RIGHT(text, num_chars)     → extract from the RIGHT
=MID(text, start_num, num_chars) → extract from the MIDDLE
```

Example Formulas — EmpData Table

Formula 1 — Extract Department Code from emp_id

emp_id format: E001. Extract the numeric part:

```
=RIGHT(A2, 3)      → "001" (last 3 characters)
=LEFT(A2, 1)       → "E"   (first character — the prefix)
=MID(A2, 2, 3)     → "001" (start at pos 2, take 3 characters)
```

Formula 2 — Extract First Name from Full Name

emp_name: Aarav Shah. Get first name only:

```
=LEFT(B2, FIND(" ", B2) - 1)

FIND(" ", B2)      → finds position of the space character (e.g., 6)
LEFT(B2, 6-1)      → takes 5 characters from left
Result: "Aarav"
```

Formula 3 — Extract Last Name

```
=MID(B2, FIND(" ", B2)+1, LEN(B2))

FIND(" ", B2)+1    → position after the space
LEN(B2)            → total length (used as num_chars to get rest of string)
Result: "Shah"
```

3.2 LEN — Count Characters

```
=LEN(text)      → returns total number of characters including spaces
```

Example

```
=LEN("Aarav Shah") → 10
=LEN(B2)           → counts characters in cell B2

-- Practical: Check if emp_id has exactly 4 characters
=IF(LEN(A2)=4, "Valid", "Invalid ID")
```

3.3 TRIM — Remove Extra Spaces

```
=TRIM(text)      → removes all leading, trailing, and extra middle spaces
                  (keeps single spaces between words)
```

INFO TRIM is the most important data cleaning function. When data is imported from databases or copied from websites, names and codes often have invisible spaces that break lookups. Always TRIM before VLOOKUP.

```
=TRIM("  Aarav  Shah  ") → "Aarav Shah"  
  
-- Clean before lookup (prevents #N/A errors)  
=VLOOKUP(TRIM(H2), $A$2:$G$11, 2, FALSE)
```

3.4 CONCAT and TEXTJOIN — Combine Text

```
=CONCAT(text1, text2, ...) → joins text, no separator  
=TEXTJOIN(delimiter, ignore_empty, ...) → joins with separator, can skip blanks  
=text1 & text2 → ampersand shortcut for joining
```

Example Formulas

Formula 1 — Create Full Address from Parts

```
=CONCAT(B2, " | ", C2, " | ", D2)  
  
Result: "Aarav Shah | Sales | Pune"
```

Formula 2 — TEXTJOIN — Join Non-Empty Cells

```
=TEXTJOIN(", ", TRUE, D2:D11)  
  
Joins all city values with ", " separator  
TRUE = ignore empty cells  
Result: "Pune, Mumbai, Delhi, Pune, Mumbai, Delhi, Mumbai, Pune, Delhi, Pune"
```

Formula 3 — Build Employee Code

```
=LEFT(C2,2) & "-" & RIGHT(A2,3)  
  
C2 = "Sales" → LEFT gives "Sa"  
A2 = "E001" → RIGHT gives "001"  
Result: "Sa-001" (custom employee code)
```

Industry Use Cases — Text Functions

- **Data Cleaning:** TRIM + UPPER/LOWER to standardise names imported from different systems.
- **ID Generation:** Create unique IDs by combining LEFT(dept,3) & "-" & row number.
- **Email Builder:** =LOWER(LEFT(B2,3)) & "." & LOWER(MID(B2,FIND(" ",B2)+1,10)) & "@company.com"
- **Report Labels:** Combine month, year, and region into a single dashboard header cell.

Section 4 — Date & Time Functions

Date functions allow you to calculate durations, find deadlines, and work with dates in reports and dashboards. Excel stores dates as serial numbers — these functions make them usable.

4.1 TODAY and NOW

```
=TODAY()    → returns today's date (no time)
=NOW()      → returns current date AND time
```

Both update automatically every time the worksheet recalculates

```
-- Days since joining
=TODAY() - F2

-- Is the employee's join date this year?
=IF(YEAR(F2)=YEAR(TODAY()), "Joined This Year", "Senior Employee")
```

4.2 DATEDIF — Calculate Age / Tenure / Duration

Theory

DATEDIF calculates the difference between two dates in years, months, or days. It is a hidden function — it does not appear in autocomplete, but works perfectly.

```
=DATEDIF(start_date, end_date, unit)

"Y"  → complete years between dates
"M"  → complete months between dates
"D"  → complete days between dates
"YM" → months ignoring years (useful for "X years Y months")
"MD" → days ignoring months and years
```

Example Formulas — EmpData Table (F = join_date)

Formula 1 — Employee Tenure in Years

```
=DATEDIF(F2, TODAY(), "Y")
```

F2 = 01-Mar-2021 → Result: 3 years (as of 2024)

Formula 2 — Tenure as "X Years Y Months"

```
=DATEDIF(F2,TODAY(),"Y") & " Years " & DATEDIF(F2,TODAY(),"YM") & " Months"
```

Result: "3 Years 4 Months"

Formula 3 — Employee Age (if DOB in column)

```
=DATEDIF(DOB_cell, TODAY(), "Y") & " Years Old"
```

4.3 EOMONTH — End of Month Date

```
=EOMONTH(start_date, months)
```

```
months = 0 → last day of the SAME month  
months = 1 → last day of NEXT month  
months = -1 → last day of PREVIOUS month
```

Example

```
=EOMONTH(TODAY(), 0) → last day of current month e.g. 31-Jan-2024  
=EOMONTH(TODAY(), 1) → last day of next month e.g. 29-Feb-2024  
  
-- Invoice due at end of next month  
=EOMONTH(invoice_date, 1)
```

4.4 NETWORKDAYS — Working Days Excluding Weekends

```
=NETWORKDAYS(start_date, end_date, [holidays])
```

```
Counts business days (Mon-Fri) between two dates  
[holidays] = optional range of public holiday dates to exclude
```

Example

```
-- Working days to complete a project  
=NETWORKDAYS(start_date_cell, end_date_cell)  
  
-- Exclude public holidays listed in H2:H10  
=NETWORKDAYS(start_date_cell, end_date_cell, $H$2:$H$10)  
  
-- Days remaining to deadline  
=NETWORKDAYS(TODAY(), deadline_date)
```

Useful Date Helper Functions

Function	Syntax	Returns
YEAR	=YEAR(date)	Year number (e.g. 2024)
MONTH	=MONTH(date)	Month number (1-12)
DAY	=DAY(date)	Day number (1-31)
TEXT	=TEXT(date,"DD-MMM-YYYY")	Formatted date as text
WEEKDAY	=WEEKDAY(date,2)	Day of week (1=Mon, 7=Sun)
DATE	=DATE(year,month,day)	Construct a date from parts
DAYS	=DAYS(end_date,start_date)	Simple day count between dates

Industry Use Cases — Date Functions

- **HR:** Calculate years of service for increment letters and long-service awards.
- **Project Management:** Count working days remaining to project deadline (NETWORKDAYS).
- **Finance:** Calculate invoice due dates as end of next month (EOMONTH).
- **Payroll:** Determine number of working days in a month for daily-rate employees.

Section 5 — Statistical & Math Functions

These functions count, sum, and average data based on one or more conditions. They are the backbone of all Excel dashboards and management reports.

5.1 COUNTIF and COUNTIFS

<code>=COUNTIF(range, criteria)</code>	→ one condition
<code>=COUNTIFS(range1, criteria1, range2, criteria2)</code>	→ multiple conditions

Example Formulas — EmpData Table

Formula 1 — Count employees in Sales department

```
=COUNTIF($C$2:$C$11, "Sales")  
Result: 3 (Aarav, Priya, Arjun, Nikhil → 4 actually)
```

Formula 2 — Count Active employees in IT

```
=COUNTIFS($C$2:$C$11, "IT", $G$2:$G$11, "Active")  
  
Both conditions must match the SAME row  
Result: 3 (Rohit, Karan, Divya are all Active in IT)
```

Formula 3 — Count employees with salary above 50000

```
=COUNTIF($E$2:$E$11, ">50000")  
  
Criteria with operators must be in quotes: ">50000", ">=", "<>Active"  
Result: 5
```

5.2 SUMIF and SUMIFS

<code>=SUMIF(range, criteria, sum_range)</code>	→ one condition
<code>=SUMIFS(sum_range, range1, criteria1, range2, criteria2)</code>	→ multiple conditions

NOTE: In SUMIFS, sum_range comes FIRST — opposite to SUMIF!

Example Formulas

Formula 1 — Total salary of Sales department

```
=SUMIF($C$2:$C$11, "Sales", $E$2:$E$11)  
  
C = dept column, "Sales" = criteria, E = salary to sum  
Result: 45000 + 41000 + 55000 + 38000 = 179000
```

Formula 2 — Total salary of Active IT employees

```
=SUMIFS($E$2:$E$11, $C$2:$C$11, "IT", $G$2:$G$11, "Active")  
  
Sum salary WHERE dept=IT AND status=Active
```

Result: 68000 + 72000 + 63000 = 203000

Formula 3 — Total sales amount for Electronics in SalesData

```
=SUMIF(SalesData!$D$2:$D$9, "Electronics", SalesData!$F$2:$F$9)
```

References another sheet using SheetName! prefix

5.3 AVERAGEIF and AVERAGEIFS

```
=AVERAGEIF(range, criteria, average_range)
```

```
=AVERAGEIFS(average_range, range1, criteria1, range2, criteria2)
```

Example Formulas

Formula 1 — Average salary in HR department

```
=AVERAGEIF($C$2:$C$11, "HR", $E$2:$E$11)
```

Result: (52000+49000+47000)/3 = 49333

Formula 2 — Average salary of Active Pune employees

```
=AVERAGEIFS($E$2:$E$11, $D$2:$D$11, "Pune", $G$2:$G$11, "Active")
```

Result: (45000+68000+63000)/3 = 58667

Summary Reference Table

Function	What it does	Conditions
COUNTIF	Count cells meeting 1 condition	1
COUNTIFS	Count cells meeting all conditions	Multiple (AND logic)
SUMIF	Sum values based on 1 condition	1
SUMIFS	Sum values meeting all conditions	Multiple (AND logic)
AVERAGEIF	Average values based on 1 condition	1
AVERAGEIFS	Average values meeting all conditions	Multiple (AND logic)

Industry Use Cases — Statistical Functions

- **Sales Dashboards:** SUMIFS to get revenue by region + product + month for pivot-style reports.
- **HR Analytics:** COUNTIFS to count employees by department, grade, and city.
- **Finance:** SUMIF to total expenses by cost centre or account head.
- **Inventory:** COUNTIF to count items below reorder level for procurement alerts.

Section 6 — Array & Dynamic Array Functions

Dynamic array functions are available in Excel 365 and Excel 2021+. They automatically SPILL results into multiple cells without pressing Ctrl+Shift+Enter. They can completely replace many manual tasks.

SPILL When a formula returns multiple values, Excel automatically fills them into adjacent empty cells. This spilled range is indicated by a blue border. If a cell in the spill range is occupied, you get a #SPILL! error.

6.1 UNIQUE — Extract Distinct Values

```
=UNIQUE(array, [by_col], [exactly_once])
```

array → the range to extract unique values from
by_col → FALSE (default) = unique rows, TRUE = unique columns
exactly_once → TRUE = only values appearing exactly once

Example Formulas

Formula 1 — Get Unique Departments

```
=UNIQUE($C$2:$C$11)
```

Returns: Sales, HR, IT (3 unique values spill into 3 cells below)

Formula 2 — Get Unique Cities for Sales Dept (nested with FILTER)

```
=UNIQUE(FILTER($D$2:$D$11, $C$2:$C$11="Sales"))
```

First FILTER gets all cities where dept=Sales
Then UNIQUE removes duplicates from that filtered list

6.2 SORT — Sort a Range Dynamically

```
=SORT(array, [sort_index], [sort_order], [by_col])
```

sort_index → column number to sort by (default 1)
sort_order → 1=ascending (default), -1=descending
by_col → FALSE=sort rows (default), TRUE=sort columns

Example Formulas

Formula 1 — Sort employees by salary descending

```
=SORT($A$2:$G$11, 5, -1)
```

5 → sort by 5th column (salary)
-1 → descending order
Entire table spills out sorted — no manual sorting needed

Formula 2 — Alphabetical list of names

```
=SORT($B$2:$B$11)
```

Sorts column B (names) A to Z automatically

6.3 FILTER — Extract Rows Meeting a Condition

```
=FILTER(array, include, [if_empty])
```

array → full data range to filter
include → logical array – TRUE rows are kept
if_empty → what to show if no results match (e.g. "No Data")

Example Formulas

Formula 1 — All Active employees from Sales

```
=FILTER($A$2:$G$11, ($C$2:$C$11="Sales") * ($G$2:$G$11="Active"), "No Match")
```

Multiplication (*) = AND logic between two conditions
Addition (+) = OR logic between two conditions
Returns all rows where dept=Sales AND status=Active

Formula 2 — Employees with salary above 55000

```
=FILTER($A$2:$G$11, $E$2:$E$11 > 55000, "No employees")
```

Returns: Rohit (68000), Karan (72000), Arjun (55000+), Divya (63000)

6.4 SEQUENCE — Generate Number Series

```
=SEQUENCE(rows, [cols], [start], [step])
```

rows → how many rows to generate
cols → how many columns (default 1)
start → starting value (default 1)
step → increment between values (default 1)

Example Formulas

```
=SEQUENCE(10) → 1,2,3,4,5,6,7,8,9,10 (vertical)  
=SEQUENCE(1, 12) → 1 to 12 in a row (months)  
=SEQUENCE(5, 1, 2024, 1) → 2024,2025,2026,2027,2028 (years)
```

```
-- Auto-generate invoice numbers  
="INV-" & TEXT(SEQUENCE(10), "000")  
Result: INV-001, INV-002, INV-003 ...
```

Industry Use Cases — Dynamic Functions

- **Reports:** SORT + FILTER to auto-build a "Top 10 customers by revenue" list that updates live.
- **Dashboards:** UNIQUE to populate dropdown lists that automatically include new categories.
- **HR:** FILTER to extract all employees due for appraisal this month.
- **Finance:** SEQUENCE to generate monthly date series for cash flow models.

Section 7 — Financial Functions

Financial functions solve problems in banking, investment, and loan analysis. They are used heavily by finance professionals, analysts, and anyone dealing with loans or investment decisions.

SIGN CONVENTION In financial functions, money paid OUT is NEGATIVE, money received is POSITIVE. This is the standard cash flow convention. For loan EMI, the PV (loan amount) is positive and PMT result comes out negative — meaning you pay it out.

7.1 PMT — Calculate Loan EMI

Theory

PMT (Payment) calculates the fixed periodic payment for a loan — the EMI. It assumes a constant interest rate and regular payments.

```
=PMT(rate, nper, pv, [fv], [type])
```

rate → interest rate PER PERIOD (annual rate ÷ 12 for monthly EMI)
nper → total number of payment periods (years × 12 for monthly)
pv → present value = loan amount (positive number)
fv → future value after last payment (default 0)
type → 0=payment at end of period (default), 1=beginning of period

Example — Home Loan EMI Calculation

Parameter	Value	Cell
Loan Amount	₹ 50,00,000	B1
Annual Interest Rate	8.5%	B2
Loan Tenure	20 years	B3

```
=PMT(B2/12, B3*12, B1)
```

B2/12 → monthly rate = 8.5%/12 = 0.708%
B3*12 → total months = 20×12 = 240
B1 → loan amount = 5000000

Result: -₹43,391 (negative = you pay this out each month)

-- Show as positive number

```
=-PMT(B2/12, B3*12, B1)
```

Result: ₹43,391 per month

7.2 PV — Present Value (What is a Future Amount Worth Today?)

```
=PV(rate, nper, pmt, [fv], [type])
```

Calculates current worth of a series of future payments at a given discount rate
pmt → regular payment each period
fv → future lump sum value

Example

```
-- What is ₹10,000/month for 5 years worth today at 8%?  
=PV(8%/12, 5*12, 10000)
```

Result: -₹4,94,210 (you would pay ~4.94 lakhs today for this stream)

7.3 FV — Future Value (How Much Will Savings Grow?)

```
=FV(rate, nper, pmt, [pv], [type])
```

Calculates the future value of regular investments at a given interest rate

Example — SIP Maturity Calculation

```
-- Invest ₹5,000/month for 10 years at 12% annual return  
=FV(12%/12, 10*12, -5000)
```

Rate: 12%/12 = 1% per month

nper: 10×12 = 120 months

pmt: -5000 (negative — you pay it out each month)

Result: ₹11,61,695 (future value of your SIP after 10 years)

7.4 NPV — Net Present Value (Is This Investment Worth It?)

```
=NPV(rate, value1, value2, ...)
```

Calculates the present value of a series of future cash flows

If NPV > 0 → investment is profitable

If NPV < 0 → investment will lose money

IMPORTANT: NPV does NOT include Year 0 investment automatically.

Add initial investment separately: =NPV(rate, future_flows) + initial_investment

Example — Business Investment Decision

Year	Cash Flow
0 (Initial Investment)	-₹5,00,000
Year 1	₹1,50,000
Year 2	₹2,00,000
Year 3	₹2,50,000
Year 4	₹2,00,000

```
=NPV(10%, 150000, 200000, 250000, 200000) - 500000
```

10% = discount rate (cost of capital)

Year 0 investment (-500000) added at the end

Result: ₹45,756 → Positive NPV → Accept the investment!

7.5 IRR — Internal Rate of Return

```
=IRR(values, [guess])
```

Returns the discount rate at which NPV = 0

If IRR > cost of capital → invest

If IRR < cost of capital → do not invest

values must include at least one negative and one positive cash flow

Example — Compare Two Projects

```
-- Cash flows in B2:B6 (Year 0 to Year 4)
```

```
=IRR(B2:B6)
```

```
-- If result is 18% and your cost of capital is 10%:
```

```
-- IRR (18%) > Cost of Capital (10%) → INVEST ✓
```

Industry Use Cases — Financial Functions

- **Banking:** PMT to calculate EMI for home loans, car loans, personal loans.
- **Corporate Finance:** NPV and IRR to evaluate capital projects and business acquisitions.
- **Wealth Management:** FV to project SIP returns and retirement corpus for clients.
- **Real Estate:** PV to find maximum property price justified by expected rental income.

Section 8 — Error Handling Functions

Excel returns error values when a formula cannot compute a result. Error handling functions detect and manage these errors gracefully — essential for professional, clean reports.

Common Excel Errors Reference

Error	Cause	Common Fix
#N/A	VLOOKUP/MATCH value not found	IFERROR or XLOOKUP with if_not_found
#DIV/0!	Dividing by zero or empty cell	=IF(denominator=0, 0, numerator/denominator)
#VALUE!	Wrong data type (text in math formula)	Check cell types, use VALUE() to convert
#REF!	Formula references a deleted cell or row	Check and correct the reference
#NAME?	Function name misspelled	Check spelling
#NUM!	Invalid number (e.g. sqrt of negative)	Validate inputs with IF
#NULL!	Incorrect range operator used	Fix the range reference
#SPILL!	Spill range is blocked by another value	Clear cells in the spill range

8.1 IFERROR — Catch ANY Error

Theory

IFERROR is the most commonly used error handler. It wraps a formula and returns a custom value if ANY error occurs, otherwise returns the formula result.

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

value → the formula to evaluate
value_if_error → what to return if ANY error occurs (#N/A, #DIV/0!, etc.)

Example Formulas

Formula 1 — Safe VLOOKUP

```
-- Without IFERROR:  
=VLOOKUP(H2, $A$2:$G$11, 2, FALSE) → shows ugly #N/A if not found  
  
-- With IFERROR:  
=IFERROR(VLOOKUP(H2, $A$2:$G$11, 2, FALSE), "Employee Not Found")  
  
Shows "Employee Not Found" instead of #N/A — professional look
```

Formula 2 — Safe Division (avoid #DIV/0!)

```
-- Revenue per employee:  
=IFERROR(B2/C2, 0)  
If C2 is 0 or empty → returns 0 instead of #DIV/0! error
```

Formula 3 — Chain formulas safely

```
=IFERROR(INDEX($E$2:$E$11, MATCH(H2,$A$2:$A$11,0)), "Not Found")
```

Wraps the entire INDEX-MATCH in IFERROR
Any failure at any point → returns "Not Found"

8.2 IFNA — Catch Only #N/A Errors

Theory

IFNA handles only #N/A errors — not other error types. Use it when you want to catch "not found" situations but still see other errors like #DIV/0! or #VALUE! so you can fix them.

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

Only activates for #N/A errors (lookup not found)
Other errors pass through unchanged

Example

```
=IFNA(VLOOKUP(H2, $A$2:$G$11, 2, FALSE), "New Employee")
```

#N/A (emp_id not in list) → "New Employee"
#VALUE! or #REF! → still shown (you need to fix those)

8.3 ISERROR / ISERR / ISNA — Test for Errors

```
=ISERROR(value) → TRUE if value is ANY error type  
=ISERR(value) → TRUE if value is any error EXCEPT #N/A  
=ISNA(value) → TRUE if value is specifically #N/A
```

These return TRUE/FALSE — often used inside IF formulas

Example Formulas

Formula 1 — Conditional formatting trigger

```
=ISERROR(VLOOKUP(H2, $A$2:$G$11, 2, FALSE))
```

Returns TRUE if emp_id not found → use in Conditional Formatting
to highlight the cell red when an invalid ID is entered

Formula 2 — Show 0 only for #N/A, flag others

```
=IF(ISNA(VLOOKUP(H2, $A$2:$G$11, 2, FALSE)), 0, VLOOKUP(H2, $A$2:$G$11, 2, FALSE))
```

#N/A → 0 (employee doesn't exist yet)
Other errors → still displayed so you investigate

8.4 IS Functions — Data Validation

Function	Returns TRUE when...	Common Use
ISNUMBER(val)	Cell contains a number	Validate inputs before calculation

Function	Returns TRUE when...	Common Use
ISTEXT(val)	Cell contains text	Check for text in numeric columns
ISBLANK(val)	Cell is empty	Skip blank rows in formulas
ISLOGICAL(val)	Cell contains TRUE/FALSE	Validate flag columns
ISNUMBER(FIND("@",val))	Text contains "@"	Basic email validation

Example — Validate Salary Input

```
=IF(ISNUMBER(E2), E2*10%, "Invalid: Salary must be a number")
```

Prevents formula errors when someone types text in the salary column

Industry Use Cases — Error Handling

- **Dashboards:** IFERROR ensures no #N/A or #DIV/0! errors appear in management reports.
- **Data Import:** ISERROR to flag rows with bad data after importing from external systems.
- **Forms:** ISNUMBER / ISBLANK to validate user-entered data in input sheets.
- **Finance:** IFERROR on ratio formulas to show 0 or "N/A" when denominator is zero.

Master Quick Reference — All Functions

Lookup & Reference

Function	Syntax (Short)	Key Use
VLOOKUP	=VLOOKUP(val, table, col#, FALSE)	Search first column, return right
HLOOKUP	=HLOOKUP(val, table, row#, FALSE)	Search first row, return down
INDEX	=INDEX(range, row#, col#)	Return value at position
MATCH	=MATCH(val, range, 0)	Return position of value
XLOOKUP	=XLOOKUP(val, search, return, "err")	Modern: any direction, built-in error

Logical

Function	Syntax (Short)	Key Use
IF	=IF(test, true, false)	Single condition decision
IFS	=IFS(t1,r1, t2,r2, TRUE,default)	Multiple conditions, no nesting
AND	=AND(cond1, cond2)	All conditions must be TRUE
OR	=OR(cond1, cond2)	At least one condition TRUE
NESTED IF	=IF(t1,r1,IF(t2,r2,default))	Cascading decisions

Text

Function	Syntax (Short)	Key Use
LEFT	=LEFT(text, n)	Extract n chars from left
RIGHT	=RIGHT(text, n)	Extract n chars from right
MID	=MID(text, start, n)	Extract from middle
LEN	=LEN(text)	Count characters
TRIM	=TRIM(text)	Remove extra spaces
CONCAT	=CONCAT(t1, t2)	Join text, no separator
TEXTJOIN	=TEXTJOIN(" ", TRUE, range)	Join with separator, skip blanks
FIND	=FIND(find_text, within)	Find position of character

Date & Time

Function	Syntax (Short)	Key Use
TODAY	=TODAY()	Current date (auto-updates)
NOW	=NOW()	Current date and time
DATEDIF	=DATEDIF(start, end, "Y")	Years/months/days between dates
EOMONTH	=EOMONTH(date, months)	Last day of a month
NETWORKDAYS	=NETWORKDAYS(start, end, holidays)	Working days between dates
YEAR/MONTH/DAY	=YEAR(date)	Extract year, month, or day
TEXT	=TEXT(date, "DD-MMM-YYYY")	Format date as text

Statistical & Math

Function	Syntax (Short)	Key Use
COUNTIF	=COUNTIF(range, criteria)	Count matching cells (1 condition)
COUNTIFS	=COUNTIFS(r1,c1, r2,c2)	Count with multiple conditions
SUMIF	=SUMIF(range, criteria, sum_range)	Sum matching values (1 condition)
SUMIFS	=SUMIFS(sum_r, r1,c1, r2,c2)	Sum with multiple conditions
AVERAGEIF	=AVERAGEIF(range, crit, avg_range)	Average with 1 condition
AVERAGEIFS	=AVERAGEIFS(avg_r, r1,c1, r2,c2)	Average with multiple conditions

Array & Dynamic

Function	Syntax (Short)	Key Use
UNIQUE	=UNIQUE(range)	Extract distinct values
SORT	=SORT(range, col#, -1)	Sort a range dynamically
FILTER	=FILTER(range, condition, "empty")	Extract matching rows
SEQUENCE	=SEQUENCE(rows, cols, start, step)	Generate number series

Financial

Function	Syntax (Short)	Key Use
PMT	=PMT(rate/12, years*12, loan_amount)	Monthly EMI for a loan
PV	=PV(rate, nper, pmt)	Present value of future payments
FV	=FV(rate, nper, -pmt)	Future value of regular investment
NPV	=NPV(rate, cf1, cf2,...) + initial	Net present value of project
IRR	=IRR(cashflow_range)	Internal rate of return

Error Handling

Function	Syntax (Short)	Key Use
IFERROR	=IFERROR(formula, "error msg")	Catch any error gracefully
IFNA	=IFNA(formula, "not found")	Catch only #N/A errors
ISERROR	=ISERROR(value)	Returns TRUE for any error
ISNA	=ISNA(value)	Returns TRUE only for #N/A
ISNUMBER	=ISNUMBER(value)	Returns TRUE if numeric
ISBLANK	=ISBLANK(value)	Returns TRUE if cell empty

CHAPTER 3 — DATA CLEANING

The most important skill for anyone working with real-world business data

INFO In the real world, 80% of an analyst's time is spent on DATA CLEANING before any analysis begins. Raw data from ERP systems, databases, and forms is almost always messy — duplicates, extra spaces, wrong formats, mixed cases. This chapter covers all the tools to fix it.

3.1 — Remove Duplicates

Theory

Duplicate rows occur when the same record appears more than once in a dataset. This happens during data imports, copy-paste operations, and system exports. Duplicates cause incorrect counts, inflated totals, and wrong analysis results.

Remove Duplicates Tool

STEP 1 Click any cell in the data — Make sure the dataset has headers

STEP 2 Go to Data tab — Data → Data Tools group → Remove Duplicates

STEP 3 Select columns to check — Tick the columns that define a "duplicate". E.g. tick emp_id only → removes rows with same emp_id

STEP 4 Click OK — Excel removes duplicates and shows how many were removed and how many unique records remain

Example — Employee Data with Duplicates

emp_id	emp_name	dept	Issue
E001	Aarav Shah	Sales	← Original
E002	Sneha Mehta	HR	← Original
E001	Aarav Shah	Sales	← DUPLICATE — same emp_id, same row
E003	Rohit Verma	IT	← Original
E002	Sneha Mehta	HR	← DUPLICATE — same emp_id, same row

After Remove Duplicates (checking emp_id column only): 3 unique rows remain. Excel message: "2 duplicate values found and removed, 3 unique values remain."

Partial Duplicate Check

INFO You choose WHICH columns define a duplicate. Checking "emp_id" alone removes rows with same ID. Checking "emp_name + dept" removes rows where both name AND dept are the same. Choose based on what makes a record truly unique.

Alternative — COUNTIF to Find Duplicates (without deleting)

```
-- Add a helper column to identify duplicates BEFORE removing:
=COUNTIF($A$2:A2, A2)

-- Copy this down: rows where result > 1 are duplicates
-- Filter for >1 to review duplicates before deleting
-- Safer approach when you want to manually decide which to keep
```

Industry Use Cases

- **CRM Data:** Customer list merged from two systems has 500 duplicate entries — Remove Duplicates cleans it in seconds.
- **Invoice Register:** Check for duplicate invoice numbers before submitting for payment — prevents double-payment.
- **Email Lists:** Marketing database exported twice by mistake — remove duplicates before sending campaign.
- **Exam Results:** Student marks imported with duplicate roll numbers — deduplicate before grade processing.

3.2 — Text to Columns

Theory

Text to Columns splits the content of one cell into multiple cells — separated by a delimiter (comma, space, hyphen, pipe) or by a fixed number of characters. It is essential when data arrives as one combined field that needs to be split.

Two Types of Text to Columns

Type	When to Use	Example
Delimited	Data separated by a character (comma, tab, space, pipe, semicolon)	"Aarav Shah,Sales,Pune" → split by comma
Fixed Width	Data in fixed positions regardless of content length	"E001 Aarav Shah" → split at position 5

Step-by-Step — Delimited

Scenario: Column A has "Aarav Shah | Sales | Pune" — split into 3 columns.

STEP 1 **Select column A** — Click the column A header to select the entire column

STEP 2 **Data → Text to Columns** — Opens the Convert Text to Columns Wizard

- STEP 3 Choose Delimited → Next** — Tick "Delimited" option
- STEP 4 Choose delimiter** — Tick "Other" → type the pipe symbol | → click Next
- STEP 5 Format each column** — Set each resulting column as General, Text, or Date format
- STEP 6 Set Destination** — Choose where the split data should go (default: overwrites same column)
- STEP 7 Click Finish** — Data is split into separate columns instantly

Step-by-Step — Fixed Width

Scenario: "E001Aarav" — first 4 characters are the ID, remaining is the name.

- STEP 1 Select column** — Select the column with combined data
- STEP 2 Data → Text to Columns → Fixed Width** — Choose Fixed Width in the wizard
- STEP 3 Click to add break lines** — Click at position 4 in the preview ruler to set the split point
- STEP 4 Click Finish** — Data splits at that character position

Other Uses of Text to Columns

Problem	Fix Using Text to Columns
Dates imported as text ("20240115")	Fixed Width: split YYYY(4) + MM(2) + DD(2), then combine with DATE()
Numbers stored as text (left-aligned)	Select column → Data → Text to Columns → Finish (no changes) — converts to numbers
CSV data pasted as one long string	Delimited by comma → splits into proper columns
Full name in one cell	Delimited by space → first name and last name in separate columns
Date-time combined "01-Jan-2024 09:30"	Delimited by space → date in col A, time in col B

Industry Use Cases

- **ERP Exports:** SAP/Tally exports account codes and names in one field — split with Text to Columns.
- **Bank Statements:** Date and description merged in CSV export — split to get clean date column for analysis.
- **Contact Data:** Full address in one cell — split by comma into street, city, pincode, state.
- **Number Fix:** Numbers from database import left-align as text — Text to Columns (just Finish) converts them instantly.

3.3 — Flash Fill (Advanced Data Cleaning)

Theory

Flash Fill uses pattern recognition to transform data based on examples you show it. Unlike Text to Columns (which only splits), Flash Fill can combine, reformat, extract, and clean data in almost any way — without writing a single formula.

NOTE Flash Fill is one-time — it fills static values, not formulas. If your source data changes later, Flash Fill results will NOT update automatically. For dynamic transformation, use formulas or Power Query.

Advanced Flash Fill Examples — Data Cleaning Scenarios

Scenario	Original Data	Flash Fill Output	Cleaning Action
Standardise Names	"AARAV SHAH"	"Aarav Shah"	UPPER → Proper Case
Extract Email Domain	"aarav@gmail.com"	"gmail.com"	Extract after @
Mask Phone Number	"9876543210"	"XXXX-X43210"	Partially mask sensitive data
Build Employee Code	"Aarav" + "Sales"	"AA-SAL-001"	Combine + abbreviate
Reformat Date	"2024-01-15"	"15 Jan 2024"	Date format change
Extract City from Address	"12 MG Road, Pune"	"Pune"	Extract after last comma
Combine First+Last Name	"Aarav" "Shah"	"A. Shah"	Initial + Last Name format
Clean Product Code	" PRD-001 "	"PRD-001"	Remove extra spaces

When Flash Fill Cannot Help — Use Formulas Instead

Situation	Use Instead
Data will be updated regularly	Write a formula (TRIM, LEFT, MID, TEXTJOIN etc.)
Pattern is too complex for recognition	Use Text to Columns or Power Query
Multiple transformations needed	Chain functions: =PROPER(TRIM(A2))
Need the result to stay dynamic	Formula updates when source changes; Flash Fill does not

Industry Use Cases

- **Data Migration:** Old system exports employee names in ALL CAPS — Flash Fill converts 500 names to Proper Case in one step.
- **HR Reports:** Create "Firstname Initial.Lastname" format (e.g. "A. Shah") for formal correspondence.
- **Customer Data:** Reformat imported phone numbers from 10-digit to 5+5 format for consistency.

3.4 — TRIM & CLEAN Functions

Theory

TRIM and CLEAN are the two most important data cleaning functions in Excel. Imported data almost always contains invisible characters that break formulas, lookups, and sorting.

TRIM — Remove Extra Spaces

`=TRIM(text)`

Removes: → all leading spaces (before text)
→ all trailing spaces (after text)
→ extra spaces between words (keeps single space)

Does NOT remove: → non-breaking spaces (from websites/HTML – use CLEAN or SUBSTITUTE)

TRIM Examples

Cell Contains	Formula	Result	Spaces Removed
" Aarav Shah"	=TRIM(A2)	"Aarav Shah"	Leading spaces removed
"Aarav Shah "	=TRIM(A2)	"Aarav Shah"	Trailing spaces removed
"Aarav Shah"	=TRIM(A2)	"Aarav Shah"	Double middle space → single
" Aarav Shah "	=TRIM(A2)	"Aarav Shah"	All three types removed

CLEAN — Remove Non-Printable Characters

=CLEAN(text)

Removes invisible control characters (ASCII 0-31): line breaks, carriage returns, null characters – common in data copied from PDFs, databases, and web pages.

Use BOTH together for thorough cleaning:

=TRIM(CLEAN(A2))

Combining TRIM with Other Functions

Formula	Purpose
=TRIM(A2)	Basic space removal
=TRIM(CLEAN(A2))	Remove spaces AND control characters (full clean)
=UPPER(TRIM(A2))	Clean spaces AND convert to UPPERCASE
=PROPER(TRIM(A2))	Clean spaces AND convert to Proper Case
=TRIM(LOWER(A2))	Clean spaces AND convert to lowercase
=IFERROR(VLOOKUP(TRIM(A2),...,...))	TRIM before VLOOKUP to prevent #N/A from trailing spaces
=SUBSTITUTE(TRIM(A2), " ", "")	Remove ALL spaces (including between words) — for codes/IDs

TIP Before any VLOOKUP or INDEX-MATCH, always TRIM both the lookup value AND the first column of the table. Most mysterious #N/A errors are caused by invisible spaces.

SUBSTITUTE — Remove Specific Characters

=SUBSTITUTE(text, old_text, new_text)

<code>=SUBSTITUTE(A2, " ", "")</code>	→ remove ALL spaces
<code>=SUBSTITUTE(A2, "-", "")</code>	→ remove all hyphens
<code>=SUBSTITUTE(A2, CHAR(160), "")</code>	→ remove non-breaking space (from HTML)
<code>=SUBSTITUTE(A2, CHAR(10), " ")</code>	→ replace line break with space

Industry Use Cases

- **VLOOKUP Failures:** Most #N/A errors are from spaces. `=VLOOKUP(TRIM(H2), EmpData, 2, FALSE)` fixes 90% of lookup failures.
- **Database Import:** ERP systems often add line breaks in description fields — CLEAN removes them for clean reports.
- **Web Data:** Copy-pasting from websites brings non-breaking spaces (CHAR 160) — SUBSTITUTE removes them.
- **Sorting Issues:** "Aarav" and " Aarav" sort differently. TRIM ensures consistent sorting.

3.5 — Find & Replace

Theory

Find & Replace is a powerful tool to locate values, text, or formats in a worksheet and optionally replace them with something else. Used correctly, it can fix thousands of inconsistencies in seconds.

Basic Find & Replace

<code>Ctrl + H</code>	→ opens Find & Replace dialog
<code>Ctrl + F</code>	→ opens Find only
Find what:	type what you want to find
Replace with:	type what you want to replace it with
Replace	→ replaces one instance at a time (preview each change)
Replace All	→ replaces all instances at once (use carefully!)

Advanced Find & Replace Options

STEP 1 Click "Options >>" in the dialog — Expands advanced settings

Option	What It Does	Example Use
Match Case	Find exact case only	Find "Sales" — ignores "sales" and "SALES"
Match Entire Cell Contents	Only matches cells with EXACTLY that text	Find "IT" — ignores "IT Department"
Look In: Values	Search displayed values (default)	Finds what you see
Look In: Formulas	Search formula text	Find all formulas containing "VLOOKUP"

Look In: Comments	Search cell comments only	Find annotated cells
Format button	Find or replace specific formatting	Find all bold cells, replace with red fill
Within: Workbook	Search all sheets at once	Find "Rohit" across every sheet in one search

Wildcard Find & Replace

Wildcards work in Find & Replace when searching Values:

* → matches any number of characters
? → matches exactly one character
~ → escapes a wildcard (find literal * or ?)

Examples:

"E00*" → finds E001, E002, E003, E0099 etc.
"E00?" → finds E001 to E009 (exactly 4 chars)
"*Shah" → finds all cells ending in "Shah"
"*Sales*" → finds cells containing "Sales" anywhere

Practical Cleaning Examples

Problem	Find What	Replace With	Result
Dept names inconsistent	"sales"	"Sales"	Standardise capitalisation
Remove unwanted prefix	"EMP-"	""	E.g. "EMP-001" → "001"
Replace old city name	"Bombay"	"Mumbai"	Update 500 rows instantly
Fix phone format	"_"	""	Remove hyphens from all phone numbers
Remove ₹ from numbers	"₹"	""	Convert "₹45000" to "45000" for calculations
Standardise status	"inactive"	"Inactive"	Fix case for filtering

NOTE "Replace All" cannot be undone if you have saved after. Always use Ctrl+Z immediately if the result is unexpected — or work on a copy of the data.

Industry Use Cases

- **Data Standardisation:** City names entered as "mumbai", "Mumbai", "MUMBAI" — Replace All to standardise to "Mumbai".
- **System Migration:** Old product codes prefixed "OLD-" need the prefix removed — Find "OLD-", replace with "".
- **Formula Audit:** Find all cells containing "SUM" in formulas — Look In: Formulas → helps audit a large model.
- **Bulk Update:** Manager changed — find all 80 rows with old manager name, replace with new name in one step.

3.6 — Data Validation

Theory

Data Validation prevents wrong data from being entered in the first place — solving data quality at the source rather than cleaning it later. You set rules on a cell: only numbers between 1 and 100, only dates after today, only values from a predefined list.

Types of Validation

Validation Type	What It Restricts	Business Example
Whole Number	Only integers within a range	Age: 18 to 65 Quantity: > 0
Decimal	Numbers with decimals within a range	Interest rate: 0.01 to 0.25
List	Only values from a predefined list	Department: Sales, HR, IT, Finance
Date	Only dates within a range	Join date: between 01-Jan-2000 and today
Time	Only times within a range	Meeting time: 09:00 to 18:00
Text Length	Only text within a character limit	Pincode: exactly 6 characters
Custom	Any formula-based rule	=AND(A2>0, MOD(A2,1)=0) — positive integers only

Step-by-Step — Create Dropdown List Validation

- STEP 1** Select the cells to validate — E.g. select C2:C100 (department column)
- STEP 2** Data → Data Validation — Opens the Data Validation dialog
- STEP 3** Settings tab → Allow: List — Choose List from the Allow dropdown
- STEP 4** Type the list or reference a range — Source: type "Sales,HR,IT,Finance" OR select a range like \$J\$2:\$J\$5
- STEP 5** Input Message tab (optional) — Add a helpful message shown when user clicks the cell
- STEP 6** Error Alert tab — Set the error message shown when invalid data is entered. Stop = blocks entry. Warning = allows with warning.
- STEP 7** Click OK — Dropdown arrows appear in the validated cells

Error Alert Styles

Style	Icon	Behaviour
Stop	 Red X	Blocks invalid entry completely — user MUST correct it
Warning	 Yellow !	Warns the user but allows them to proceed anyway
Information	 Blue i	Informs the user but always allows entry — mildest

Advanced Validation Examples

Formula-Based Validation — Only unique values allowed

Allow: Custom

Formula: =COUNTIF(\$A\$2:\$A\$100, A2) = 1

→ Blocks duplicate entries in the emp_id column

→ COUNTIF checks if this value already exists; = 1 means only this row has it

Dependent Dropdown — City changes based on State selected

1. Name ranges: Maharashtra = (Pune, Mumbai, Nagpur), Delhi = (Connaught Place, Dwarka)

2. State dropdown in B2 validated with a list: Maharashtra, Delhi, Goa

3. City dropdown in C2 validated with formula: =INDIRECT(B2)

→ Selecting "Maharashtra" in B2 auto-limits C2 to Pune, Mumbai, Nagpur

→ INDIRECT converts the text "Maharashtra" to the named range Maharashtra

Industry Use Cases

- **HR Forms:** Dropdown for Department, Grade, Location — prevents typos like "Salles" or "H R".
- **Finance:** Cost centre codes validated against a master list — no invalid codes can be entered.
- **Purchase Orders:** Quantity must be > 0 and a whole number — validation blocks negative or decimal values.
- **Data Collection:** Survey forms with date validation — join date cannot be before 1990 or after today.

3.7 — Conditional Formatting

Theory

Conditional Formatting automatically changes the appearance of a cell (fill color, font color, borders, icons) based on its value or a formula. It makes patterns, outliers, and trends instantly visible without any charts — the data highlights itself.

Types of Conditional Formatting

Type	What It Does	Business Example
Highlight Cell Rules	Format cells that meet a simple condition	Salary > 60000 → green fill
Top/Bottom Rules	Highlight top or bottom N values or %	Top 10% salaries highlighted in gold
Data Bars	In-cell bar chart proportional to value	Revenue bars inside salary column
Color Scales	2 or 3-colour gradient based on value range	Red-Yellow-Green heat map on marks
Icon Sets	Arrows, traffic lights, stars based on value	▲ green for growth, ▼ red for decline

New Rule (Custom Formula)	Any formula — most powerful option	Highlight entire row if status = Inactive
---------------------------	------------------------------------	---

How to Apply — Highlight Cell Rules

- STEP 1** Select the range to format — E.g. select E2:E11 (salary column)
- STEP 2** Home → Conditional Formatting — Opens the Conditional Formatting menu
- STEP 3** Highlight Cell Rules → Greater Than — Choose the appropriate rule type
- STEP 4** Enter the value and choose a format — Type 60000 → choose "Green Fill with Dark Green Text"
- STEP 5** Click OK — All cells with salary > 60000 are immediately highlighted green

Advanced — Custom Formula Rules

Custom formula rules are the most powerful. The formula applies to the FIRST cell in your selection — Excel extends it relatively to all other cells.

Example 1 — Highlight entire row if status = "Inactive"

1. Select entire data range: `$A$2:$G$11`
2. Home → Conditional Formatting → New Rule
3. "Use a formula to determine which cells to format"
4. **Formula:** `= $G2="Inactive"`
 - `$G` locks the column (always check column G)
 - `2` is relative (adjusts for each row)
5. Choose a format → Red fill
6. Click OK → All inactive employee rows turn red

Example 2 — Highlight salary below department average

Formula: `=E2<AVERAGEIF($C$2:$C$11,$C2,$E$2:$E$11)`

- Compares each salary to the AVERAGE of its own department
- Below-average earners in each department are highlighted automatically

Example 3 — Alternating row colours (without a Table)

Select range → New Rule → Formula:
=MOD (ROW () ,2)=0

- `MOD(ROW(),2)` returns 0 for even rows, 1 for odd rows
- Even rows get the formatting colour → creates zebra striping

Icon Sets and Data Bars Examples

Visual Type	Good For	Quick Steps
Data Bars	Comparing values quickly inside cells without a chart	Select range → CF → Data Bars → choose a color
3-Color Scale	Heat map of performance — red=low, yellow=mid, green=high	Select range → CF → Color Scales → Red-White-Green
Icon Set — Traffic Light	Status indicator for KPIs	Select range → CF → Icon Sets → 3 Traffic Lights
Icon Set — Arrows	Show growth direction	Select range → CF → Icon Sets → 3 Arrows (colored)

Manage & Clear Conditional Formatting

Home → Conditional Formatting → Manage Rules → view, edit, delete all rules
Home → Conditional Formatting → Clear Rules → From Selected Cells / Entire Sheet

Priority: rules at the TOP of the Manage Rules list take precedence.
Use "Stop If True" to prevent lower rules from applying after a match.

Industry Use Cases

- **Sales Dashboards:** Traffic light icons on KPI sheet — green if target achieved, red if missed.
- **Finance:** Color scale heat map on expense rows — immediately spots which months or departments overspent.
- **HR:** Highlight employees with tenure < 1 year in yellow (probation period tracking).
- **Data Quality:** Highlight blank cells in red across the entire data entry form — shows which fields are missing.

3.8 — Case Studies & Assignments

Case Study 1 — HR Data Cleaning Project

Scenario: Your company received an employee export from the old HR system. The file has 200 rows and many data quality issues. Clean it completely before importing into the new system.

Issues Found in the Raw Data

#	Issue Found	Tool to Fix
1	Employee names in ALL CAPS — e.g. "AARAV SHAH"	Flash Fill or =PROPER(TRIM(A2))
2	Extra spaces in names — " Sneh Mehta "	=TRIM(A2)
3	Department names inconsistent — "sales", "Sales", "SALES"	Find & Replace → standardise to "Sales"
4	47 duplicate employee IDs	Remove Duplicates → check emp_id column
5	City and pincode in one column — "Pune 411001"	Text to Columns → space delimiter

6	Salary column has ₹ symbol — "₹45,000"	Find & Replace ₹ with "" and , with ""
7	Status column has typos — "Active", "Inactiv"	Find & Replace to fix each typo
8	Dates in wrong format — "15/01/24" and "15-Jan-24" mixed	Text to Columns or Text formula to standardise
9	Non-printable characters in email column	=CLEAN(TRIM(A2))
10	No validation on Department — anyone can type anything	Data Validation → List dropdown

Assignment Steps

- **Step 1:** Download the raw data file (provided by trainer).
- **Step 2:** Make a backup copy of the original sheet (right-click tab → Move or Copy → tick "Create a copy").
- **Step 3:** Fix each issue in order using the appropriate tool from Chapter 3.
- **Step 4:** After cleaning, add Conditional Formatting to highlight any remaining blank cells in red.
- **Step 5:** Add Data Validation to the Department, City, and Status columns.
- **Step 6:** Sort the cleaned data by Department → Name alphabetically.
- **Step 7:** Save as "EmployeeData_Cleaned_YourName.xlsx".

Case Study 2 — Sales Data Dashboard Prep

Scenario: You received monthly sales data from 5 regional offices. Each office sent a separate file with slightly different column names and formats. Merge and clean them into one consistent dataset.

Cleaning Checklist

Check	Problem	Fix
✓ Remove Duplicates	Same transaction appears in two regional files	Remove Duplicates on Txn_ID column
✓ Standardise Columns	"Sale_Amount", "SaleAmt", "Amount" — 3 different names	Rename all to "Sale_Amount" via Find & Replace in headers
✓ Fix Number Format	Amount column has commas and ₹ — stored as text	Find & Replace then Text to Columns
✓ Standardise Dates	Some offices use DD/MM/YYYY, others MM-DD-YYYY	TEXT + DATE formula to standardise all to YYYY-MM-DD
✓ Validate Region Values	"West", "WEST", "west" all in Region column	Find & Replace → add Data Validation dropdown
✓ Flag Missing Data	Some rows have blank Sale_Amount	Conditional Formatting → highlight blanks in red
✓ Sort Final Data	Sort by Region → Date → Employee_ID	Data → Sort → multi-level sort

Practice Assignments

Assignment 1 — Basic (Chapter 1 Skills)

- Create an Employee Master Sheet with 15 employees across 3 departments.
- Format salary column with ₹#,##0 format.
- Apply AutoFill to generate employee IDs: EMP-001 to EMP-015.
- Sort by Department (A→Z) and within each department by Salary (high to low).
- Freeze the first row and first column.
- Set print area to the data only and configure headers to repeat on every page.
- Protect all formula cells. Leave name and salary cells editable.

Assignment 2 — Intermediate (Chapter 3 Skills)

- Import the provided dirty dataset (100 rows) with known quality issues.
- Remove 12 duplicate rows using Remove Duplicates.
- Split the "Full_Address" column into City and Pincode using Text to Columns.
- Clean the Employee_Name column using TRIM and PROPER.
- Standardise all 4 variations of department names using Find & Replace.
- Add dropdown Data Validation for Department, Status, and Region columns.
- Apply Conditional Formatting: rows with status="Inactive" → red, salary > avg → green, top 3 earners → gold icon.
- Write 3 SUMIFS formulas: total salary by dept, by city, and by status+dept combined.

Quick Reference — Chapter 3 Data Cleaning Tools

Problem	Best Tool	Keyboard / Location
Duplicate rows	Remove Duplicates	Data → Data Tools → Remove Duplicates
One column needs splitting	Text to Columns	Data → Text to Columns
Pattern transformation	Flash Fill	Ctrl + E or Data → Flash Fill
Extra spaces	TRIM function	=TRIM(A2) — formula
Non-printable characters	CLEAN function	=CLEAN(A2) — formula
Inconsistent text values	Find & Replace	Ctrl + H
Prevent future bad data	Data Validation	Data → Data Validation
Visually highlight patterns	Conditional Formatting	Home → Conditional Formatting

PRACTICE DATASETS — FILLED EXAMPLES

The following ready-to-use datasets have been added so students can perform the Excel exercises in the chapters without starting from empty tables.

Dataset 1 — Employee Master / Data Cleaning

emp_id	emp_name	dept	city	pincode	salary	status
E001	Aarav Shah	Sales	Pune	411001	45000	Active
E002	Sneha Mehta	HR	Mumbai	400001	52000	Active
E003	Rohit Verma	IT	Pune	411014	68000	Active
E004	Priya Patil	Finance	Nashik	422001	61000	Inactive
E005	Karan Joshi	Sales	Mumbai	400050	48000	Active
E006	Neha Kulkarni	IT	Pune	411038	72000	Active
E007	Vikas Singh	HR	Delhi	110001	55000	Active
E008	Anjali Rao	Finance	Pune	411005	64000	Active
E009	Manish Gupta	Sales	Nashik	422003	47000	Inactive
E010	Pooja Deshmukh	IT	Mumbai	400076	75000	Active

Dataset 2 — Sales Transactions / Dashboard Preparation

Txn_ID	Date	Region	Employee_ID	Product	Sale_Amount
TXN001	2026-01-05	North	E001	Laptop	65000
TXN002	2026-01-08	West	E005	Monitor	18000
TXN003	2026-01-12	South	E003	Keyboard	3500
TXN004	2026-01-18	East	E007	Printer	22000
TXN005	2026-02-02	West	E002	Laptop	72000
TXN006	2026-02-10	North	E004	Mouse	1800
TXN007	2026-02-15	South	E006	Monitor	21000
TXN008	2026-02-22	East	E008	Printer	24500

Dataset 3 — Dirty Data for Cleaning Practice

emp_id	emp_name	dept	city_pincode	salary	status
E011	AARAV SHAH	sales	Pune 411001	₹45,000	Active
E012	Sneha Mehta	SALES	Mumbai 400001	₹52,000	Active
E013	ROHIT VERMA	IT	Pune 411014	₹68,000	Inactiv
E011	AARAV SHAH	Sales	Pune 411001	₹45,000	Active
E014	Priya Patil	finance	Nashik 422001	₹61,000	Active
E015	Karan Joshi	HR	Delhi 110001	₹55,000	Active
E016	Neha Kulkarni	IT	Pune 411038	₹72,000	Active
E017	Vikas Singh	HR	Delhi 110002	₹49,000	Inactive
E018	Anjali Rao	FINANCE	Pune 411005	₹64,000	Active
E019	Manish Gupta	sales	Nashik 422003	₹47,000	inactive
E020	Pooja Deshmukh	IT	Mumbai 400076	₹75,000	Active
E020	Pooja Deshmukh	IT	Mumbai 400076	₹75,000	Active

CHAPTER 4 — PIVOT TABLES

Pivot Tables • Pivot Charts • Grouping • Slicers • Timelines

Why Pivot Tables?

A PivotTable is an Excel tool used to quickly summarize, analyze, and reorganize large datasets without changing the original source data. It can calculate totals, counts, averages, percentages, and other summaries by categories such as region, product, department, or month.

PivotTables are especially useful as the foundation for interactive reports and dashboards.

What a PivotTable Looks Like

SOURCE DATA

Region	Product	Sales	Profit
North	Laptop	65000	18000
West	Monitor	42000	11000
South	Laptop	69000	19000
East	Printer	56000	14000

Raw transactions

PIVOTTABLE

Region	Sum of Sales	Sum of Profit
East	56000	14000
North	65000	18000
South	69000	19000
West	42000	11000

Summarized automatically

Figure: Example of how raw Excel data is summarized in a PivotTable.

4.1 — Pivot Tables

Theory

A PivotTable summarizes a dataset by placing fields into Rows, Columns, Values, and Filters. Excel then calculates the required summary automatically.

Pivot Area	Purpose	Example
Rows	Creates categories down the report	Region
Columns	Creates categories across the report	Month
Values	Performs calculations	Sum of Sales
Filters	Filters the complete PivotTable	Department

Steps to Create a PivotTable

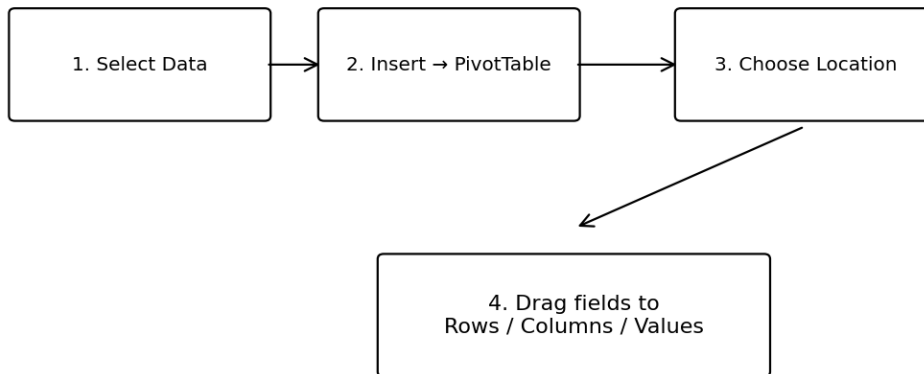


Figure: Practical steps for creating a PivotTable.

Creating a PivotTable

- 1. Prepare the data** — Use a clean table with one header row and no blank rows or columns.
- 2. Select the dataset** — Click any cell inside the source data.
- 3. Insert PivotTable** — Go to Insert → PivotTable.
- 4. Choose the source** — Confirm the Table/Range and choose where to place the PivotTable.
- 5. Arrange fields** — Drag fields into Rows, Columns, Values, and Filters.
- 6. Choose the calculation** — Use Sum, Count, Average, Max, Min, or another suitable calculation.

Example — Sales by Region

Region	Total Sales	Order Count
North	₹2,980,000	38
South	₹2,540,000	34
East	₹2,310,000	31
West	₹3,220,000	42

Changing Value Calculations

Calculation	Use
Sum	Total sales/revenue
Count	Number of transactions
Average	Average sales/order
Max	Highest transaction
Min	Lowest transaction

Value Field Settings → Summarize Values By → Sum

4.2 — Pivot Charts

Theory

A PivotChart is a chart connected to a PivotTable. When the PivotTable is filtered or rearranged, the PivotChart updates to reflect the selected data.

1. **Create a PivotTable** — Build the summary required for the analysis.
2. **Select the PivotTable** — Click anywhere inside the PivotTable.
3. **Insert PivotChart** — Go to PivotTable Analyze → PivotChart.
4. **Select a chart** — Choose Column, Bar, Line, Pie, or another appropriate type.
5. **Format the chart** — Add a clear title and remove unnecessary visual elements.

Business Question	Recommended PivotChart
Which region has the highest sales?	Column/Bar
How are sales changing monthly?	Line
What is the category share?	Pie/Doughnut
Which products rank highest?	Bar

Tip

PivotCharts are especially useful when combined with Slicers and Timelines because users can interactively change the chart.

Steps to Create a PivotTable

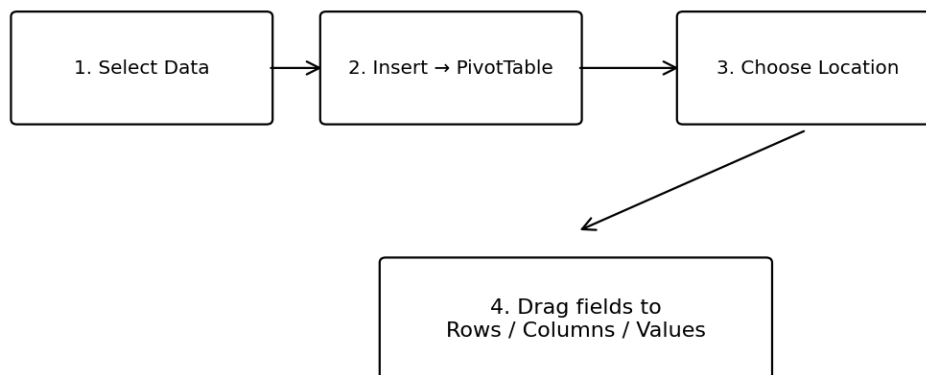


Figure: PivotTable workflow reminder before applying grouping.

4.3 — Grouping

Theory

Grouping combines PivotTable items into meaningful categories. It is commonly used to group dates into Years, Quarters, Months, or Days and to group numeric values into ranges.

Date Grouping

1. **Add a date field** — Place Date in Rows or Columns.
2. **Right-click a date** — Choose Group.
3. **Select periods** — Choose Years, Quarters, Months, or Days.
4. **Confirm** — Excel creates grouped date fields in the PivotTable.

Date	Grouped Result
05-Jan-2026	Jan 2026
18-Jan-2026	Jan 2026
02-Feb-2026	Feb 2026
15-Mar-2026	Mar 2026

Number Grouping

Numeric values can be grouped into intervals such as 0–9,999; 10,000–19,999; and so on. This is useful for salary ranges, sales ranges, age groups, and transaction values.

Sales Amount	Sales Range
₹4,500	₹0–₹9,999
₹12,500	₹10,000–₹19,999
₹27,000	₹20,000–₹29,999
₹43,000	₹40,000–₹49,999

What a Slicer Looks Like

REGION SLICER

All

North

West

Click a button → report filters automatically

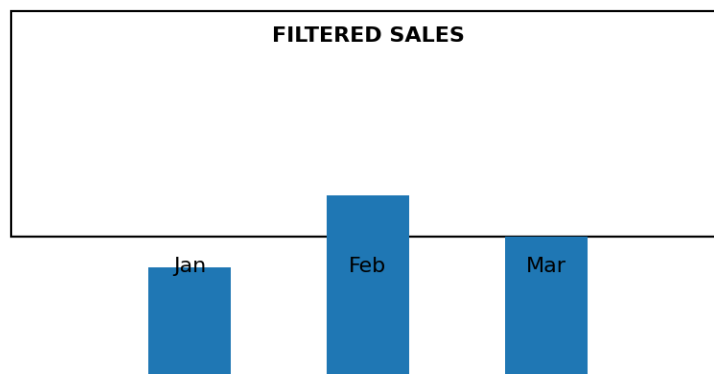


Figure: Example of a Slicer used to filter a PivotTable/report.

4.4 — Slicers

Theory

A Slicer is a visual filtering control that lets users filter a PivotTable or PivotChart by clicking buttons. Slicers make reports easier to use than traditional filter dropdowns.

1. **Select the PivotTable** — Click anywhere inside it.
2. **Insert Slicer** — Go to PivotTable Analyze → Insert Slicer.
3. **Choose fields** — Select fields such as Region, Product, or Salesperson.
4. **Use the slicer** — Click a button to filter the report.
5. **Clear the filter** — Use the clear-filter icon on the slicer.

Slicer Selection	Dashboard Result
All Regions	Displays all regions
West	Shows only West data
Laptop	Shows only Laptop records
West + Laptop	Shows Laptop sales in West

Connecting One Slicer to Multiple PivotTables

1. **Create the PivotTables** — Use the same source dataset.
2. **Select the Slicer** — Click the slicer.
3. **Open Report Connections** — Use Slicer → Report Connections.
4. **Select PivotTables** — Tick the PivotTables that should respond.
5. **Test the slicer** — Click different items and confirm all connected reports update.

What a Timeline Looks Like

DATE TIMELINE

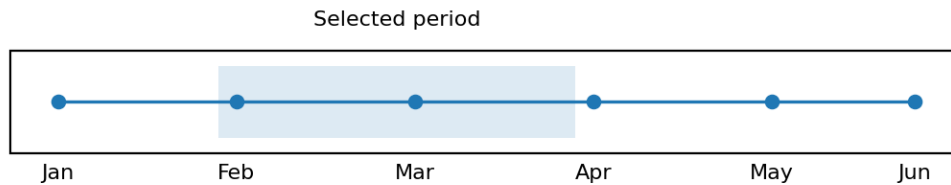


Figure: Example of a Timeline used for date-based filtering.

4.5 — Timelines

Theory

A Timeline is a visual date filter used with PivotTables. It allows users to filter dates by Years, Quarters, Months, or Days.

- 1. Select the PivotTable** — Click anywhere inside the PivotTable.
- 2. Insert Timeline** — Go to PivotTable Analyze → Insert Timeline.
- 3. Select Date** — Choose the date field.
- 4. Choose the time level** — Select Years, Quarters, Months, or Days.
- 5. Drag the selection** — Move the timeline handle to the desired period.

Timeline Level	Example Selection	Use
Years	2026	Year-over-year analysis
Quarters	Q1 2026	Quarterly performance
Months	Jan–Mar 2026	Monthly analysis
Days	05–15 Jan 2026	Detailed date filtering

Slicer vs Timeline

Slicer → best for categorical fields such as Region, Product, Department, or Salesperson.
Timeline → best for date fields and time-based filtering.

4.6 — Complete PivotTable Workflow

- 1. Clean the source data** — Remove duplicates, fix headers, and ensure dates/numbers are valid.
- 2. Convert to an Excel Table** — Use Ctrl+T so the source can expand when new records are added.

3. Create PivotTables — Build summaries for sales, orders, profit, and category performance.

4. Create PivotCharts — Turn important summaries into visuals.

5. Add Slicers — Allow users to filter by important categories.

6. Add Timeline — Allow users to filter by date.

7. Format the report — Use consistent number formats, titles, and alignment.

8. Refresh — Refresh PivotTables whenever source data changes.

4.7 — Practice Dataset

Txn_ID	Date	Region	Product	Salesperson	Sale_Amount	Profit
P001	2026-01-05	North	Laptop	Aarav	65000	18000
P002	2026-01-08	West	Monitor	Sneha	42000	11000
P003	2026-01-12	South	Keyboard	Rohit	18000	5000
P004	2026-01-18	East	Printer	Priya	52000	12000
P005	2026-02-02	West	Laptop	Karan	72000	21000
P006	2026-02-10	North	Mouse	Neha	15000	4000
P007	2026-02-15	South	Monitor	Vikas	48000	13000
P008	2026-02-22	East	Printer	Anjali	56000	14000
P009	2026-03-03	West	Laptop	Manish	81000	24000
P010	2026-03-09	North	Monitor	Pooja	46000	12000
P011	2026-03-16	South	Laptop	Aarav	69000	19000
P012	2026-03-25	East	Keyboard	Sneha	22000	6000

Assignment

Task	Expected Output
PivotTable 1	Sales by Region
PivotTable 2	Sales by Product
PivotTable 3	Monthly Sales
PivotChart	Sales by Region
Grouping	Group Date by Month
Slicer	Filter by Region and Product
Timeline	Filter by Month
Final Report	Interactive PivotTable report

CHAPTER 5 — DASHBOARDS

Steps to Create an Interactive Dashboard

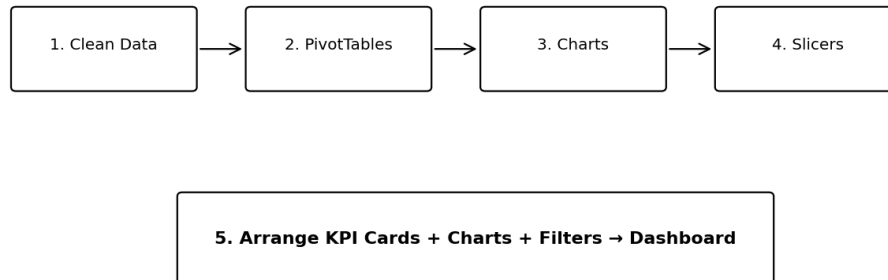


Figure: Practical workflow for building an interactive dashboard.

KPI Cards • Interactive Dashboards • Charts • Layout Design • Dynamic Dashboards

Why Dashboards?

A dashboard converts raw data into a visual summary that helps a business understand performance quickly.

A good dashboard answers three questions: What is happening? Why is it happening? What action should be taken?

Excel dashboards commonly combine KPI cards, charts, tables, slicers, timelines, formulas, and conditional formatting.

5.1 — KPI Cards

Theory

A KPI (Key Performance Indicator) is a measurable value used to track business performance against a target. A KPI card displays one important metric prominently so a user can understand it at a glance.

KPI	Meaning	Example	Business Question
Total Sales	Total revenue generated	₹12,50,000	How much did we sell?
Total Orders	Number of completed orders	1,248	How many orders were received?
Average Order Value	Sales divided by orders	₹1,002	What is the average order size?
Profit	Revenue minus cost	₹3,10,000	How much did we earn?
Profit Margin	Profit as % of sales	24.8%	How efficiently are we earning?
Target Achievement	Actual ÷ Target	108%	Did we meet the target?

Common KPI Formulas

```
=SUM(Sales[Sale_Amount])
=COUNTA(Sales[Txn_ID])
=SUM(Sales[Sale_Amount])/COUNTA(Sales[Txn_ID])
=SUM(Sales[Profit])
=SUM(Sales[Profit])/SUM(Sales[Sale_Amount])
=Actual/Target
```

Building a KPI Card

1. **Prepare the source value** — Create a formula cell that returns the KPI.
2. **Create the label** — Use a short title such as TOTAL SALES or PROFIT MARGIN.
3. **Format the value** — Use ₹, %, or number formatting appropriate to the metric.
4. **Add comparison** — Show target, previous period, or variance below the main KPI.
5. **Apply visual emphasis** — Use a large font and consistent card formatting.

Best Practice

Keep KPI cards simple. One card should communicate one metric.
Use the same size and alignment for all KPI cards.
Do not use too many colors. Reserve attention-grabbing formatting for exceptions.

KPI Card Example

Card	Value	Comparison	Interpretation
TOTAL SALES	₹12,50,000	+8.4% vs last month	Sales are increasing
ORDERS	1,248	+5.1%	Order volume improved
PROFIT	₹3,10,000	+11.2%	Profit grew faster than sales
MARGIN	24.8%	+0.7 pp	Profitability improved

5.2 — Interactive Dashboards

Theory

An interactive dashboard allows users to change the view without rebuilding the report. In Excel, interactivity can be created with Tables, PivotTables, PivotCharts, Slicers, Timelines, dropdowns, and formulas.

Main Interactive Controls

Control	Purpose	Example
Slicer	Filters a PivotTable/PivotChart visually	Region = West
Timeline	Filters date fields by period	Jan–Mar 2026
Dropdown	Lets the user select a category	Select Department
PivotTable	Summarises data interactively	Sales by Region
PivotChart	Visualises PivotTable results	Monthly sales chart

Dashboard Flow

1. **Prepare clean data** — Use one row per transaction and consistent column names.
2. **Convert data to a Table** — Select the dataset → Ctrl+T.
3. **Create PivotTables** — Summarise sales, profit, orders, and categories.
4. **Create PivotCharts** — Choose charts that answer specific questions.
5. **Insert Slicers** — Filter by Region, Product, Department, or Salesperson.
6. **Insert Timeline** — Add date-based filtering where a date field is available.
7. **Arrange the dashboard** — Place KPIs at the top and detailed visuals below.

Interactive Dashboard Example

User Selection	KPI Result	Chart Result
All Regions	Total sales for all regions	Monthly sales for all regions

West	Only West-region sales	West monthly trend
Product = Laptop	Laptop sales	Laptop sales by region
Month = Jan 2026	January KPIs	January category comparison

5.3 — Charts

Choosing the Right Chart

Chart	Best Use	Avoid When
Column	Compare categories	There are hundreds of categories
Bar	Compare long category names	You need a time-series trend
Line	Show trend over time	Categories are unrelated
Pie/Doughnut	Show simple share of a whole	There are many categories
Combo	Compare metrics with different scales	The message is already simple
Scatter	Study relationship between two numeric variables	The variables are categorical

Dashboard Chart Rules

- Use a clear title that explains what the chart shows.
- Keep unnecessary gridlines and decoration to a minimum.
- Sort categories when ranking is important.
- Use data labels only when they improve readability.
- Do not use a 3-D chart when a simple 2-D chart communicates the data better.

Example — Monthly Sales Trend

Month	Sales	Target
Jan	₹8,20,000	₹8,00,000
Feb	₹8,75,000	₹8,50,000
Mar	₹9,10,000	₹9,00,000
Apr	₹9,60,000	₹9,25,000
May	₹10,20,000	₹9,75,000
Jun	₹10,80,000	₹10,25,000

5.4 — Layout Design

Recommended Dashboard Layout

Area	Content
Top row	Dashboard title, reporting period, last refresh date
KPI row	4–6 KPI cards
Main visual area	Trend chart and comparison chart
Secondary area	Category/region chart and detail table
Filter area	Slicers, dropdowns, timeline
Footer	Notes, definitions, data source

Dashboard Design Checklist

One-screen view where practical
 Consistent fonts and spacing
 Clear visual hierarchy
 Important KPIs at the top
 Filters grouped together
 Charts aligned to a common grid
 No unnecessary merged cells in the source data

5.5 — Dynamic Dashboards

Theory

A dynamic dashboard updates when new data is added or when the user changes a selection. Excel Tables, PivotTables, dynamic formulas, and named ranges can help keep dashboard outputs current.

Dynamic Techniques

Technique	How It Helps
Excel Table	Automatically expands when rows are added
PivotTable Refresh	Updates summary calculations from source data
Slicer	Changes the displayed dashboard view
Dynamic formulas	Return results based on user selections
Named ranges	Provide reusable references for dashboard elements

Example — Dynamic Selected Region

```
=SUMIFS(Sales[Sale_Amount],Sales[Region],B2)
=COUNTIFS(Sales[Region],B2)
=AVERAGEIFS(Sales[Sale_Amount],Sales[Region],B2)
```

If B2 contains the selected region, these formulas automatically return the corresponding sales, order count, and average sale.

Industry Use Cases

- Management Dashboard: monitor sales, profit, orders, and target achievement.
- HR Dashboard: headcount, department strength, attrition, and attendance.
- Finance Dashboard: revenue, expenses, budget, and variance.
- Operations Dashboard: production, defects, delivery status, and inventory.

5.6 — Project Work / Assignments

Project — Sales Performance Dashboard

Build a one-page interactive Excel dashboard using the Sales Dashboard dataset provided at the end of this chapter.

Requirement	Expected Output
4 KPI cards	Sales, Orders, Profit, Profit Margin
Trend chart	Monthly Sales vs Target
Comparison chart	Sales by Region
Category chart	Sales by Product
Interactive filter	Region slicer or dropdown
Date filter	Timeline or month selector
Conditional formatting	Highlight below-target KPIs
Professional layout	Title, cards, charts, filters, notes

Dataset — Dashboard Practice

Txn_ID	Date	Region	Product	Salesperson	Sale_Amount	Profit
T001	2026-01-05	North	Laptop	Aarav	65000	18000
T002	2026-01-08	West	Monitor	Sneha	42000	11000
T003	2026-01-12	South	Keyboard	Rohit	18000	5000
T004	2026-01-18	East	Printer	Priya	52000	12000
T005	2026-02-02	West	Laptop	Karan	72000	21000
T006	2026-02-10	North	Mouse	Neha	15000	4000
T007	2026-02-15	South	Monitor	Vikas	48000	13000
T008	2026-02-22	East	Printer	Anjali	56000	14000
T009	2026-03-03	West	Laptop	Manish	81000	24000
T010	2026-03-09	North	Monitor	Pooja	46000	12000
T011	2026-03-16	South	Laptop	Aarav	69000	19000
T012	2026-03-25	East	Keyboard	Sneha	22000	6000

CHAPTER 6 — DATA ANALYSIS

Comparison • Trend • Percentage • Variance • Top/Bottom • Descriptive Statistics

Purpose of Data Analysis

Data analysis turns cleaned data into meaningful information for decision-making.

The objective is not only to calculate numbers but to compare, explain patterns, identify exceptions, and communicate findings.

6.1 — Comparison Analysis

Theory

Comparison analysis evaluates two or more categories, periods, products, regions, employees, or targets. It answers questions such as which region performed better, which product generated more profit, or whether this month exceeded last month.

Comparison	Formula / Method	Example
Category vs Category	Compare totals or averages	North sales vs West sales
Actual vs Target	Actual - Target	₹9.5L - ₹9L = ₹0.5L
Current vs Previous	Current - Previous	March sales - February sales
Group Average	AVERAGEIF/AVERAGEIFS	Average salary by department

Example

```
=SUMIFS(Sales[Sales_Amount],Sales[Region],A2)  
=SUMIFS(Sales[Profit],Sales[Region],A2)  
=B2-C2
```

Industry Use Cases

- Compare regional sales performance.
- Compare product profitability.
- Compare actual expenses with budget.
- Compare employee performance against targets.

6.2 — Trend Analysis

Theory

Trend analysis studies how a metric changes over time. A trend can be upward, downward, stable, seasonal, or irregular.

Trend Type	Meaning	Example
Upward	Values generally increase	Sales rise from Jan to Jun
Downward	Values generally decrease	Complaints decline each month
Stable	Values remain within a narrow range	Monthly headcount stays similar
Seasonal	Pattern repeats at regular periods	Higher sales every festive season

Useful Excel Tools

- Line charts for visual trends.
- MONTH/YEAR functions for grouping dates.
- AVERAGE and moving averages for smoothing.
- TREND/FORECAST for modelling future values.

Monthly Trend Example

Month	Sales	Growth %
Jan	820000	—
Feb	875000	6.71%
Mar	910000	4.00%
Apr	960000	5.49%
May	1020000	6.25%
Jun	1080000	5.88%

= (B3-B2)/B2
Format the result as Percentage.

6.3 — Percentage Analysis

Common Percentage Measures

Measure	Formula	Meaning
Percentage of Total	Part / Total	Share of total sales
Growth %	(Current - Previous) / Previous	Rate of change
Achievement %	Actual / Target	Target attainment
Margin %	Profit / Revenue	Profitability
Contribution %	Category / Total	Category contribution

Examples

=B2/SUM(\$B\$2:\$B\$6)
= (C2-B2)/B2
=B2/C2
=D2/B2

Important

Always check the denominator. A percentage is meaningful only when the base being compared is correct.
Format percentage results using an appropriate number of decimal places.

6.4 — Variance Analysis

Theory

Variance measures the difference between an actual value and a planned, budgeted, standard, or previous value.

Variance Type	Formula	Interpretation
Absolute Variance	Actual - Budget	Positive means actual is above budget
Variance %	(Actual-Budget)/Budget	Difference relative to budget
Favourable/Unfavourable	Depends on metric	Higher sales may be favourable; higher cost may be unfavourable

Example — Budget vs Actual

Department	Budget	Actual	Variance	Variance %
Sales	900000	960000	60000	6.67%
Marketing	250000	230000	-20000	-8.00%
Operations	500000	540000	40000	8.00%
HR	180000	175000	-5000	-2.78%

=C2-B2
= (C2-B2)/B2

Industry Use Cases

- Monthly expense variance against budget.
- Sales actual vs sales target.
- Production actual vs planned quantity.
- Inventory actual vs expected stock.

6.5 — Top/Bottom Analysis

Theory

Top/Bottom analysis identifies the highest or lowest performers. It is useful for prioritisation, ranking, and exception management.

Requirement	Excel Method
Top 5 salespeople	SORT/SORTBY or LARGE
Bottom 3 products	SORT/SORTBY or SMALL
Highest sale	MAX
Lowest sale	MIN
Rank each employee	RANK.EQ
Return top values	LARGE

Formula Examples

```
=RANK.EQ(B2,$B$2:$B$11,0)  
=LARGE($B$2:$B$11,1)  
=SMALL($B$2:$B$11,3)
```

Example — Sales Ranking

Salesperson	Sales	Rank
Pooja	750000	1
Karan	690000	2
Neha	645000	3
Aarav	610000	4
Sneha	580000	5
Rohit	520000	6

6.6 — Descriptive Statistics

Theory

Descriptive statistics summarise the main characteristics of a dataset. Common measures are count, mean, median, mode, minimum, maximum, range, standard deviation, and quartiles.

Measure	Excel Function	Meaning
Count	COUNT	Number of numeric observations
Average / Mean	AVERAGE	Arithmetic average
Median	MEDIAN	Middle value after sorting
Mode	MODE.SNGL	Most frequently occurring value
Minimum	MIN	Smallest value
Maximum	MAX	Largest value
Range	MAX-MIN	Spread between largest and smallest
Standard Deviation	STDEV.S	Typical spread around the sample mean
Quartile	QUARTILE.INC	Position within the distribution

Example

```
=COUNT(B2:B101)  
=AVERAGE(B2:B101)  
=MEDIAN(B2:B101)
```

```

=MODE.SNGL(B2:B101)
=MIN(B2:B101)
=MAX(B2:B101)
=MAX(B2:B101)-MIN(B2:B101)
=STDEV.S(B2:B101)

```

Interpretation Example

Suppose the average monthly sale is ₹80,000 and the median is ₹65,000. The higher mean may indicate that a few very large sales are pulling the average upward. Median can therefore be useful when the data contains extreme values.

6.7 — Case Study & Assignment

Analyse the Sales Analysis dataset and prepare a management analysis sheet.

Task	Expected Analysis
Comparison	Sales by region and product
Trend	Monthly sales trend
Percentage	Each region's share of total sales
Variance	Actual sales vs target
Top/Bottom	Top 3 products and bottom 3 products
Statistics	Mean, median, minimum, maximum, standard deviation

Dataset — Data Analysis Practice

Month	Region	Product	Actual_Sales	Target_Sales
Jan	North	Laptop	820000	800000
Jan	West	Monitor	760000	780000
Jan	South	Printer	690000	650000
Jan	East	Keyboard	610000	600000
Feb	North	Laptop	875000	850000
Feb	West	Monitor	820000	800000
Feb	South	Printer	720000	700000
Feb	East	Keyboard	650000	620000
Mar	North	Laptop	910000	900000
Mar	West	Monitor	880000	850000
Mar	South	Printer	760000	740000
Mar	East	Keyboard	690000	670000

CHAPTER 7 — FORECASTING

Steps to Create a Sales Forecast

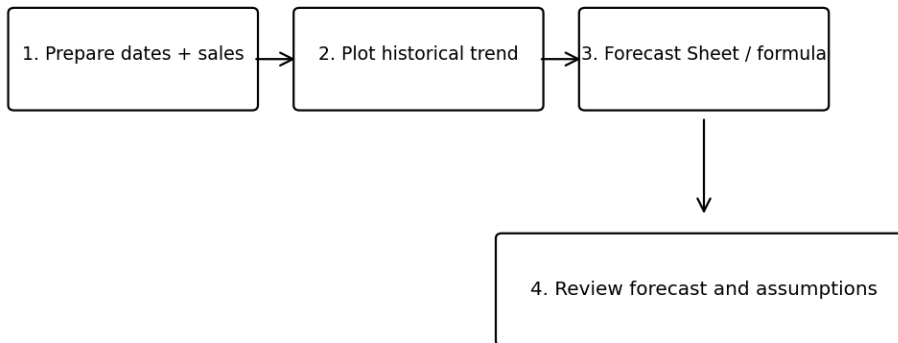


Figure: Practical workflow for creating a sales forecast.

Forecast Sheet • FORECAST • FORECAST.LINEAR • TREND • Moving Average • Sales Forecasting

What is Forecasting?

Forecasting uses historical data to estimate future values.

Excel forecasting tools can help estimate future sales, demand, revenue, expenses, inventory requirements, and other time-based measures.

A forecast is an estimate, not a guarantee. Results depend on data quality, historical patterns, and business conditions.

7.1 — Forecast Sheet

Theory

Excel's Forecast Sheet creates a forecast from historical time-series data. It can generate a forecast table and chart and is useful when dates and corresponding numeric values are available.

Requirements

- A date/time column arranged in chronological order.
- A corresponding numeric value column such as sales or demand.
- Consistent time intervals give better results.

Step-by-Step

1. **Prepare historical data** — Keep dates in one column and values in the next column.
2. **Select both columns** — Include the historical date and value range.
3. **Open Forecast Sheet** — Go to Data → Forecast Sheet.
4. **Choose chart type** — Use a line chart for most time-series forecasts.

5. Set the forecast end date — Choose how far into the future Excel should forecast.

6. Review options — Check seasonality, confidence interval, and timeline settings.

7. Create Forecast — Excel creates a new worksheet containing the forecast.

Setting	Purpose
Forecast End	Defines the last future period to estimate
Confidence Interval	Shows a likely range around the forecast
Seasonality	Controls repeating seasonal patterns
Timeline	Controls the time sequence used by the forecast

7.2 — FORECAST Function

Theory

The FORECAST function estimates a future y-value using existing x and y values and linear regression.

```
=FORECAST(x, known_y_values, known_x_values)
```

Example

Month Number (X)	Sales (Y)
1	500000
2	540000
3	575000
4	610000
5	655000
6	690000

```
=FORECAST(7,B2:B7,A2:A7)
```

The result estimates sales for month 7.

Remember

X represents the independent variable, such as month number.

Y represents the value being forecast, such as sales.

7.3 — FORECAST.LINEAR

Theory

FORECAST.LINEAR performs the same type of linear forecasting using a linear regression relationship between known x and y values. It is the modern Excel function name for linear forecast calculations.

```
=FORECAST.LINEAR(x, known_y_values, known_x_values)
```

```
=FORECAST.LINEAR(7,B2:B7,A2:A7)
```

Function	Use
FORECAST	Legacy function for linear forecasting
FORECAST.LINEAR	Modern function for linear forecasting

7.4 — TREND

Theory

TREND returns values along a linear trend line. It can be used to estimate future values and also to understand the underlying direction of a dataset.

```
=TREND(known_y_values, known_x_values, new_x_values)  
=TREND(B2:B7,A2:A7,A8:A10)
```

Example

Month	Actual Sales	Trend / Estimated
1	500000	500000
2	540000	540000
3	575000	580000
4	610000	620000
5	655000	660000
6	690000	700000
7	—	740000
8	—	780000

FORECAST.LINEAR vs TREND

Function	Output	Best For
FORECAST.LINEAR	One or more predicted y values	Forecasting a specific future x
TREND	Trend-line values for one or more x values	Generating a series of trend estimates

7.5 — Moving Average

Theory

A moving average smooths short-term fluctuations by averaging a fixed number of consecutive observations. It is useful when the underlying trend is more important than individual month-to-month noise.

Month	Sales	3-Month Moving Average
Jan	500000	—
Feb	540000	—
Mar	575000	538333
Apr	610000	575000
May	655000	613333
Jun	690000	651667
Jul	735000	693333

```
=AVERAGE(B2:B4)  
=AVERAGE(B3:B5)  
=AVERAGE(B4:B6)
```

How to Create a Moving Average

1. **Choose the period** — For example, use 3 months for a 3-month moving average.
2. **Calculate the first average** — Average the first three observations.
3. **Move one period forward** — Drop the oldest observation and include the next one.
4. **Copy the formula down** — Continue until the dataset ends.
5. **Create a line chart** — Compare actual values with the moving average.

Interpretation

A shorter moving-average period reacts faster to changes but is less smooth.
A longer period is smoother but may react slowly to sudden changes.

7.6 — Sales Forecasting

Forecasting Workflow

Step	Action
1	Collect historical monthly/weekly sales
2	Clean missing, duplicate, and incorrect values
3	Check the time sequence
4	Visualise the historical trend
5	Calculate a moving average
6	Use FORECAST.LINEAR or Forecast Sheet where appropriate
7	Compare forecast with business targets
8	Document assumptions and limitations

Sales Forecast Dataset

Month	Actual_Sales
2025-01-01	500000
2025-02-01	540000
2025-03-01	575000
2025-04-01	610000
2025-05-01	655000
2025-06-01	690000
2025-07-01	735000
2025-08-01	760000
2025-09-01	805000
2025-10-01	850000
2025-11-01	900000
2025-12-01	960000
2026-01-01	920000
2026-02-01	975000
2026-03-01	1020000
2026-04-01	1080000
2026-05-01	1120000
2026-06-01	1180000

Practice — Forecast the Next 6 Months

1. Add a Month_Number column from 1 to 18.
2. Plot Month vs Actual Sales using a Line Chart.
3. Calculate a 3-month moving average.
4. Use FORECAST.LINEAR to estimate months 19–24.
5. Create a comparison table: Actual/Forecast.
6. Calculate the expected percentage increase from month 18 to month 24.

Useful Forecast Formulas

```
=FORECAST.LINEAR(19,$B$2:$B$19,$C$2:$C$19)
=TREND($B$2:$B$19,$C$2:$C$19,C20:C25)
=AVERAGE(B17:B19)
```

Forecasting Cautions

- Do not forecast from dirty or incomplete data.

- A strong historical trend does not guarantee the same future conditions.
- Seasonality, promotions, market changes, and unusual events can affect actual sales.
- Always compare forecasts with business knowledge and explain important assumptions.

7.7 — Project Work / Assignments

Project — Sales Forecasting Model

Requirement	Expected Output
Historical data	18 months of sales
Trend chart	Actual monthly sales
Moving average	3-month moving average
Linear forecast	Forecast months 19–24
Forecast Sheet	Create Excel Forecast Sheet
Comparison	Historical vs forecast values
Management insight	Write 3 conclusions from the forecast

Student Questions

- Which month has the highest historical sales?
- Is the overall sales trend increasing or decreasing?
- How does the moving average differ from actual sales?
- What is the forecasted sales value for month 24?
- What assumptions might make the forecast inaccurate?
- If the company target is ₹12,00,000 per month, which forecast months meet the target?

7.8 — Quick Reference

Topic	Formula / Tool	Purpose
Forecast Sheet	Data → Forecast Sheet	Create a time-series forecast
FORECAST	=FORECAST(x,y,x)	Linear forecast
FORECAST.LINEAR	=FORECAST.LINEAR(x,y,x)	Modern linear forecast
TREND	=TREND(y,x,new_x)	Generate trend-line estimates
Moving Average	=AVERAGE(range)	Smooth short-term fluctuations
Sales Forecast	Combine trend + forecast + business assumptions	Estimate future sales

CHAPTER 8 — FORMS IN EXCEL

Data Form • Developer Tab • Form Controls • Interactive Forms

Chapter Overview

- Learn how Excel forms and controls make worksheets interactive and user-friendly.
- Understand how users can select, enter, increase, decrease, and control values using Form Controls.
- Build practical interactive forms using Check Boxes, Option Buttons, Group Boxes, Combo Boxes, Spin Buttons, and Scroll Bars.

8.1 — Data Form

Theory

A Data Form in Excel provides a simple interface for entering, editing, searching, and deleting records from a structured list. It allows users to work with one record at a time instead of scrolling across many columns.

When is a Data Form Useful?

Situation	Benefit
Large number of columns	Enter one record without horizontal scrolling
Data entry	Makes adding new records easier
Record editing	Quickly modify existing records
Searching	Find records using criteria
Deleting	Remove selected records

Steps to Use a Data Form

1. **Prepare the dataset** — Create a table with clear column headers.
2. **Add Form command** — Add the Form command to the Quick Access Toolbar.
3. **Select the dataset** — Click any cell inside the table.
4. **Open Form** — Click the Form command.
5. **Enter a record** — Use New to add a record.
6. **Find records** — Use Criteria and Find Prev/Find Next.
7. **Edit or delete** — Modify values or use Delete.

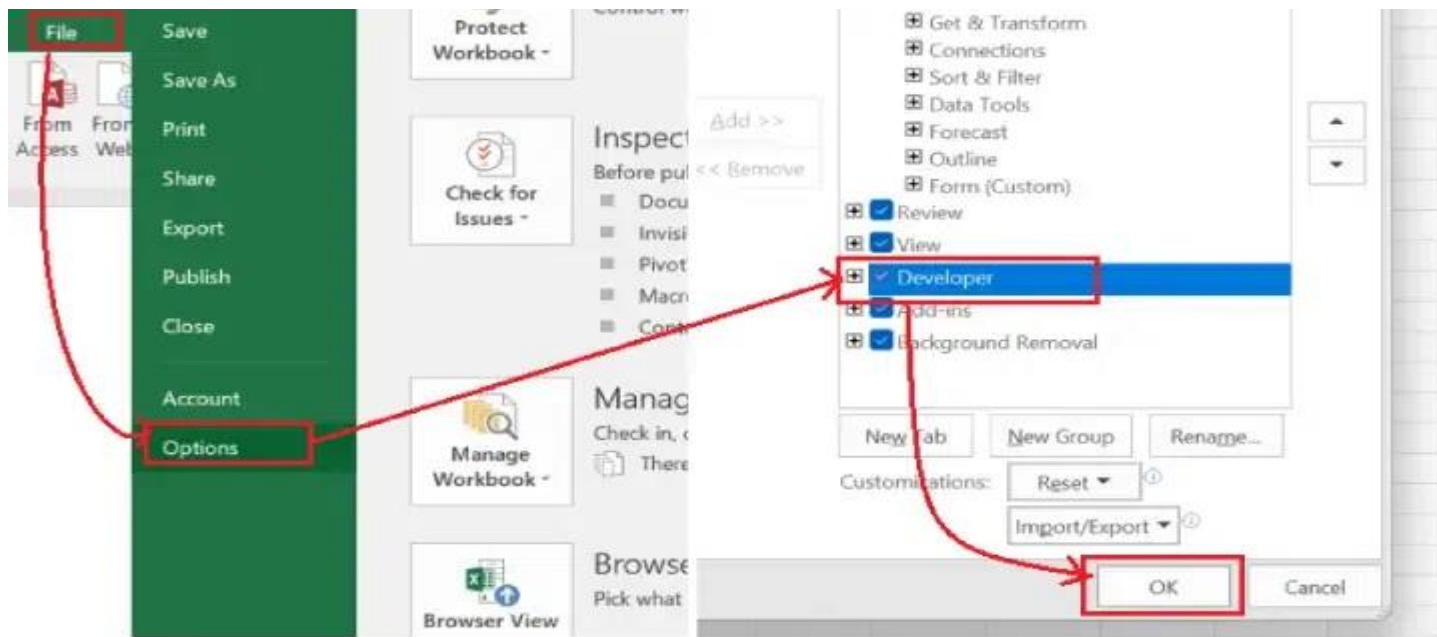
Example Dataset for Data Form

- Student_ID | Student_Name | Course | City | Mobile | Fee_Paid
- S101 | Rahul Sharma | Data Analytics | Pune | 9876543210 | Yes
- S102 | Priya Patil | Excel | Mumbai | 9876501234 | No
- S103 | Aman Verma | Power BI | Nashik | 9876123456 | Yes

8.2 — Developer Tab

Theory

The Developer Tab contains advanced Excel tools used for automation, controls, macros, VBA, and interactive form creation. It must be enabled before inserting Form Controls.



How to Enable Developer Tab

1. **Open Excel Options** — Go to File → Options.
2. **Select Customize Ribbon** — Open ribbon customization settings.
3. **Find Developer** — Locate Developer in the Main Tabs list.
4. **Enable it** — Tick the Developer checkbox.
5. **Click OK** — The Developer tab appears on the ribbon.

Developer Tool	Purpose
Visual Basic	Open VBA editor
Macros	Run or manage macros
Insert	Add Form Controls and ActiveX Controls
Design Mode	Modify control properties
Properties	Configure controls
Add-ins	Manage Excel add-ins

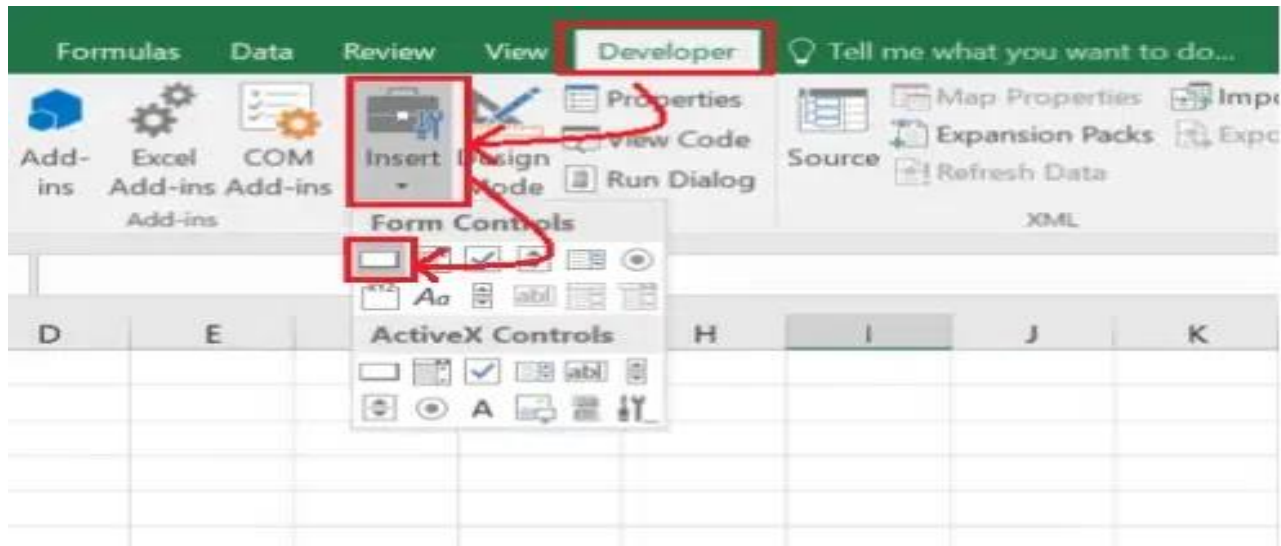
8.3 — Form Controls

Theory

Form Controls are interactive objects that can be inserted into a worksheet. They allow users to provide input by clicking buttons, selecting options, or changing values.

Form Control	Purpose	Example
Check Box	Select multiple choices	Choose multiple products
Option Button	Select one option	Choose payment mode
Group Box	Organize related controls	Separate option groups
Combo Box	Select from dropdown	Choose department
Spin Button	Increase/decrease number	Select quantity
Scroll Bar	Change linked value	Interactive report

How to Insert Form Controls



1. **Open Developer Tab** — Click Developer on the ribbon.
2. **Click Insert** — Open the Controls section.
3. **Choose Form Controls** — Select the required control.
4. **Draw the control** — Click and drag on the worksheet.
5. **Format Control** — Right-click and choose Format Control.
6. **Link the control** — Connect it to a worksheet cell.

8.4 — Check Box

Theory

A Check Box allows the user to select or deselect an option. Multiple Check Boxes can be selected because each option works independently.

Feature	Example
Multiple selection	Laptop + Monitor + Keyboard
Linked cell output	TRUE or FALSE
Common use	Product selection, task completion, preferences

Steps

1. **Insert Check Box** — Developer → Insert → Check Box.
2. **Rename it** — Change the displayed text.
3. **Copy if required** — Duplicate for multiple choices.
4. **Link to cell** — Use Format Control → Cell link.
5. **Use result** — Linked cells return TRUE or FALSE.

8.5 — Option Button

Theory

An Option Button is used when the user must choose only one option from a set of alternatives. Related Option Buttons are mutually exclusive.

Question	Options
Select Payment Mode	☒ Cash

	☐ UPI
	☐ Credit Card

Practical Use Cases

Payment method selection

Choose Monthly, Quarterly, or Yearly report

Choose Basic, Standard, or Premium plan

Gender selection in a form

Steps

1. **Insert Option Buttons** — Developer → Insert → Option Button.
2. **Create choices** — Add all required options.
3. **Use Group Box** — Place related options inside a Group Box.
4. **Link to a cell** — Use Format Control → Cell link.
5. **Read result** — Excel returns the selected option number.

8.6 — Group Box

Theory

A Group Box visually organizes related controls. It is especially useful when a worksheet contains multiple sets of Option Buttons.

Group 1: Payment Mode	Group 2: Membership
☐ Cash	☐ Basic
☐ UPI	☐ Premium
☐ Card	☐ Enterprise

Important Concept

- Option Buttons inside one Group Box work as one set.
- Separate Group Boxes allow separate selections.
- This prevents one selection from affecting unrelated Option Buttons.

8.7 — Combo Box

Theory

A Combo Box provides a dropdown list from which users can select one item. It is useful when there are many choices and displaying all options separately would take too much space.

Select Department
Sales
Human Resources
Finance
IT
Marketing

Steps to Create a Combo Box

1. **Prepare list** — Enter the list of options in worksheet cells.
2. **Insert Combo Box** — Developer → Insert → Combo Box.
3. **Draw control** — Place the Combo Box on the worksheet.
4. **Format Control** — Right-click → Format Control.
5. **Set Input Range** — Select the cells containing the list.
6. **Set Cell Link** — Choose a cell to store selected position.
7. **Return item** — Use INDEX if required.

Formula Example

- If the Combo Box returns the selected position in G2:
- =INDEX(A2:A6,G2)
- This returns the selected department name.

8.8 — Spin Button

Theory

A Spin Button allows users to increase or decrease a linked numeric value using small up and down arrows.

Property	Example
Current Value	5
Minimum Value	1
Maximum Value	100
Increment	1
Cell Link	B2

Example Use Cases

Select quantity of products
Change number of months
Adjust discount percentage
Create interactive calculations

Interactive Calculation

- Quantity in B2 = controlled by Spin Button
- Price in C2 = ₹500
- Total = B2*C2

8.9 — Scroll Bar

Theory

A Scroll Bar allows the user to move through a range of values. It is often used to create dynamic reports and interactive charts.

Property	Example
Minimum	1

Maximum	12
Current Value	3
Cell Link	B2
Use	Change month dynamically

Example

A Scroll Bar can control the month displayed in a report. When the user moves the bar, the linked cell changes and formulas or charts update automatically.

Dynamic Formula Example

- If B2 contains a month number selected by Scroll Bar:
- =INDEX(B5:B16,B2)
- The displayed result changes whenever the Scroll Bar value changes.

8.10 — Interactive Forms

Theory

Interactive Forms combine multiple Form Controls with formulas to create a user-friendly Excel application. User actions change linked cells, formulas calculate results, and the output updates automatically.

Field	Control Type	Example
Customer Name	Text Input	Rahul Sharma
Product	Combo Box	Laptop
Accessories	Check Boxes	Mouse + Keyboard
Payment Mode	Option Buttons	UPI
Quantity	Spin Button	3
Discount	Scroll Bar	10%

Workflow

1. **Create input section** — Design labels for user inputs.
2. **Insert controls** — Add the required Form Controls.
3. **Link cells** — Connect controls to worksheet cells.
4. **Create formulas** — Use linked values in calculations.
5. **Display results** — Show totals, selected options, or messages.
6. **Test form** — Change controls and verify outputs update.

Interactive Form Concept

- User Action → Form Control Changes → Linked Cell Updates → Formula Calculates → Result Changes Automatically

8.11 — Case Studies / Assignments

Case Study 1 — Course Registration Form

Create an interactive student course registration form using Excel Form Controls.

Requirement	Suggested Control
Student Name	Text input

Select Course	Combo Box
Preferred Subjects	Check Boxes
Learning Mode	Option Buttons
Number of Modules	Spin Button
Course Fee	Formula

Create the complete registration form.

Use at least one Combo Box and three Check Boxes.

Use Option Buttons for Online/Offline mode.

Use a Spin Button to select modules.

Calculate total course fee dynamically.

Case Study 2 — Interactive Product Order Form

Input	Requirement
Product	Use Combo Box
Accessories	Use Check Boxes
Payment Method	Use Option Buttons
Quantity	Use Spin Button
Discount	Use Scroll Bar
Final Amount	Calculate using formulas

Case Study 3 — Interactive Sales Report

Create an interactive sales report where the user can select Region and Month using Form Controls.

Task	Expected Result
Select Region	Display selected region data
Select Month	Change month using Scroll Bar
Use linked cells	Store selected values
Use formulas	Calculate selected sales
Create chart	Display dynamic sales chart

Practice Dataset

Order_ID	Product	Category	Region	Quantity	Price	Discount
O101	Laptop	Electronics	North	2	65000	5%
O102	Monitor	Electronics	West	3	18000	10%
O103	Keyboard	Accessories	South	5	2500	0%
O104	Mouse	Accessories	East	8	1200	5%
O105	Printer	Electronics	North	1	22000	8%
O106	Laptop	Electronics	West	4	70000	12%
O107	Monitor	Electronics	South	2	20000	5%
O108	Keyboard	Accessories	East	6	3000	10%

Final Assignment

Build a Complete Interactive Excel Form

- Create a professional order or registration form.
- Use at least five different Form Controls.
- Use linked cells correctly.
- Apply formulas to generate automatic results.
- Use formatting to make the form user-friendly.
- Test every control and verify that the output changes dynamically.

Macros In Excel With Examples

Excel macros are a versatile tool that can automate repetitive tasks, saving we time and increasing productivity. Whether we're trying to **enable Excel macros**, **record a macro in Excel**, or automate specific actions within our spreadsheet, macros are an invaluable feature.

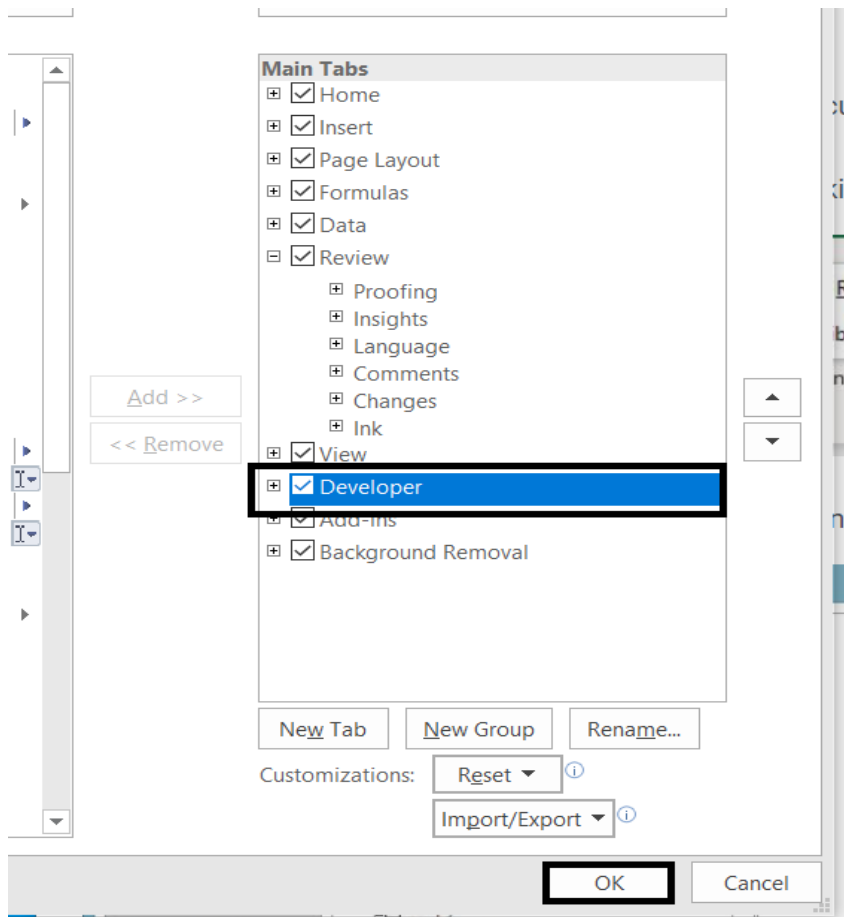
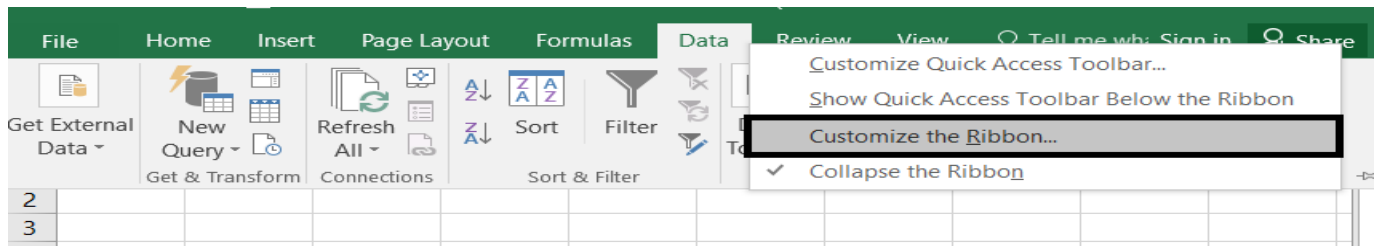
9.1. Prerequisites for Using Excel Macros

Before learning how to create macros in Excel we have to turn the developer mode in Excel. Follow these steps:

Step 1: Open MS Excel

Step 2: Right-click on the Ribbon

Step 3: Check the Developer box and Click on "OK"



9.2. Enabling Macros on Windows

1. Accessing the Trust Center

Steps:

1. **Open Workbook:** Launch Excel and open the macro-enabled workbook (.xlsm).
2. **Navigate to Options:** Go to File > Options > Trust Center > Trust Center Settings.
3. **Select Macro Settings:** In the Trust Center window, click Macro Settings.

2. Configuring Macro Settings

Steps:

1. Choose Option:

- **Disable all macros without notification:** Blocks all macros, no prompts.
- **Disable all macros with notification:** Blocks macros but prompts to enable (recommended).
- **Disable all macros except digitally signed:** Allows only trusted, signed macros.
- **Enable all macros:** Allows all macros (use cautiously due to security risks).

2. Save Settings:

Click OK twice to exit Trust Center and Options.

3. Enable Content:

If a yellow Security Warning bar appears, click Enable Content to run macros.

9.3. Enabling Macros on Mac

1. Accessing Preferences

Steps:

1. **Open Workbook:** Launch Excel and open the macro-enabled workbook.
2. **Navigate to Preferences:** Go to Excel > Preferences > Security.

2. Configuring Macro Settings

Steps:

1. **Enable Macros:** Check "Enable all macros" (use cautiously due to security risks).
2. **Save Changes:** Click OK.
3. **Reopen Workbook:** Close and reopen the workbook if macros don't activate.

Important Security Tip:

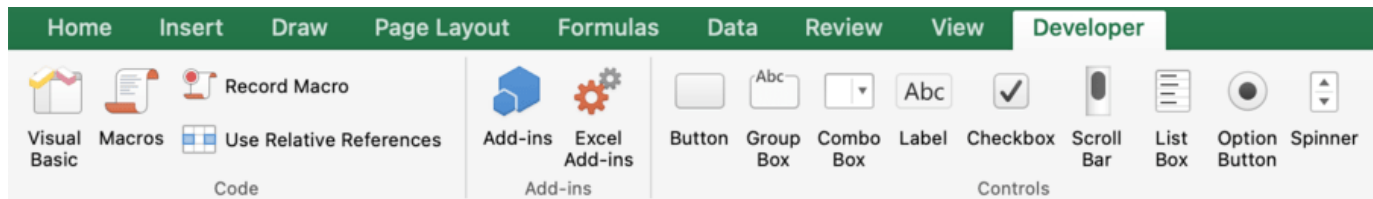
Macros can contain potentially harmful code. It is always best to enable macros from trusted sources only. Consider using "Disable all macros with notification" as our default setting, and then enable macros on a case-by-case basis.

9.4. How to Record a Macro in Excel

Recording a macro is one of the easiest ways to get started with Excel Macros:

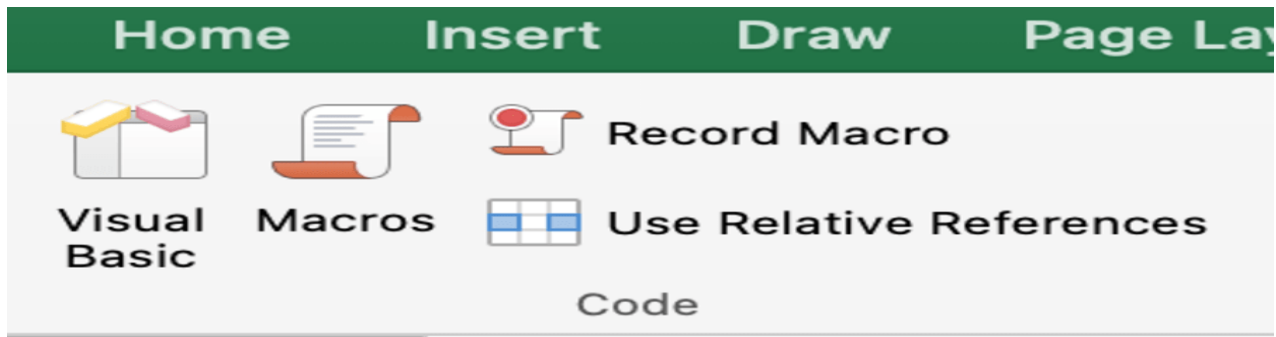
Step 1: Click on the Developer Tab

Go to the Developer tab located at the ribbon.



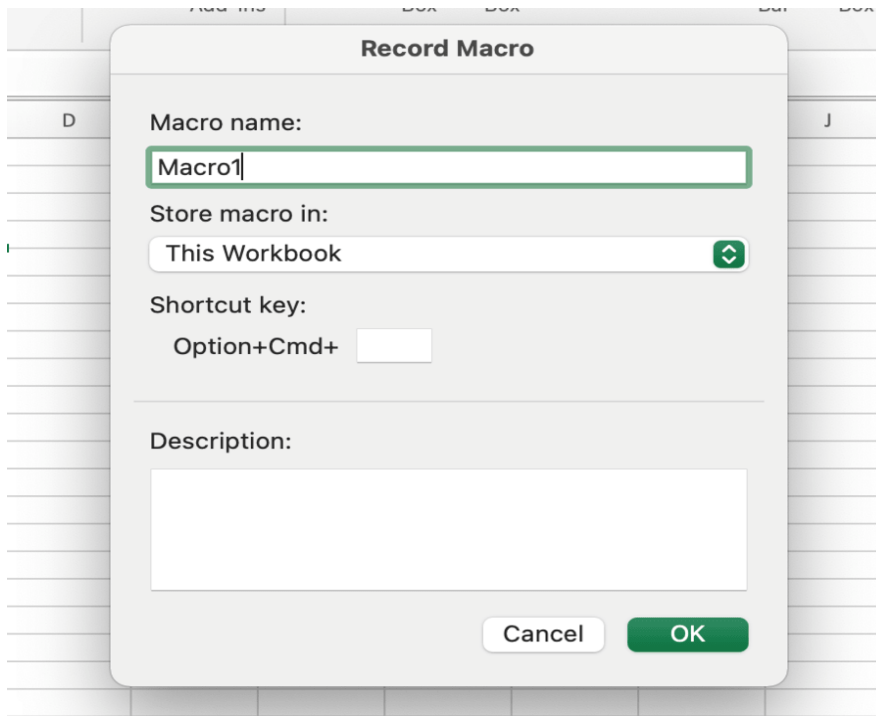
Step 2: Click on the Record Macro

In the Code Group, the tool **Record Macros** is located, to excel macros record we have to click on it.



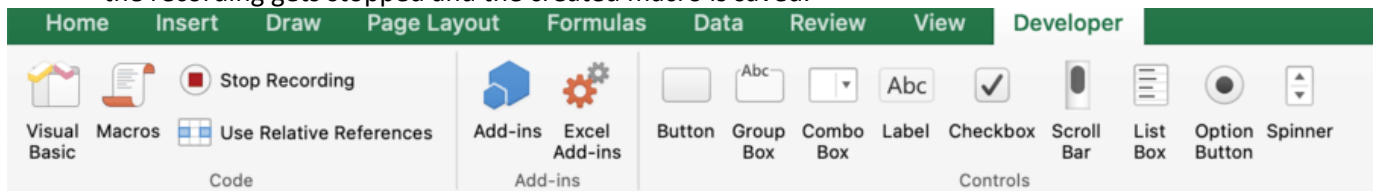
Step 3: Fill the Record Macro Dialog Box

As macros are selected, a dialog box is opened which contains the following options to fill.



In the dialog box, enter the name of the Excel macro in the Macro name. However, space cannot be used in the Macro name to separate the words, **underscore(_)** can be used for this purpose. Example: **Macro_name**. There are some other options also in the dialog box, i.e., where to store the macro, the shortcut key which can be used to activate the macro, and the description for the macro created. They can be filled accordingly.

- After filling in the dialog box, **click on OK**.
- As soon as OK is clicked, **Excel Macro** records the set of operations the user is practicing.
- Once, the work is done, in the **Developer Tab**, an option "**Stop Recording**" can be seen. Clicking on which the recording gets stopped and the created macro is saved.



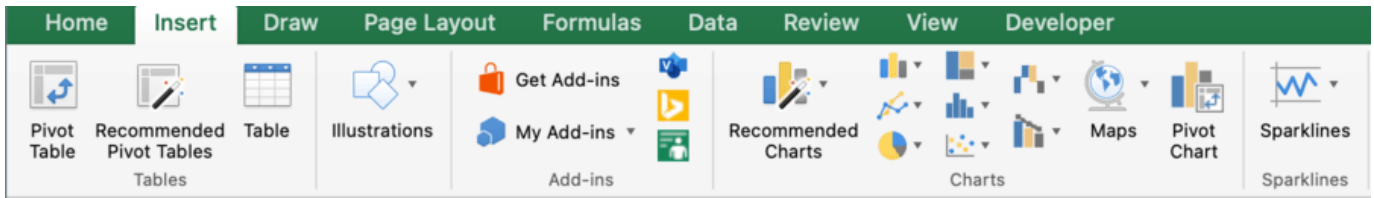
9.5. How to Run the Macro in Excel

There are multiple ways to run macros after **creating macros in Excel**, here we will learn how to run excel macros:

1. Run the Macro by Clicking on a Shape

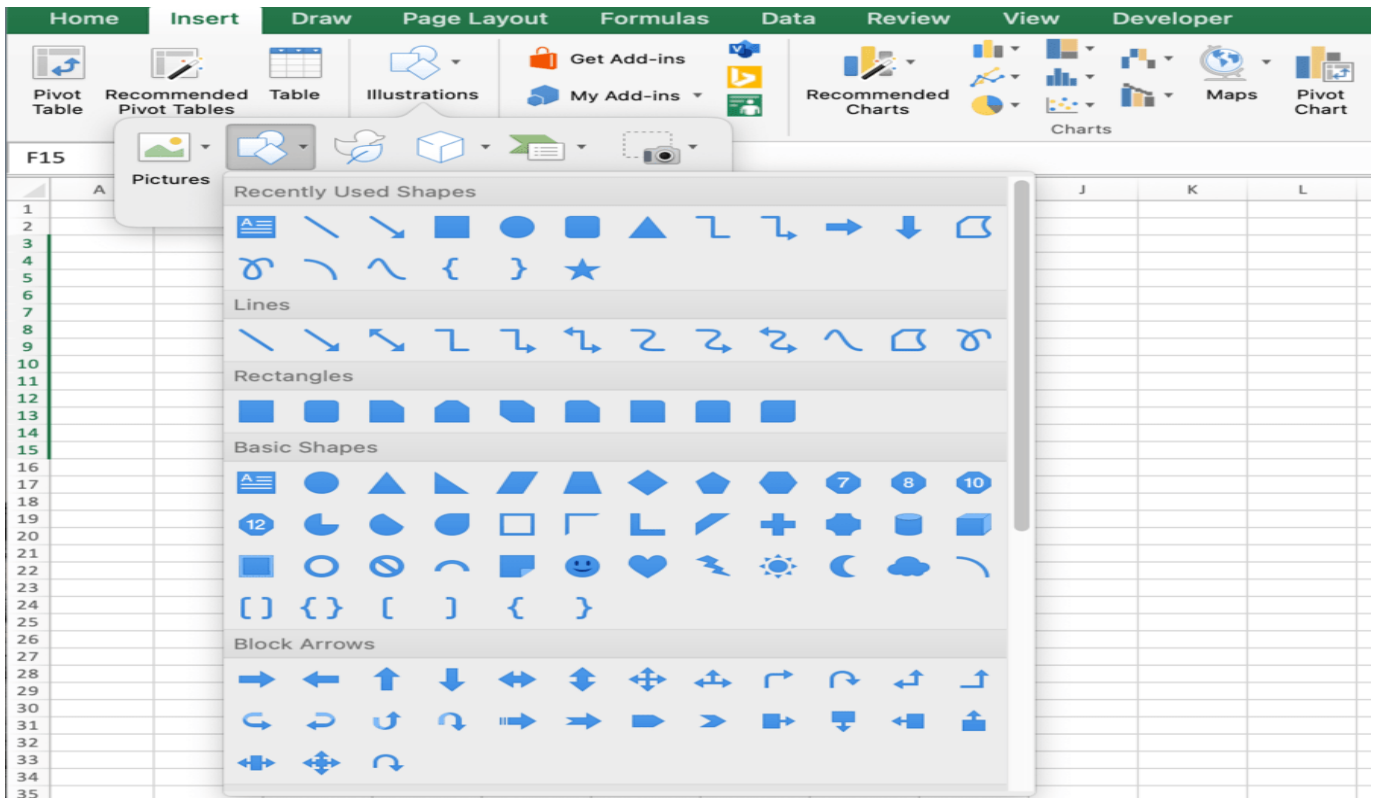
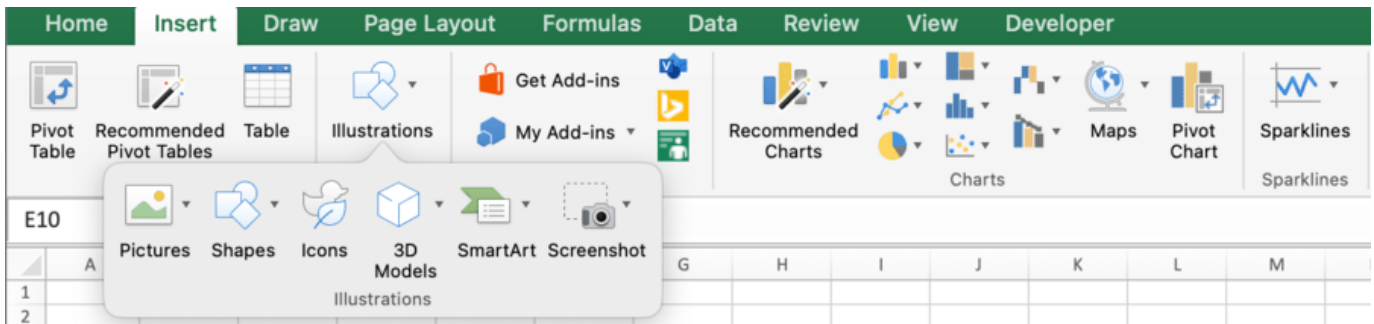
It is one of the easiest ways to run a macro to create any shape in the worksheet and use it for running the macro.

Step 1: Click on the Insert Tab on the ribbon



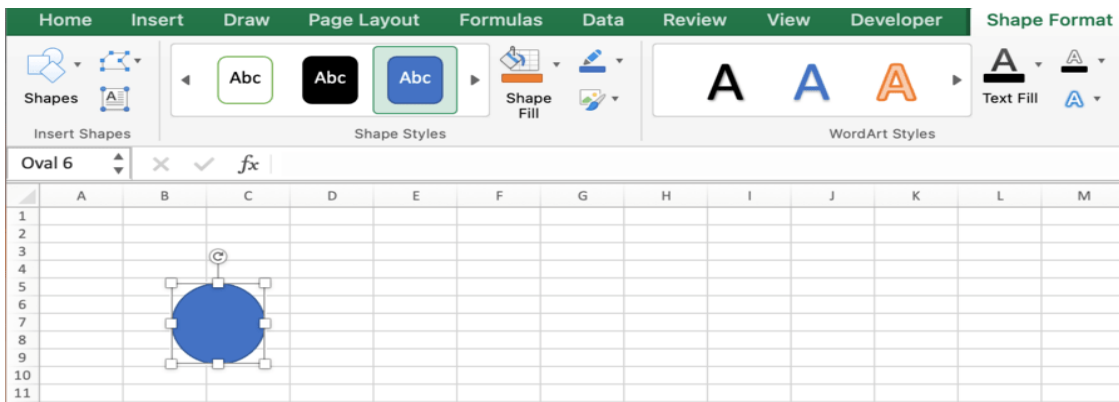
Step 2: Choose any Shape

Go to the Illustrations group and click on the Shapes icon. In the sheet, choose any shape we like and want it to be assigned as the macro.



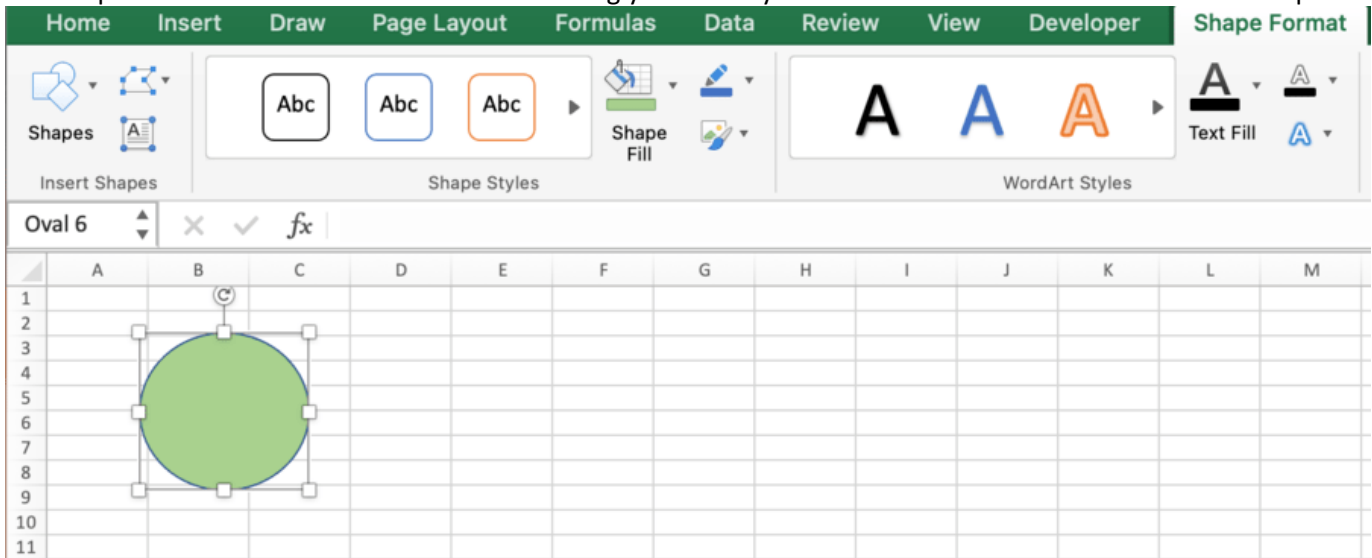
Step 3: Add the Shape

Click where we want to add the shape in the sheet and the shape will automatically get inserted.

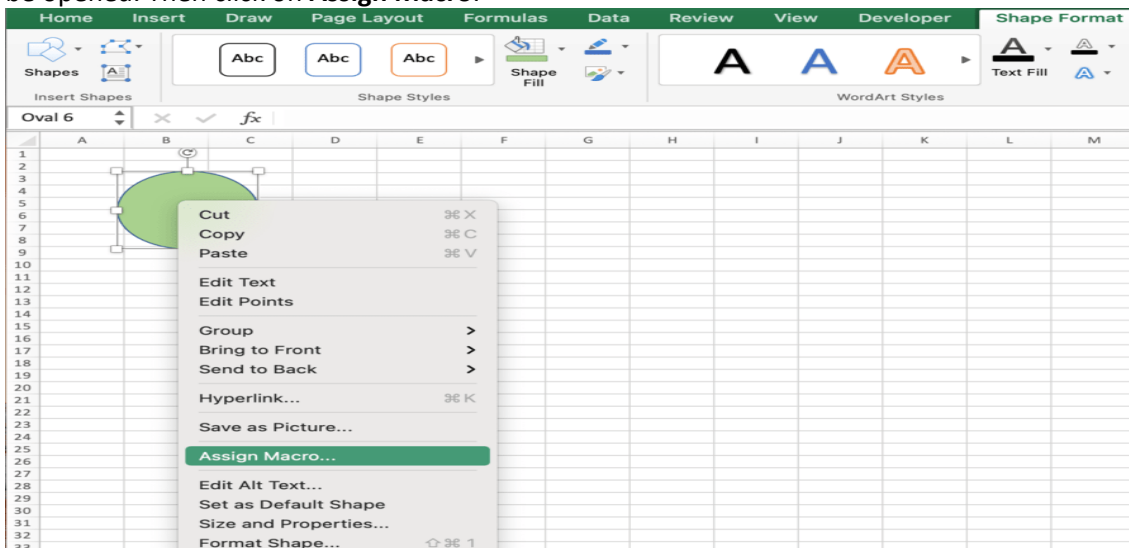


Step 4: Edit the Shape

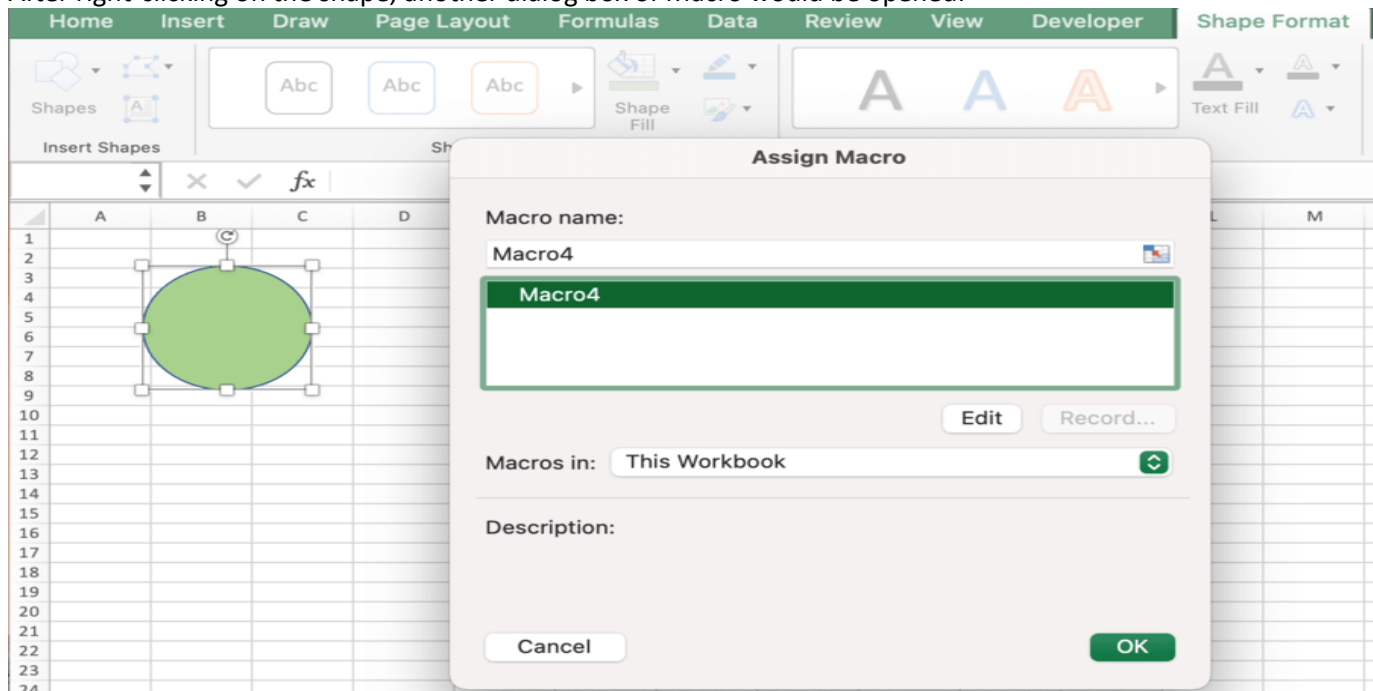
The shape can be resized or re-formatted accordingly to the way we want. Text can also be added to the shape.



Step 5: Right Click on the Shape and Click on Assign Macro Then Right-Click on the shape and a dialog box would be opened. Then click on **Assign Macro**.



After right-clicking on the shape, another dialog box of Macro would be opened.



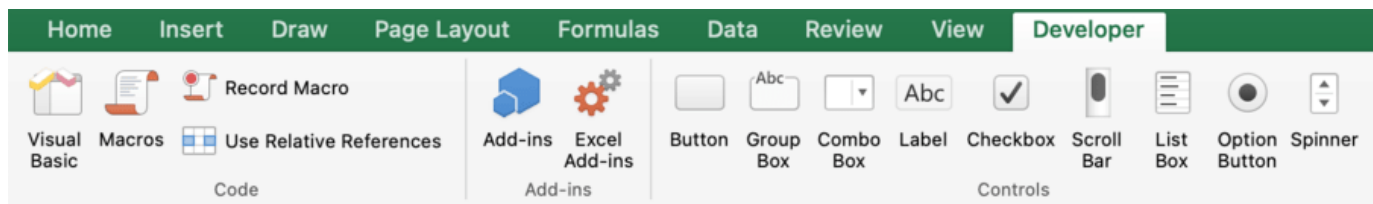
Step 6: Now Select the Macro

In the dialog box, select the macro from the list we want to assign to our shape and click the **OK button**. Now the shape would work as a button and whenever we click on it, it will run the assigned macro.

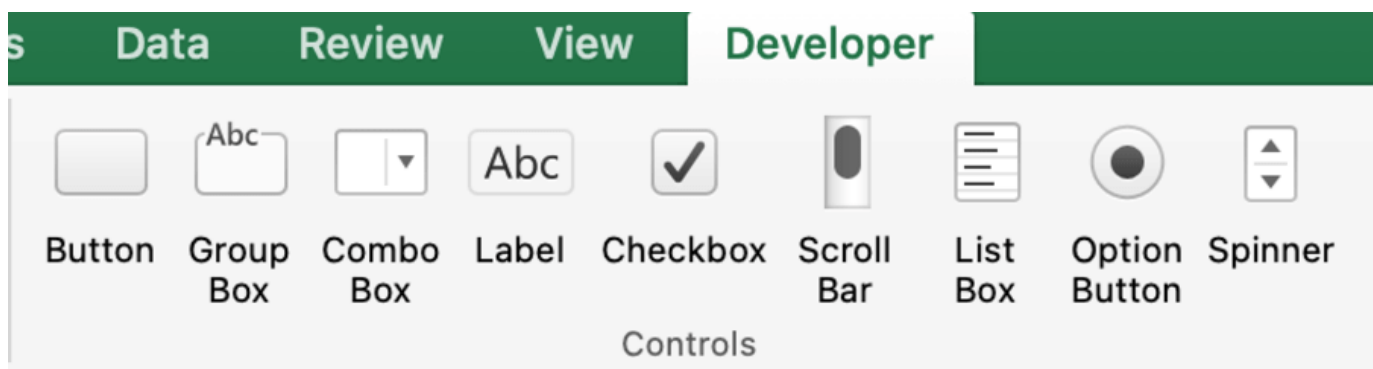
9.6. How to Run Macro By Clicking a Button

While the shape is something we can format and play with it, a button has a standard format. A macro can also be assigned to a button for easy use and then can run the macro in Excel by simply clicking that button.

Step 1: Go to the Developer Tab on the Ribbon

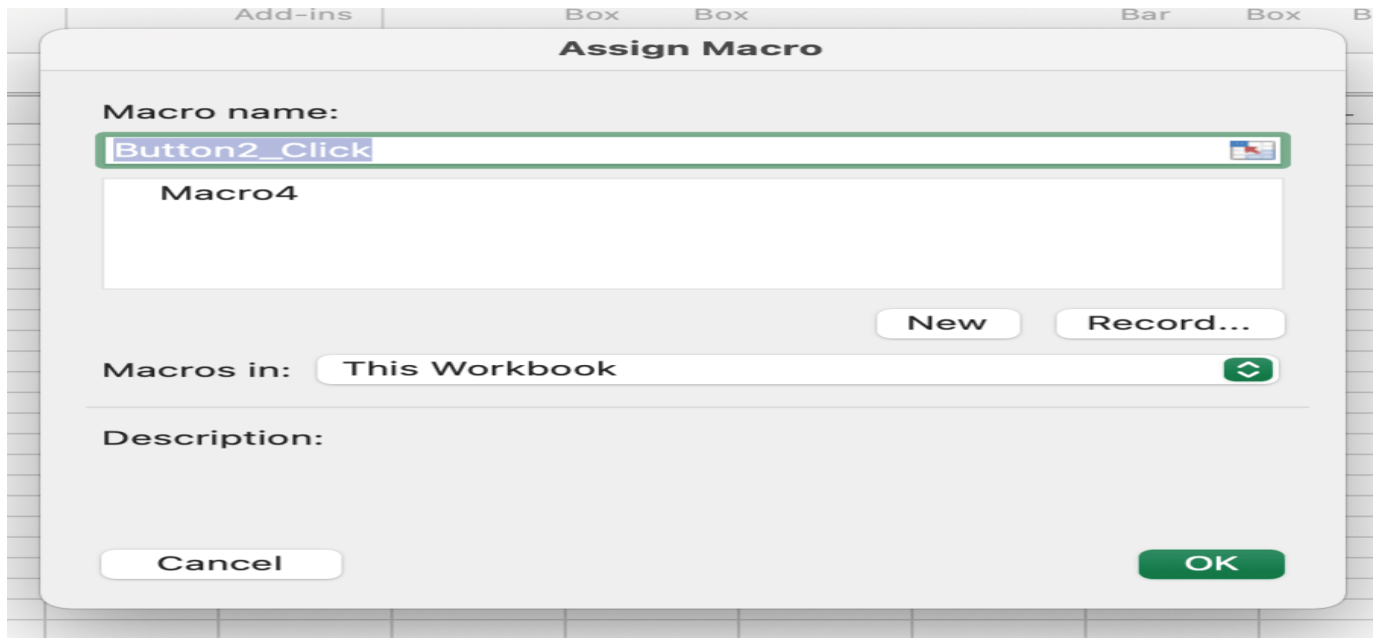


Step 2: In the Controls group, click on the Button Tool



Step 3: Click on the Excel Sheet and Fill the Details in the Dialog box

Now, click anywhere on the Excel sheet. As soon as we do this, the Assign Macro dialog box would be opened.

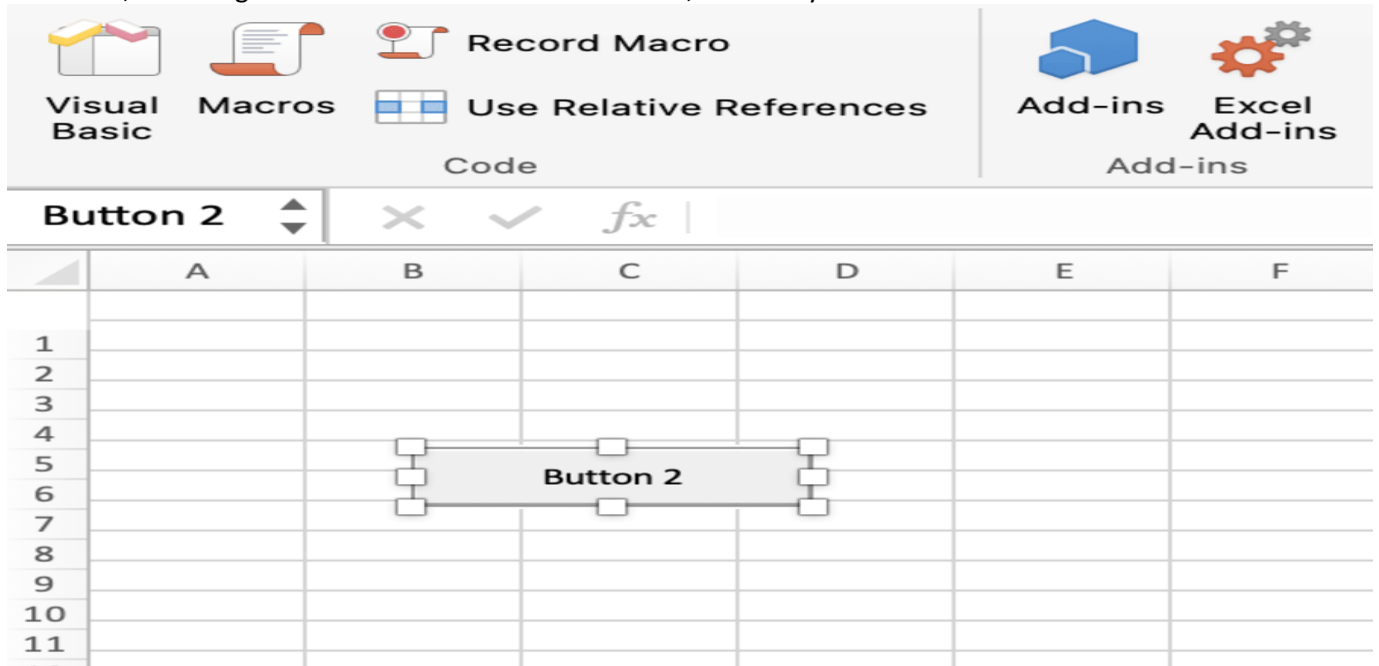


Step 4: Select the Macro we want to Assign and Click OK

Select the macro we want to assign to that button and fill in the other options. Then click on OK.

Step 5: Preview the Shape

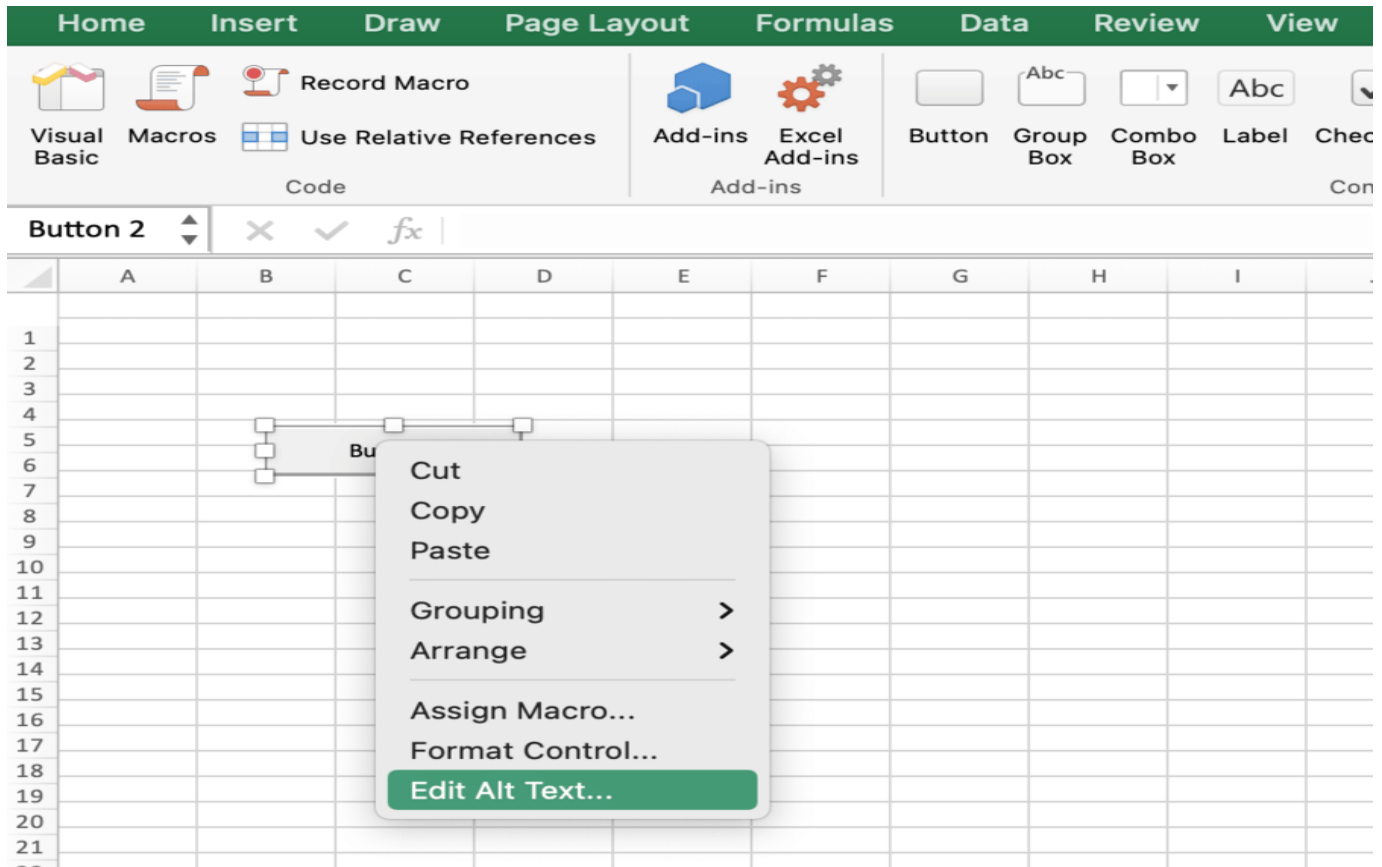
After that, a rectangular button can be seen on the sheet, and ready to be used.



Step 6: Format the Button

The button which appears on the sheet has default settings, and we can't format it like the color, or shape of the button.

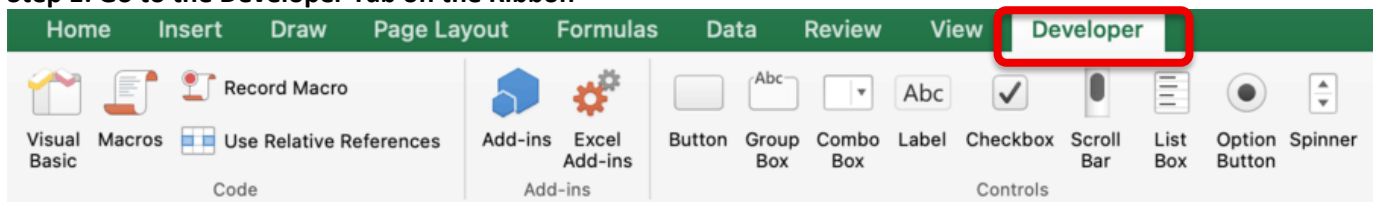
However, the text which appears on the button can be changed. Just a right-click on the button and a dialog box would open. Click on Edit Alt Text and change the text.



9.7. How to Create Macro in VBA Editor

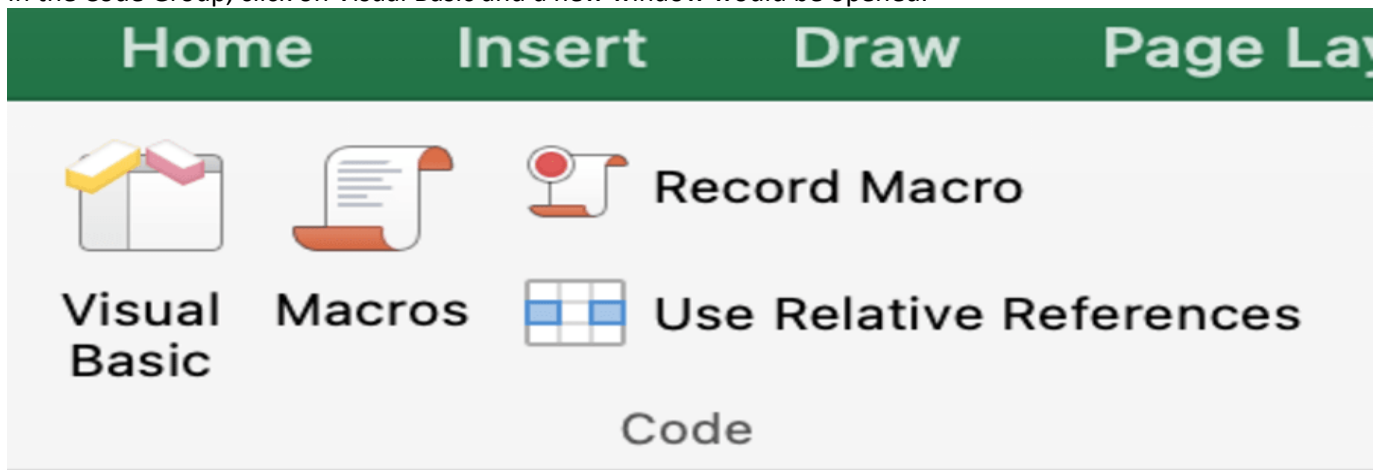
While recording macros is a quick way to automate tasks, writing **VBA code** allows us to create more complex macros with advanced functionality. To write custom VBA macros, follow these steps:

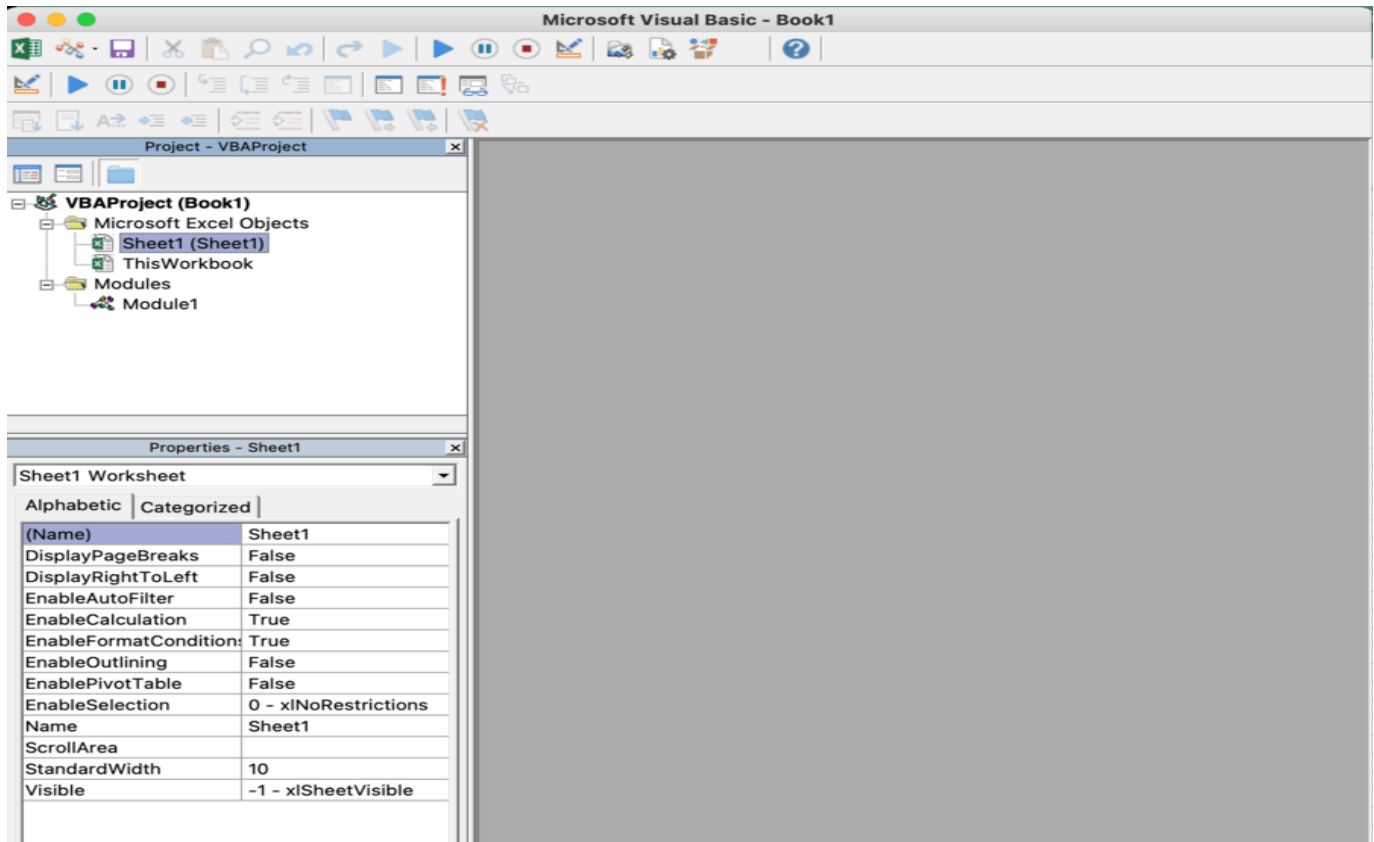
Step 1: Go to the Developer Tab on the Ribbon



Step 2: Click on the Visual Basic

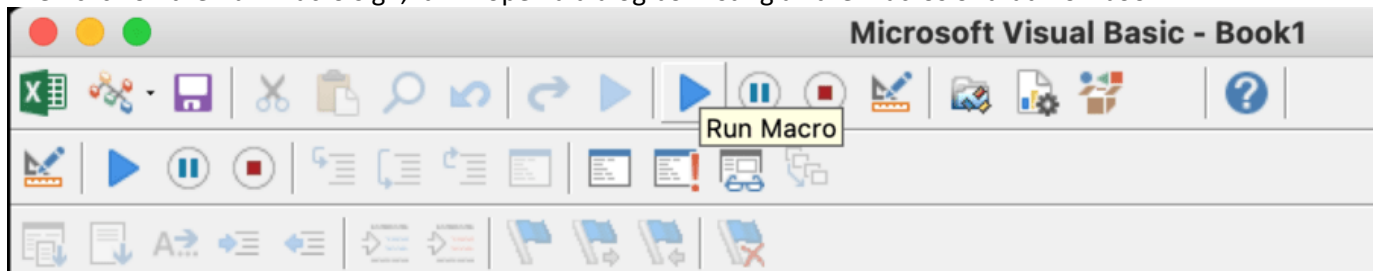
In the Code Group, click on Visual Basic and a new window would be opened.





Step 3: Click on Run Macro

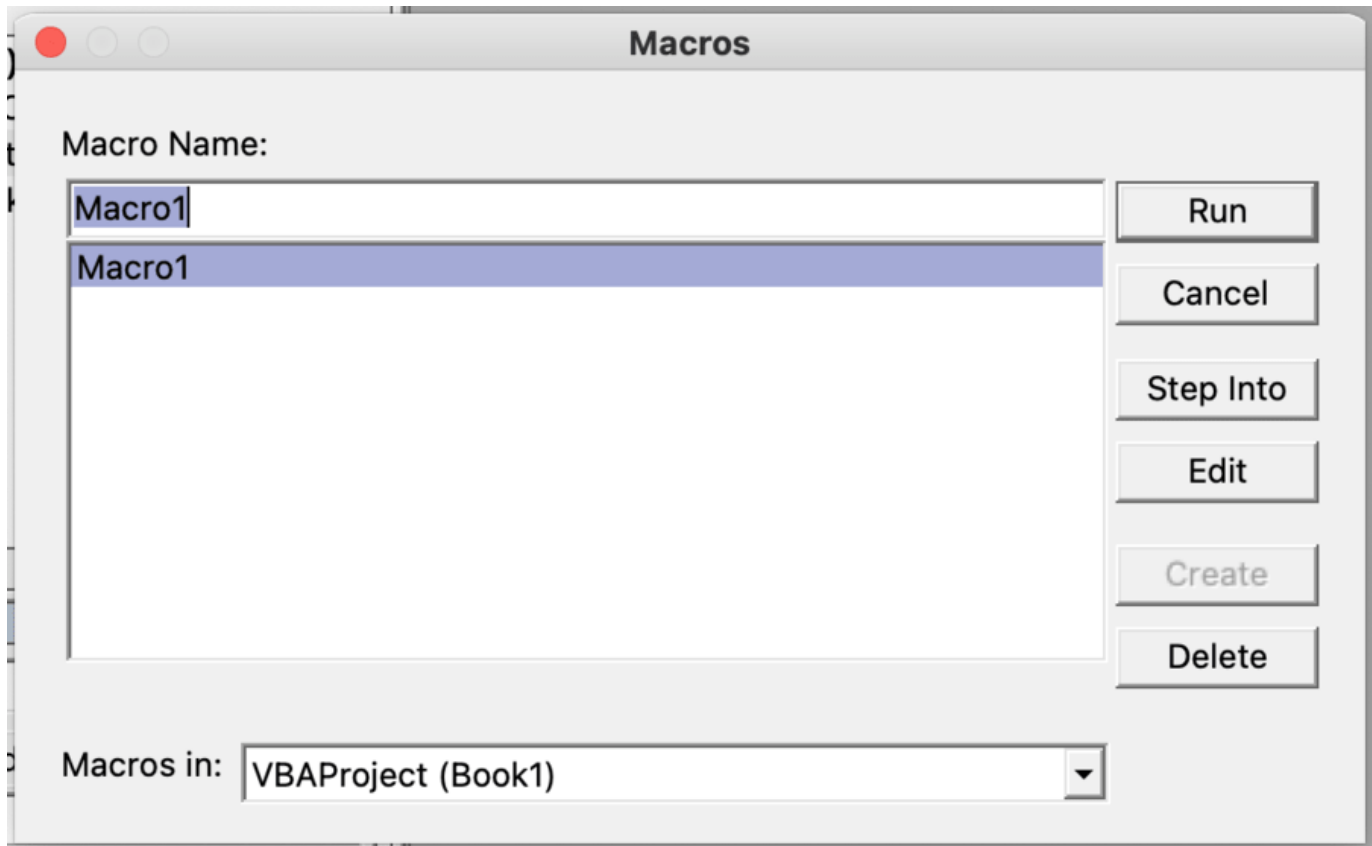
Then click on the Run Macro sign, it will open a dialog box listing all the macros of that workbook.



Step 4: Select the Macro from the list

Select the macro we want to run from the list and click Run.

As soon as Run is clicked, the macro would be executed on the excel-sheet. If we can only see the VB Editor window and not the sheet, then we may not see the changes happening in the worksheet. To see the changes, minimize/close the VB Editor window. we can also edit a macro as per our need.



8. How to View Macros in MS Excel

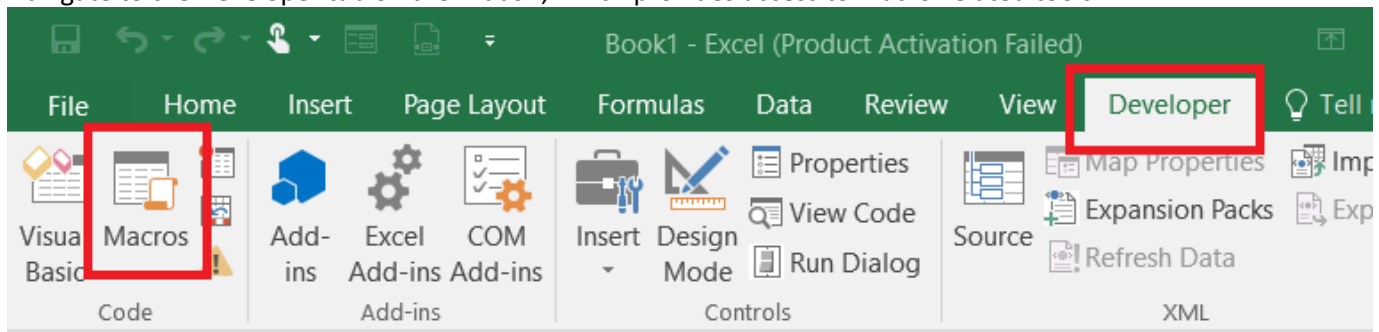
To view Macros in Excel follow the steps given below:

Step 1: Open Microsoft Excel

Launch Microsoft Excel by clicking the Excel icon on our computer (Windows or Mac). This opens the application, allowing us to access workbooks containing macros, such as a sales report with automation scripts.

Step 2: Click on the Developer Tab

Navigate to the Developer tab on the Ribbon, which provides access to macro-related tools.



Step 3: Click Macros to see a list of available macros in the active workbook

Click the Macros button in the Developer tab to open the Macro dialog box, displaying all macros in the active workbook.

Step 4: Select a Macro to View

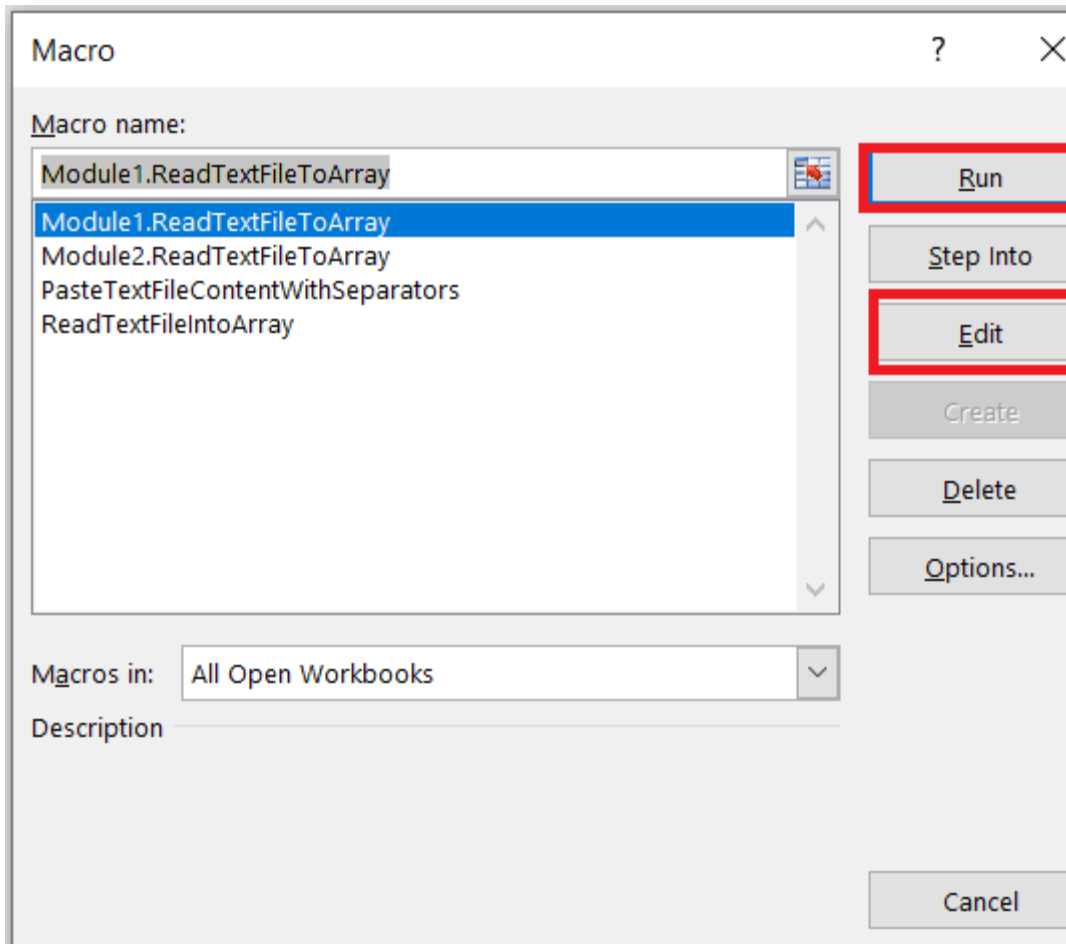
In the Macro dialog, click a macro name (e.g., "FormatSalesTable") to highlight it for review.

Step 5: Run or Edit the Macro

From the Macro dialog, we can run the selected macro to execute its actions or edit it in the VBA Editor to view or modify its code.

Step 6: Close the Macro dialog

Click Cancel in the Macro dialog to close it when finished reviewing or managing macros.

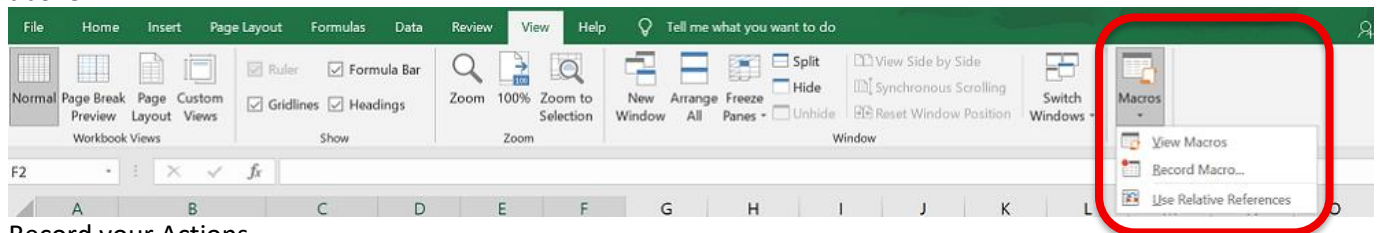


9. How to Save the Recorded Macros in Excel

After creating or recording a macro, save our workbook as an Excel Macro-Enabled Workbook (.xlsm) to keep the macro accessible in the future.

Step 1: Record our Actions

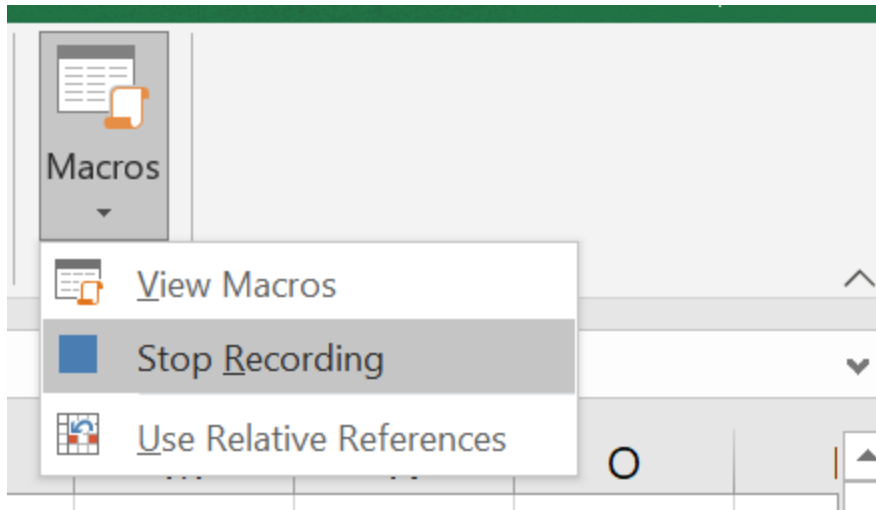
To save Excel macro we have to record it first so if we don't how to record Excel macro follow the steps given above.



Record your Actions

Step 2: Click on Stop Recording

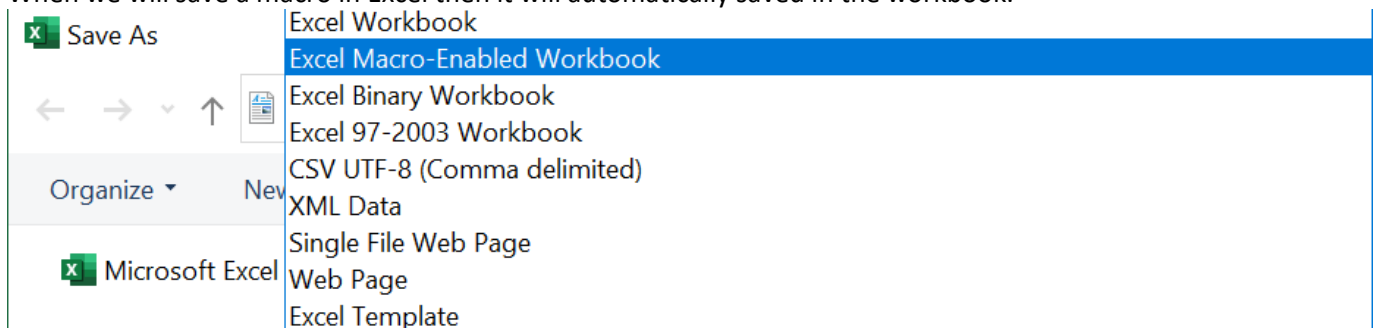
Stop the recording Excel Macro when we are done recording a macro in Excel.



Click on Stop Recording

Step 3: Save our workbook as “.xlsm” (macro-enabled)

When we will save a macro in Excel then it will automatically saved in the workbook.



Save your workbook as “.xlsm”

Step 4: Excel Macro Saved

Now our macro in excel is saved and we can use it whenever we want.

10. Example Macros for Practical Use

Macros can help automate a variety of tasks in Excel:

- **Data Formatting:** Create a macro to format a dataset automatically.
- **Report Generation:** Automate the generation of reports with specific parameters.
- **Sorting and Filtering:** Use macros to quickly sort or filter data.

CHAPTER 8 — WHAT-IF ANALYSIS

Goal Seek • Scenario Manager • Data Tables • Sensitivity Analysis

Chapter Overview

What-If Analysis lets you change assumptions and study how those changes affect a result. It is useful for pricing, budgeting, loan calculations, profit planning, and business decision-making.

10.1 — Goal Seek

Theory

Goal Seek works backwards from a desired result. You specify the formula cell that should reach a target value and the input cell Excel may change to reach that target.

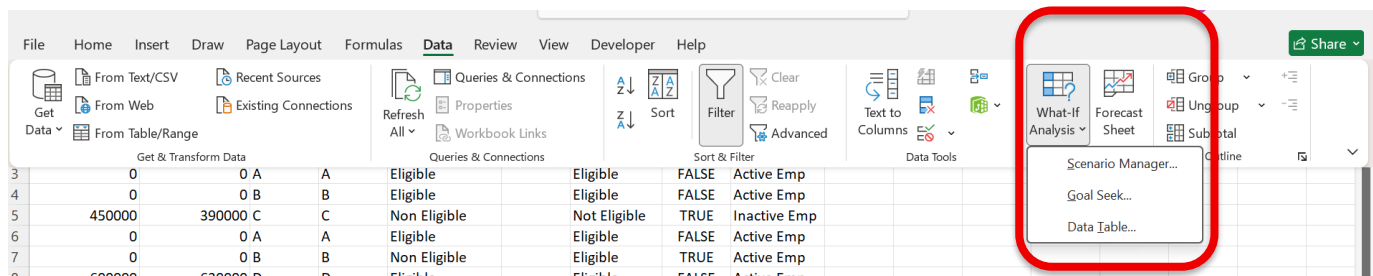


Figure: Goal Seek connects a desired result to the input Excel should change.

Example — Target Profit

Item	Value
Selling Price	₹500
Quantity	100
Cost per Unit	₹200
Current Profit	₹30,000
Target Profit	₹50,000

Suppose Profit = (Price – Cost) × Quantity. Goal Seek can calculate the selling price required to achieve ₹50,000 profit.

Steps to Perform Goal Seek

1. **Create the formula** — Calculate Profit in a result cell.
2. **Open Goal Seek** — Data → What-If Analysis → Goal Seek.
3. **Set cell** — Select the Profit formula cell.
4. **To value** — Enter 50000.
5. **By changing cell** — Select Selling Price.
6. **Run Goal Seek** — Click OK and review the solution.

Figure: Practical Goal Seek workflow.

Industry Use Cases

Finance: find the revenue needed to reach a profit target.

Pricing: find the selling price needed to achieve a required margin.

Loan Planning: find an input needed to meet an EMI target.
 Sales: find the units required to achieve a revenue objective.

10.2 — Scenario Manager

Theory

Scenario Manager stores different sets of input values and lets you switch between them. It is useful for comparing Base, Best, Worst, or other planning cases.

Figure: Scenario Manager compares different input assumptions and their resulting outputs.

Example — Business Profit Scenarios

Input	Base Case	Best Case	Worst Case
Selling Price	₹500	₹550	₹450
Units Sold	1,000	1,200	800
Cost per Unit	₹300	₹280	₹330
Expected Profit	₹2,00,000	₹3,24,000	₹96,000

Steps to Create a Scenario

1. **Prepare model** — Create formulas using clearly labeled assumption cells.
2. **Open Scenario Manager** — Data → What-If Analysis → Scenario Manager.
3. **Add scenario** — Click Add and name the scenario.
4. **Select changing cells** — Choose inputs that change between scenarios.
5. **Enter values** — Enter values for the selected scenario.
6. **Repeat** — Create Base, Best, Worst, and other cases.
7. **Show scenario** — Select a scenario and click Show.

Best Practice

Keep changing cells together and clearly labeled. This makes the model easier to understand and maintain.

10.3 — One-Variable Data Table

Theory

A one-variable Data Table shows how one formula result changes when one input value changes across several alternatives.

Figure: Data Tables show results for many possible input values in one view.

Example — EMI by Interest Rate

Interest Rate	Monthly EMI
6%	₹9,660
7%	₹9,901
8%	₹10,145
9%	₹10,400
10%	₹10,660

Steps

1. **Create the model** — Build a formula for the result, such as EMI.
2. **List alternatives** — Enter possible interest rates.
3. **Select the data-table block** — Include the formula cell and alternatives.
4. **Open Data Table** — Data → What-If Analysis → Data Table.
5. **Set the input cell** — Select the model input cell that changes.
6. **Click OK** — Excel calculates each alternative.

Industry Use Cases

Loan EMI under different interest rates
 Profit under different selling prices

Revenue under different unit volumes
Investment returns under different growth rates

10.4 — Two-Variable Data Table

Theory

A two-variable Data Table studies the effect of two changing inputs at the same time. One variable runs across the top and the other down the first column.

Rate / Loan	₹1,00,000	₹2,00,000	₹3,00,000
6%	₹9,660	₹19,320	₹28,980
8%	₹10,145	₹20,290	₹30,435
10%	₹10,660	₹21,320	₹31,980

Steps

1. **Build result formula** — Create a formula dependent on two inputs.
2. **Prepare row values** — List the first input across the top.
3. **Prepare column values** — List the second input down the left.
4. **Select the full table** — Include the formula and all alternatives.
5. **Open Data Table** — Data → What-If Analysis → Data Table.
6. **Set Row Input Cell** — Select the model cell for the top-row variable.
7. **Set Column Input Cell** — Select the model cell for the first-column variable.
8. **Click OK** — Excel fills the result matrix.

10.5 — Sensitivity Analysis

Theory

Sensitivity Analysis examines how sensitive a result is to changes in important assumptions. Data Tables and Scenario Manager can be used to build sensitivity analysis.

Figure: A sensitivity matrix compares the output under multiple combinations of assumptions.

Price \ Quantity	80 Units	100 Units	120 Units
₹450	₹16,000	₹20,000	₹24,000
₹500	₹20,000	₹25,000	₹30,000
₹550	₹24,000	₹30,000	₹36,000

How to Interpret Sensitivity

Large output changes from small input changes indicate high sensitivity.
Small output changes indicate lower sensitivity.
Focus attention on assumptions with the greatest business impact.

Industry Use Cases

Budgeting: test profit under different revenue and cost assumptions.
Pricing: identify prices that protect profit when demand changes.
Finance: evaluate interest-rate and investment-return assumptions.
Operations: test capacity, volume, and cost combinations.

10.6 — What-If Analysis Comparison

Tool	How It Works	Best Use
Goal Seek	Changes one input to achieve one target	Find required input for a desired output
Scenario Manager	Stores multiple sets of assumptions	Compare Base/Best/Worst cases
One-Variable Data Table	Changes one input across many	Study one-variable sensitivity

	values	
Two-Variable Data Table	Changes two inputs across a grid	Study combined sensitivity
Sensitivity Analysis	Evaluates impact of assumptions	Identify important business drivers

10.7 — Case Studies / Assignments

Case Study 1 — Profit Target

A company sells a product for ₹500. Variable cost is ₹300 and current volume is 1,000 units. Use Goal Seek to determine the selling price required to reach ₹3,00,000 profit.

Case Study 2 — Best / Base / Worst Scenario

Create three scenarios using Price, Units Sold, and Cost per Unit as changing cells. Compare the resulting profit.

Scenario	Price	Units	Cost
Base	₹500	1,000	₹300
Best	₹550	1,200	₹280
Worst	₹450	800	₹330

Case Study 3 — EMI Sensitivity

Build a one-variable Data Table for interest rates from 6% to 12%. Then create a two-variable Data Table for loan amounts ₹1 lakh to ₹5 lakh and interest rates 6%, 8%, 10%, and 12%.

Case Study 4 — Pricing Sensitivity

Create a sensitivity matrix showing profit for Price values ₹400, ₹450, ₹500, ₹550, ₹600 and Quantity values 500, 750, 1,000, 1,250, and 1,500.

Practice Dataset

Product	Price	Cost	Units	Target Profit
Laptop	65000	52000	20	300000
Monitor	18000	12000	40	250000
Printer	22000	15000	30	200000
Keyboard	2500	1400	100	100000
Mouse	1200	650	150	70000
Tablet	32000	24000	25	250000
Headset	3500	1900	80	120000
Webcam	4500	2600	60	100000

Final Assignment — Build a What-If Analysis Model

Create formulas for Revenue, Cost, and Profit.

Use Goal Seek to find the price required to reach Target Profit.

Create Base, Best, and Worst scenarios using Scenario Manager.

Create a one-variable Data Table for price changes.

Create a two-variable Data Table for price and units.

Prepare a sensitivity analysis and write three business conclusions.