



Excel for Office Work

A Workbook for Training Participants

Contents

1. How to Use This Workbook	4
Before You Start	4
How to Read the Formula Notation	4
Structure of One Exercise	5
2. Formatting and Tables	7
Why We Start With How the Sheet Looks	7
1. Cleaning Up a Worksheet and Formatting Numbers and Dates	7
2. Converting a Range Into an Excel Table	9
3. Table Styles and Filtering Headers	10
3. Formulas and References	12
Two Kinds of References, One Common Mistake	12
4. Your First Formula: Net Price to Gross Price	12
5. Relative References When Copying a Formula Down	13
6. Absolute References to a Fixed Rate	14
4. The IF Function	16
One Decision, Three Levels of Difficulty	16
7. A Simple Condition: Order Fulfillment Status	16
8. Nested IF: Three Bonus Tiers	17

9.	IF With AND and OR: Discount Qualification	18
5.	VLOOKUP and XLOOKUP	21
	Finding Data in Another Table	21
10.	Your First VLOOKUP: a Product's Price from the Price List	21
11.	The #N/A Trap and the Fourth Argument	22
12.	Switching to XLOOKUP	23
6.	Conditional Functions: SUMIFS and Its Family	26
	Summing, Counting, and Averaging by Condition	26
13.	SUMIFS With a Single Condition	26
14.	SUMIFS With Multiple Conditions	27
15.	COUNTIFS and AVERAGEIFS	28
7.	Sorting and Filtering	31
	Putting a Large List in Order	31
16.	Multi-Level Sorting of an Order List	31
17.	Custom Filter: a Range of Values	32
18.	An Autofilter Across Multiple Columns at Once	33
8.	PivotTables	35
	Summarizing Without Writing Formulas	35
19.	Your First PivotTable: Sales by Department	35
20.	Grouping Dates in a PivotTable	36
21.	A Calculated Column and Percentage Share	37
9.	Charts	39
	When Numbers Are Better Shown Than Listed	39
22.	A Column Chart Comparing Departments	39
23.	A Line Chart for a Trend Over Time	40
24.	A Chart From a PivotTable	42

10. Conditional Formatting	44
Color as a Signal, Not Decoration	44
25. Color Scales for Sales Values	44
26. A Rule With Its Own Formula: Overdue Deadlines	46
27. Data Bars and Managing Rules	47
11. Final Project: Monthly Department Report	49
One Data Set, Every Skill Learned	49
28. Preparing and Cleaning the Source Data	49
29. Summarizing With Formulas and a PivotTable	50
30. Visualization and Signaling: a Chart and Conditional Formatting	52
12. Summary and Next Steps	54
A Map of Thirty Exercises	54
What Comes Next When This Workbook Is Done	55
Before You Close This Workbook	55

01

How to Use This Workbook

Before You Start

This workbook accompanies an Excel training course for office staff, and it assumes you have the program open next to you, not just in front of your eyes. Each exercise states a goal, gives input data in a ready table, walks through it step by step, and tells you what the result should look like. Type the data from the table into your own blank worksheet, or download the exercise file from your trainer if the course provides one, and work on it alongside your reading.

The examples are based on the fictional company Biurplex, an office-supplies distributor with the departments Sales, Marketing, Logistics, and Finance, and five cities: Warsaw, Krakow, Poznan, Gdansk, and Wroclaw. Product names, figures, and results are illustrative and repeat across exercises, so you can build one worksheet over the course of several sessions instead of starting each task from zero.

How to Read the Formula Notation

Formulas are given in English notation, with a comma as the argument separator, matching what the English version of Excel displays. Cell references are written in capital letters (A1, B2:B11), and the sheet name is added to a formula only where it reaches into another tab.

💡 Save your file after each chapter

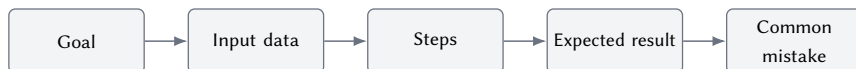
After each block of exercises, save the file under a new name (e.g. `excel_ex_01-03.xlsx`). If a formula stops working, it is easier to go back to an earlier version than to rebuild it from memory.

Excel Table

A data range converted into a named object with a fixed structure (the Ctrl+T shortcut), with automatic formula fill-down and header filtering. Distinguish it from a plain, unformatted range of cells, because many functions in this workbook rely on exactly that Table structure.

Structure of One Exercise

Every entry in this workbook follows the same layout, so you can practice without searching for what to read next.



The Expected result field is not an answer to copy, but a checkpoint: if your worksheet shows a different number or a different message, go back to the previous step instead of moving on. The Common mistake field describes the most frequent slip for that topic, noticed in earlier training sessions, and it is not a list of every possible error, only the one worth watching for first.

 Excel version

Function names and filter behavior can differ slightly between Excel in Microsoft 365 and older versions of the Office suite. XLOOKUP (Chapter 5) requires Excel 365 or Excel 2021; on an older version, replace it with VLOOKUP from the same chapter.

02

Formatting and Tables

Why We Start With How the Sheet Looks

Before a worksheet can calculate anything for you, it has to correctly recognize what was typed into it. A number entered as text will not enter a sum, and a date in the wrong format will not sort chronologically. These three exercises clean up the input data and turn a plain range into an Excel Table, which every later chapter in this workbook relies on.

EXERCISE 1

Cleaning Up a Worksheet and Formatting Numbers and Dates

Goal: recognize and fix data entered in the wrong format before it reaches a formula.

Table 2.1: A fragment of the Biurplex order list (input data)

Product	Date	Qty	Price
A4 Paper 80g	2024-03-02	40	18.90
HP 05A Toner	03.02.2024	3	249
Office chair	March 3	2	389.00

Steps:

1. Select the Date column and check whether every cell is right-aligned (that is how Excel aligns numbers and dates it recognizes). A left-aligned cell, like „March 3“, is text.
2. Fix the text dates to the month/day/year format and check the alignment again.
3. Select the Price column and in the Home tab set the number format to Currency with two decimal places, so that 249 and 18.90 look the same way.
4. Select the whole table and in the Format Cells window (Ctrl+1) check the format of each column under the Number tab.

Expected result: every date is right-aligned, every price has two decimal places and a currency symbol, and no number is stored as text (no green triangle in the top-left corner of the cell).

A date entered as text

„March 3“ or „03.02.2024“ typed by hand is often accepted by Excel as text, not as a date. A cell like that cannot be subtracted from another date and will not sort together with the rest of the column. Check alignment, not just the look of the number.

EXERCISE 2

Converting a Range Into an Excel Table

Goal: turn a plain range of cells into an Excel Table and see what that changes about working with formulas.

Steps:

1. Click any cell in the order list and press Ctrl+T.
2. In the Create Table window, confirm the selected range and leave „My table has headers” checked.
3. Go to the Table Design tab and rename the table from the default Table1 to tblOrders.
4. Add a new row below the last existing data row and notice that the table’s formatting extends to it automatically.

Expected result: the data appears as a Table named tblOrders, with filter buttons on the headers, and the new row inherits the formatting without you copying it by hand.

Structured references

A formula written inside a Table column refers to it by column name, for example `=tblOrders[Quantity]*tblOrders[Price]`, and extends itself to new rows automatically. That is more convenient than keeping track of a range like A2:A50 and fixing it after every new order.

EXERCISE 3

Table Styles and Filtering Headers

Goal: give the table a readable style and filter it by a value in the header, without writing a single formula.

Table 2.2: tblOrders after adding a Department column

Product	Department	Qty
A4 Paper 80g	Logistics	40
HP 05A Toner	Logistics	3
Office chair	Sales	2
Product catalog	Marketing	150

Steps:

1. From the Table Design tab, pick one of the built-in table styles and apply it.
2. Click the filter arrow in the Department column header.
3. Uncheck „Select All” and check only Logistics, to show only that department’s rows.
4. Clear the filter (the arrow again, „Clear Filter from Department”) and instead sort the table alphabetically by the Department column.

Expected result: after filtering, only two Logistics rows are visible, the row numbers on the left are blue and non-consecutive, and after clearing the filter all rows return, grouped by department.

✓ Formatting and Tables

- ☐ I checked the alignment of cells with dates and prices (right = number/date recognized).
- ☐ I converted a range into an Excel Table and gave it its own name.
- ☐ I wrote at least one formula with a structured reference to a Table column.
- ☐ I filtered and cleared a filter on one header column.
- ☐ I sorted the table alphabetically by a text column.

Formulas and References

Two Kinds of References, One Common Mistake

A formula in Excel almost always refers to other cells rather than to numbers typed in permanently. The trouble starts when you copy that formula down a column: a relative reference shifts along with it, while an absolute reference is meant to stay put. Confusing these two behaviors is one of the most common reasons a copied formula calculates correctly in the first row and incorrectly in every row after it.

EXERCISE 4

Your First Formula: Net Price to Gross Price

Goal: build a formula that multiplies and adds, referring to two cells.

Table 3.1: Biurplex price list (input data)

Product	Net price	VAT rate
A4 Paper 80g	15.00	23%
HP 05A Toner	202.44	23%
Office chair	316.26	23%

Steps:

1. In column D, type the header Gross price.
2. In cell D2, type the formula $=B2*(1+C2)$.
3. Confirm with Enter and check the result for A4 paper (15.00 times 1.23).
4. Copy the formula to D3 and D4 by dragging the small black square in the bottom-right corner of D2.

Expected result: D2 shows 18.45, D3 shows 249.00, D4 shows 389.00 (rounded for display to two decimal places).

A finished result typed in by hand

Typing 18.45 into D2 instead of the formula gives the same visible result, but it stops updating when the net price in B2 changes. A formula has to refer to the cells, not to a value calculated beforehand.

EXERCISE 5

Relative References When Copying a Formula Down

Goal: observe how a relative reference moves together with a formula copied to the next rows.

Steps:

1. Select cell D2 from the previous exercise and look at the formula in the formula bar.
2. Copy D2 (Ctrl+C), select D5, and paste (Ctrl+V).

- 3. Read the formula in D5 in the formula bar: it should refer to B5 and C5, not to B2 and C2.
- 4. Paste the formula once more into cell F10 and compare which rows it refers to now.

Expected result: every pasted copy refers to the cells in its own row, shifted by however many rows the formula itself was moved.

 A quick preview without pasting

Instead of actually pasting a formula somewhere else, select the cell with the formula and look at the blue and green outlines around the cells it refers to. That is the same information without changing the worksheet’s contents.

EXERCISE 6

Absolute References to a Fixed Rate

Goal: lock a reference to one shared cell with a dollar sign, so it does not shift when the formula is copied.

Table 3.2: A single markup for the whole price list

Parameter	Value
Trade markup (cell G1)	20%

Steps:

1. In cell G1, type 20% as a single, shared trade markup for the whole price list.
2. In E2, type the formula `=D2*(1+$G$1)`, referring to the gross price in column D and the markup in G1.
3. Copy E2 to E3 and E4.
4. Change the value in G1 to 25% and check whether all three results in column E recalculated together.

Expected result: after changing the markup in G1, every row in column E recalculates at once, without editing the formula in each of them.

The dollar sign freezes a reference

`$G$1` does not change when the formula is copied in any direction, because the dollar sign before the letter locks the column and the dollar sign before the number locks the row. `B2` without a dollar sign shifts along with the formula, while `$B$2` stays in place no matter where the formula is pasted.

Formulas and References

- ☐ I wrote a formula that refers to cells, not to a hard-coded result.
- ☐ I copied a formula with a relative reference and checked that it shifted along with it.
- ☐ I locked a reference with a dollar sign (`$B$2`) to one shared cell.
- ☐ I changed the value in the locked cell and watched every row recalculate.

04

The IF Function

One Decision, Three Levels of Difficulty

IF is the first function in this workbook that makes a decision instead of just calculating. It checks a condition and, depending on the result, returns one of two values. The three exercises in this chapter show it with a single condition, nested inside itself for several conditions in a row, and combined with the AND and OR functions, when a decision depends on more than one factor at once.

EXERCISE 7

A Simple Condition: Order Fulfillment Status

Goal: build your first IF formula with a single logical condition.

Table 4.1: Biurplex orders to evaluate for status

Order	Shipped	Qty
ZB-101	YES	40
ZB-102	NO	3
ZB-103	YES	2

Steps:

1. In column D, type the header Status.
2. In D2, type the formula `=IF(B2="YES", "Fulfilled", "Pending")`.
3. Copy the formula to D3 and D4.
4. Change the value in B3 from NO to YES and check whether the status in D3 updated automatically.

Expected result: D2 and D4 show „Fulfilled”, D3 shows „Pending” before the change and „Fulfilled” after it.

Quotation marks around text

The logical argument `B2="YES"` compares the cell's contents with text in quotation marks. Leaving out the quotes (`B2=YES`) makes Excel read it as a reference to a cell named YES, which does not exist, and return a #NAME? error.

EXERCISE 8

Nested IF: Three Bonus Tiers

Goal: build a formula with two nested conditions that split the result into three, not two, possible outcomes.

Steps:

1. In column B (after moving in the data), type the header Bonus.
2. In B2, type `=IF(A2>=20000, "High bonus", IF(A2>=10000, "Standard bonus", "No bonus"))`, where A2 is that employee's sales.

Table 4.2: An employee's monthly sales result

Employee	Sales
Kowalska	18,500
Nowak	9,200
Zalewski	24,000

3. Copy the formula to the remaining rows.
4. Check the result for each of the three employees in the table above.

Expected result: Kowalska (18,500) gets „Standard bonus”, Nowak (9,200) gets „No bonus”, Zalewski (24,000) gets „High bonus”.

 Check conditions from the highest threshold down

The second condition in a nested IF only runs when the first one was false, so the order of thresholds matters. Starting from the highest threshold (20,000) and working down avoids a high result being wrongly caught by a lower threshold.

EXERCISE 9

IF With AND and OR: Discount Qualification

Goal: combine IF with the AND and OR functions, so the condition depends on two columns at once.

Table 4.3: Customer data for discount evaluation

Customer	Years as client	Annual revenue
Lipa Foundation	1	42,000
Hurt-Serwis	6	8,000
AKS Design Studio	4	61,000

Steps:

1. In column D, type the header Discount.
2. In D2, type =IF(AND(B2>=3,C2>=50000),"10% discount","5% discount"), where B2 is years as a client and C2 is annual revenue.
3. Copy the formula to D3 and D4.
4. Swap AND for OR in the same formula, copy it to a new column E, and compare the results for Hurt-Serwis.

Expected result: with AND, only AKS Design Studio (4 years and 61,000) gets „10% discount”. With OR, Hurt-Serwis (6 years, despite low revenue) also gets „10% discount”, because only one of the two conditions needs to hold.

Biurplex's accounting team, quarter close

Swapping AND for OR in a discount rule without checking with the sales team can extend the discount to customers who were not meant to receive it, because they meet only one of two conditions. Before rolling out a formula

like that on a real worksheet, it is worth running it over the full customer list and comparing the number of discounts „before” and „after”.

☒ The IF Function

- ☐ I built an IF formula with a single text condition in quotation marks.
- ☐ I nested a second IF, starting the check from the highest threshold.
- ☐ I combined IF with AND to require two conditions at once.
- ☐ I swapped AND for OR and understood why the result changed.

VLOOKUP and XLOOKUP

Finding Data in Another Table

Up to this point, every formula calculated using cells from the same row. In real office work, data more often sits in two separate tables: for instance, the order list has only a product code, while the price for that code sits in a separate price list. VLOOKUP and the newer XLOOKUP function join tables like that through a shared column.

EXERCISE 10

Your First VLOOKUP: a Product's Price from the Price List

Goal: pull a price from a separate price-list table based on the product code in the order list.

Table 5.1: Biurplex price list (helper table, columns A:C)

Code	Name	Price
P-100	A4 Paper 80g	18.90
T-205	HP 05A Toner	249.00
K-310	Office chair	389.00

Steps:

1. In the order list, in the product-code column (e.g. F2), type one of the codes from the price list, such as T-205.
2. In cell G2, type `=VLOOKUP(F2,$A$2:$C$4,3,FALSE)`.
3. Check whether G2 returned 249.00, the toner's price.
4. Change the third argument from 3 to 2 and check that the formula now returns the product name instead of the price.

Expected result: with the third argument set to 3, the formula returns the price (249.00); with 2, it returns the name (HP 05A Toner), because that argument counts columns from the left edge of the price-list table, counting the code column as the first one.

 Lock the price-list table with dollar signs

`$A$2:$C$4` has a locked reference, so the formula can be copied to the next rows of the order list without the price-list range shifting. Without the dollar signs, the second copy would search in an already-shifted, partly empty range.

EXERCISE 11**The #N/A Trap and the Fourth Argument**

Goal: understand what the fourth argument, `FALSE`, does, and see what happens when the searched value does not exist in the price list.

Steps:

1. In cell F3, type a code that is not in the price list, such as X-999.
2. Copy the formula from G2 to G3 and check the result.
3. Change the formula's fourth argument from FALSE to TRUE (or remove it) and check whether the result changed.
4. Restore the fourth argument to FALSE, because on an unsorted list of codes TRUE returns a random, incorrect match instead of flagging that the code is missing.

Expected result: with FALSE, the formula for X-999 returns a #N/A error, clearly signaling no match. With TRUE, the formula reports no error at all, instead returning the result for the nearest smaller code in the unsorted list, which in this price list is a false result rather than a missing one.

TRUE as the default behavior

Leaving out the fourth argument behaves like TRUE, meaning an approximate match. That is the most common reason a VLOOKUP formula returns numbers that look believable but belong to a different product. Type FALSE deliberately; do not leave the argument blank.

EXERCISE 12

Switching to XLOOKUP

Goal: build the same lookup with the XLOOKUP function and see how it differs from VLOOKUP.

Steps:

1. In a new column, H2, type =XLOOKUP (F2, \$A\$2:\$A\$4, \$C\$2:\$C\$4, "code not found").
2. Compare the result with G2: for T-205, both formulas should show 249.00.
3. For code X-999, compare the result in H3 with G3: H3 shows the text „code not found” instead of #N/A.
4. Change the third array to \$B\$2:\$B\$4, so XLOOKUP returns the product name instead of the price, without changing the column where it searches for the code.

Expected result: XLOOKUP returns the same values as VLOOKUP for existing codes, its own text instead of #N/A for code X-999, and changing the result column does not require recalculating a column number, only pointing to a different array.

Why XLOOKUP Does Not Count Columns

VLOOKUP requires the searched column to be the first one in the range, and the result is the N-th column counted from it. XLOOKUP splits that into two independent arrays, a search array and a return array, which can sit anywhere in the worksheet, even to the left of each other.

☒ **VLOOKUP and XLOOKUP**

- ☐ I built a VLOOKUP with the price-list range locked by dollar signs.

- ☐ I set the fourth argument to FALSE and checked the #N/A error for a missing code.
- ☐ I built an XLOOKUP returning the same result, with its own message instead of #N/A.
- ☐ I changed the result column in XLOOKUP without recalculating column numbers.

Conditional Functions: SUMIFS and Its Family

Summing, Counting, and Averaging by Condition

IF decides about a single row. SUMIFS acts on a whole column at once and sums only the rows that meet one or several conditions at the same time. COUNTIFS and AVERAGEIFS have the exact same argument structure, except instead of summing they count rows or average them.

EXERCISE 13

SUMIFS With a Single Condition

Goal: sum values from one column, limited to rows that meet one condition in another column.

Table 6.1: Biurplex sales in March (fragment)

Department	City	Value
Sales	Warsaw	12,400
Sales	Krakow	8,100
Marketing	Warsaw	3,200
Sales	Poznan	6,700

Steps:

1. In an empty cell, say F1, type the header Sales department total.
2. In F2, type `=SUMIFS(C2:C5,A2:A5,"Sales")`, where C is the Value column and A is the Department column.
3. Check whether the result is the sum of the three Sales department rows, excluding the Marketing row.
4. Change the text in the condition to „Marketing” and check that the result drops to one row.

Expected result: for the „Sales” condition the formula returns 27,200 (12,400 + 8,100 + 6,700); for „Marketing” it returns 3,200.

💡 The first argument is always the range to sum

In SUMIFS, the first argument is the range of numbers to sum, and only after it come pairs of range and condition. That is a different order than SUMIF with a single condition, where the range to sum is often the last argument, so moving between the two functions makes it easy to mix up the argument order.

EXERCISE 14

SUMIFS With Multiple Conditions

Goal: add a second condition to the same formula and see that both must hold at once.

Steps:

1. In F3, type `=SUMIFS(C2:C5,A2:A5,"Sales",B2:B5,"Warsaw")`, where B is the City column.
2. Check the result: it should cover only the Sales-in-Warsaw row.
3. Add a new row to the table: Sales, Warsaw, 5,000, and check that the formula updated its result automatically once the data was entered inside the range the formula covers.
4. Swap the second condition for „Poznan” and compare the result.

Expected result: Sales plus Warsaw returns 12,400 before the new row and 17,400 after it. Sales plus Poznan returns 6,700.

Conditions act like a logical AND

Every additional range-and-condition pair in SUMIFS narrows the result, never widens it. Adding a third condition, even a very broad one, will not raise the sum above the result with two conditions, because a row has to satisfy every condition given at once.

EXERCISE 15

COUNTIFS and AVERAGEIFS

Goal: count orders that meet a condition and average their value, using the same argument logic as SUMIFS.

Table 6.2: Logistics department orders to count

Department	Value
Logistics	1,200
Logistics	980
Sales	4,500
Logistics	2,300

Steps:

1. In an empty cell, type `=COUNTIFS(A2:A5,"Logistics")`, where A is the Department column.
2. Check that the result is the number of Logistics rows, not the sum of their values.
3. In the next cell, type `=AVERAGEIFS(B2:B5,A2:A5,"Logistics")`, where B is the Value column.
4. Compare the result with the average calculated by hand from the three Logistics values.

Expected result: COUNTIFS returns 3, AVERAGEIFS returns 1,493.33 (rounded for display), i.e. $(1,200+980+2,300)/3$.

 One family, one argument structure

COUNTIFS has no range to sum, since it only counts rows, so it starts straight away with a range-and-condition pair. SUMIFS and AVERAGEIFS

begin with a range of values (to sum or average), followed by the same range-and-condition pairs as COUNTIFS.

☒ Conditional Functions

- ☐ I summed a column by a condition from one other column (SUMIFS).
- ☐ I added a second condition from another column and understood it narrows the result.
- ☐ I counted the rows meeting a condition (COUNTIFS).
- ☐ I averaged by condition and compared it with a manual calculation.

Sorting and Filtering

Putting a Large List in Order

The formulas from the previous chapters calculate correctly no matter the order of the rows. Sorting and filtering do not change the result of any calculation; they change which rows you see at once, a separate and equally necessary skill when working on a list of dozens or hundreds of orders.

EXERCISE 16

Multi-Level Sorting of an Order List

Goal: sort a list first by one column, and within repeated values, by a second column.

Table 7.1: Order list to sort (fragment)

Department	Date	Value
Marketing	2024-03-12	1,100
Sales	2024-03-05	4,500
Sales	2024-03-20	2,300
Logistics	2024-03-08	900

Steps:

1. Select the whole table and in the Data tab choose Sort.
2. Set the Department column as the first level, A to Z.
3. Click „Add Level” and set the Date column as the second level, oldest to newest.
4. Confirm and check that rows from the same department now sit next to each other, ordered chronologically within it.

Expected result: rows grouped alphabetically by department (Logistics, Marketing, Sales), and within the Sales department the row dated 2024-03-05 comes before the row dated 2024-03-20.

Sorting only one column

Selecting only the Value column before sorting, without the rest of the table, shuffles numbers out of sync with the departments and dates in the other columns, wrecking the rows. Before sorting, select the whole table or click one cell of an Excel Table, which pulls full rows along with it on its own.

EXERCISE 17

Custom Filter: a Range of Values

Goal: show only orders with a value from a given range, without writing a formula.

Steps:

1. Click the filter arrow in the Value column header.
2. Choose Number Filters, then Between.
3. In the first box type 1000, in the second 3000, keeping „is greater than or equal to” and „is less than or equal to”.
4. Confirm and check which rows became visible.

Expected result: only the Marketing row (1,100) and the Sales row dated 2024-03-20 (2,300) stay visible, the other two rows are hidden, and the filter arrow in the header changes icon, signaling an active filter.

Hidden, not deleted

Filtered-out rows are not removed from the worksheet, only temporarily hidden. Formulas that calculate over the whole column (such as SUM) still include the hidden rows, unless you use the SUBTOTAL function, which skips whatever the filter has hidden on its own.

EXERCISE 18

An Autofilter Across Multiple Columns at Once

Goal: apply filters to two columns at once and see that they work together, not interchangeably.

Steps:

1. Clear the previous filter from the Value column (Data, Clear).
2. In the Department column header, uncheck „Select All” and check only Sales.
3. In the Value column header, set the Between filter again, 1000 to 3000.
4. Check that only one row remains, meeting both conditions at once.

Expected result: only the Sales row, 2024-03-20, 2,300, is visible, because it has to meet both the department filter and the value filter at the same time, similar to the conditions inside SUMIFS from the previous chapter.

Biurplex's accounting department, quarter close

When closing a quarter, Biurplex's accounting department filters the order list by department and by a date range for a single month, instead of scanning the whole list by hand. Two filters applied to different columns narrow the list to a dozen-odd rows out of several hundred, before anything needs to be calculated or compiled into a report.

☒ Sorting and Filtering

- ☐ I sorted a table on two levels (department, then date).
- ☐ I applied a Number Filter of Between to one column.
- ☐ I cleared a filter before applying a new one.
- ☐ I applied filters to two columns at once and checked that they work together.

PivotTables

Summarizing Without Writing Formulas

A PivotTable calculates what we have been calculating so far with SUMIFS or COUNTIFS formulas, but it does it by dragging column names in a panel instead of typing syntax. You gain the most from it when the question you ask of the data changes every few minutes, for example from „how much did the Sales department sell” to „how much did each department sell in each city”.

EXERCISE 19

Your First PivotTable: Sales by Department

Goal: build a PivotTable showing the sum of sales broken down by department.

Steps:

1. Click any cell in the data and in the Insert tab choose PivotTable.
2. Confirm the data range and place the PivotTable on a new sheet.
3. In the PivotTable Fields panel, drag Department into the Rows area.
4. Drag Value into the Values area.

Table 8.1: Source data (Table tblSales)

Department	City	Value
Sales	Warsaw	12,400
Marketing	Krakow	3,200
Sales	Poznan	6,700
Logistics	Warsaw	1,800

Expected result: the PivotTable shows three rows (Logistics, Marketing, Sales) and, for each, the sum of the values, labeled by default „Sum of Value”, with a Grand Total row at the bottom.

 A field in Values sums by default

Excel sums a numeric field dragged into Values by default, but for a text field it counts occurrences instead of summing. If a numeric field shows up as „Count of...” instead of „Sum of...”, check whether the source column has even one cell stored as text, since that is enough to switch Excel’s aggregation.

EXERCISE 20

Grouping Dates in a PivotTable

Goal: group daily order dates into months, without changing the source data.

Steps:

1. In the source data, add a Date column with several different days in March and April.
2. Build a new PivotTable with Date in the Rows area and Value in Values.
3. Right-click any date in the PivotTable and choose Group.
4. In the grouping window, check Months (unchecking Days) and confirm.

Expected result: instead of individual days, the PivotTable's rows show „March” and „April”, each with the sum of all orders from that month.

Grouping requires valid dates

Grouping by month is unavailable or behaves incorrectly if the source column has even one date stored as text, for the same reason as in the chapter on formatting. Go back to the Chapter 2 exercise and check the Date column's alignment before grouping.

EXERCISE 21

A Calculated Column and Percentage Share

Goal: add a second values column to the PivotTable, shown as a percentage of the grand total, without a separate formula in the source worksheet.

Steps:

1. In the PivotTable with departments, drag the Value field into the Values area a second time, creating a second column.
2. Right-click the new column and choose Show Values As, then % of Grand Total.
3. Rename the second column's header to „Share %”.
4. Check that the three percentage values add up to 100%.

Expected result: the first column shows sums in dollars as before, the second column shows each department's share of the total, for example Sales around 63%, and the sum of all the shares comes to 100%.

**One field, two ways of showing it**

The same source field, Value, can be placed in the Values area more than once, each copy with a different display setting (sum, percentage, ranking). That is a faster way to get a second perspective on the same data than building a separate PivotTable from scratch.

**PivotTables**

- ☐ I built a PivotTable with one field in Rows and one in Values.
- ☐ I grouped daily dates into months without changing the source data.
- ☐ I added the same field to Values a second time with a different display setting.
- ☐ I checked that the percentage shares add up to 100%.

When Numbers Are Better Shown Than Listed

A table with three department totals reads quickly, but a list of twelve months or a comparison of five cities starts to take effort before anything stands out. A chart shifts that work onto the eye, not onto reading numbers one by one.

EXERCISE 22

A Column Chart Comparing Departments

Goal: build a column chart from a two-column table, without any extra formatting.

Table 9.1: Total sales by department (chart data)

Department	Value
Sales	27,200
Marketing	3,200
Logistics	1,800

Steps:

1. Select the two-column range with headers, Department and Value.
2. In the Insert tab, choose Column Chart, the first option in the 2-D group.
3. Click the chart area and use the plus button in the top-right corner to add Data Labels.
4. Change the chart title from the default „Value” to „Sales by Department, March”.

Expected result: three columns of different heights, tallest for Sales, with a number above each column and its own descriptive chart title.

💡 Select the headers together with the data

Selecting only the numbers, without the header row, makes Excel name the series „Series1” instead of Value, and number the categories 1, 2, 3 instead of department names. Always select together with the header, even though it looks like an unnecessary step.

EXERCISE 23

A Line Chart for a Trend Over Time

Goal: show change over time with a line chart, the right choice for data arranged chronologically.

Table 9.2: Monthly sales for the trend chart

Month	Value
January	18,400
February	21,100
March	27,200
April	24,600

Steps:

1. Select the Month and Value range with headers.
2. In the Insert tab, choose Line Chart.
3. Click the plus button next to the chart and add a Trendline, choosing Linear.
4. Check which way the trendline leans relative to the four data points.

Expected result: the line connecting the four points rises from January to March, drops in April, and the trendline runs through the middle of those swings, pointing to an overall upward direction despite the drop in the last month.

 A column chart for data over time

A column chart for twelve months is still readable, but a line better shows the pace of change itself, not just each month's height on its own. The choice of chart type depends on the question: „which month was highest” wants a column, „is sales growing” wants a line.

EXERCISE 24

A Chart From a PivotTable

Goal: build a chart linked to the PivotTable from Chapter 8 that recalculates along with it.

Steps:

1. Click any cell in the PivotTable with departments.
2. In the PivotTable Analyze tab, choose PivotChart.
3. Pick the Column type and confirm.
4. On the PivotTable, change the Department field's filter, removing Logistics, and check the chart.

Expected result: the PivotChart initially shows three columns, and after removing Logistics from the filter, only two remain automatically, with no manual fix to the chart.

A PivotChart Is a View, Not a Copy

A PivotChart has no data of its own, frozen in place; it only reads the PivotTable it is connected to. A filter, a grouping, or adding a field in the PivotTable changes the chart right away, the same way changing the source data changes the PivotTable itself.

☒ Charts

- ☐ I built a column chart with headers selected together with the data.
- ☐ I added data labels and my own descriptive chart title.
- ☐ I built a line chart with a trendline for data over time.

- ☐ I built a PivotChart and checked that it reacts to the PivotTable's filter.

Conditional Formatting

Color as a Signal, Not Decoration

Conditional formatting changes a cell's appearance based on its value, without touching the number itself. The three exercises in this chapter show a color scale for quickly comparing many numbers at once, a rule built on its own formula, and managing several rules layered onto the same range.

EXERCISE 25

Color Scales for Sales Values

Goal: apply a color scale to a column of numbers, so high and low values stand out right away.

Table 10.1: City sales to highlight

City	Value
Warsaw	12,400
Krakow	8,100
Poznan	6,700
Gdansk	15,900

Steps:

1. Select the Value column, without the header.
2. In the Home tab, choose Conditional Formatting, Color Scales, and the green-yellow-red variant.
3. Check which row came out the most green, and which the most red.
4. Change Poznan's value to 18,000 and watch how the whole scale recalculates.

Expected result: before the change, Gdansk is the most green and Poznan the most red; after changing Poznan's value, the colors across the whole column shift, because the scale is calculated relative to the current minimum and maximum of the selected range, not against fixed thresholds.

 A scale recalculates automatically

A color scale does not save its colors permanently; it recalculates them every time a value in the selected range changes. That is different from manually coloring cells, which would stay the same even after the number changed.

EXERCISE 26

A Rule With Its Own Formula: Overdue Deadlines

Goal: build a conditional-formatting rule based on a logical formula, instead of a ready-made template.

Table 10.2: Order fulfillment deadlines

Order	Deadline
ZB-201	2024-03-01
ZB-202	2024-04-15
ZB-203	2024-03-20

Steps:

1. Select the Deadline column.
2. In Conditional Formatting, choose New Rule, then „Use a formula to determine which cells to format”.
3. Type the formula `=B2<TODAY()`, where B2 is the first cell of the selected range.
4. In the Format button, set a red fill and confirm the rule.

Expected result: every deadline earlier than today’s date (read by the TODAY function) gets a red fill, while future deadlines stay unchanged; the result depends on the date you open the file on, and will change on its own from day to day.

 The reference in the rule's formula

A conditional-formatting rule's formula always refers to the first, top cell of the selected range, and Excel shifts that reference row by row on its own, similar to copying an ordinary formula. A reference to B2 typed while B2:B10 was selected checks every row below it, not just the second one.

EXERCISE 27

Data Bars and Managing Rules

Goal: apply a second rule (data bars) to the same range and set the priority between two conditional-formatting rules.

Steps:

1. Select the Value column from the first exercise in this chapter.
2. Add a second rule: Conditional Formatting, Data Bars, blue gradient.
3. Open Conditional Formatting, Manage Rules, and see both rules layered onto the same range.
4. Check „Stop If True” next to the data bar, move it to the top of the list, and check whether the color scale is still showing.

Expected result: before checking „Stop If True”, both effects show at once (bar and fill color); after checking that option on the rule listed higher, the second rule (the color scale) no longer applies.

💬 Too Many Rules on One Range

Layering four or five conditional-formatting rules onto the same column, instead of making it easier to read, blurs it, because a cell mixes several colors and bars at once. The Manage Rules panel lets you check how many rules actually act on a given range before adding another.

☑ Conditional Formatting

- ☐ I applied a color scale to a column of numbers and checked that it recalculates after a change.
- ☐ I built a rule with its own logical formula, referring to the top cell of the range.
- ☐ I added a second rule to the same range and opened the Manage Rules panel.
- ☐ I set rule priority with the „Stop If True” option.

Final Project: Monthly Department Report

One Data Set, Every Skill Learned

The chapters so far practiced one skill at a time. This chapter combines them on a single, larger data set: from cleaning up a raw list, through formulas and a PivotTable summary, to a chart and visual signaling. This is the task closest to what office work calls a „monthly report”.

EXERCISE 28

Preparing and Cleaning the Source Data

Goal: bring a raw order list to a state where you can safely calculate on it and summarize it.

Table 11.1: Raw order list, Sales department, April (fragment)

City	Date	Value	Status
Warsaw	04.02.2024	5,200	YES
Krakow	April 9	3,100	NO
Poznan	2024-04-15	4,800	YES
Gdansk	2024-04-22	2,900	YES

Steps:

1. Copy the data into a new sheet and check the alignment of every date, fixing any typed in as text (as in Chapter 2).
2. Convert the range into an Excel Table (Ctrl+T) and name it tblReport.
3. Add a Fulfillment Status column with an IF formula turning YES/NO in the Status column into „Fulfilled”/„Pending” (as in Chapter 4).
4. Sort the table by the Date column, oldest to newest.

Expected result: a tblReport table with corrected dates, all right-aligned, with an extra text-status column, and rows in chronological order.

Skipping the data cleanup

Jumping straight to a PivotTable on a raw list with uncorrected dates carries the error forward: the grouping by month from Chapter 8 will not work correctly until every date is recognized as a date rather than as text.

EXERCISE 29

Summarizing With Formulas and a PivotTable

Goal: calculate key totals with conditional formulas and build a PivotTable as a second, flexible view of the same data.

Table 11.2: Report questions and the functions that answer them

Report question	Function
How much did Warsaw and Gdansk sell combined	SUMIFS
How many orders are still pending	COUNTIFS
What is the average order value	AVERAGEIFS

Steps:

1. Below tblReport, use SUMIFS to calculate the value sum for Warsaw, a separate formula for the sum for Gdansk, and add the two together in a third cell.
2. Use COUNTIFS to count the rows with „Pending” status.
3. Build a new PivotTable from tblReport, with City in Rows and Value in Values.
4. Add a second Value field to the same PivotTable, shown as % of Grand Total (as in Chapter 8).

Expected result: the Warsaw-plus-Gdansk sum from the formulas matches the sum of those two rows read from the PivotTable, the number of pending orders is 1 (Krakow), and the PivotTable’s percentage column adds up to 100%.

 Two independent calculations as a check

Calculating the same sum with a formula and with a PivotTable separately is not duplicated work, but a way to check yourself: if the two results differ, there is an error in one of the two approaches, before the report goes any further.

EXERCISE 30

Visualization and Signaling: a Chart and Conditional Formatting

Goal: finish the report with a chart built from the PivotTable and conditional formatting that flags orders needing attention.

Steps:

1. Select the PivotTable from the previous exercise and insert a Column PivotChart from it (as in Chapter 9).
2. Add the title „Sales by City, April” and data labels.
3. On tblReport, apply conditional formatting to the Fulfillment Status column with a text rule: cells equal to „Pending” get a red fill.
4. Apply a color scale to the Value column, so the highest and lowest orders stand out without reading the numbers one by one.

Expected result: a PivotChart showing four cities with a descriptive title, one red row (Krakow, pending) visible right away in the table, and the Value column with a smooth color scale from the lowest to the highest amount.

 A Report as a Chain, Not a Set of Separate Tricks

The data cleanup in this chapter's first exercise made the correct formulas and grouping in the second one possible, and together they produced a reliable chart and sensible color signaling in the third. A mistake made right at the start, such as a date stored as text, travels through that entire chain and surfaces only at the end, usually in the least obvious place.

 Final Project

- ☐ I cleaned up the raw list and converted it into an Excel Table.
- ☐ I calculated the same sums with a conditional formula and with a PivotTable, and compared the results.
- ☐ I added a percentage column to the PivotTable.
- ☐ I built a PivotChart and applied two conditional-formatting rules.

12

Summary and Next Steps

A Map of Thirty Exercises

Thirty exercises in this workbook fall into four skill groups that together cover most of the daily work with a data worksheet in an office.

Table 12.1: Skills by exercise group

Group	Chapters
Data order and formulas	Formatting, formulas, IF
Joining and aggregating tables	VLOOKUP/XLOOKUP, SUMIFS
Working with large lists	Sorting, filtering, PivotTables
Presenting the result	Charts, conditional formatting

The final project in Chapter 11 showed that these groups are not independent tricks to be pulled out one at a time, but successive stages of a single piece of work on the same data set. The data order from the first group has to be correct before the second group can calculate anything from it, and the third and fourth groups only make sense on data that was already calculated properly.

What Comes Next When This Workbook Is Done

Two skills naturally extend what you have practiced, and it is worth reaching for them once this workbook stops offering new challenges.

- Macros and recording actions (Developer, Record Macro) automate repeated steps, for example applying the same formatting to a new file every month.
- Power Query (Data tab, Get Data) cleans up and joins data from multiple files or sources before it reaches the worksheet, replacing part of the manual cleanup from Chapter 2 with one, repeatable step.

💡 Go back to a specific chapter, not to the start

When a task at work calls for XLOOKUP or a PivotTable, go back to the right chapter of this workbook and repeat one exercise on your own data, instead of working through the whole workbook again. Every chapter stands on its own and does not require remembering the previous ones, apart from the data order from Chapter 2.

Before You Close This Workbook

Twenty-seven of the thirty exercises worked on one fictional company, Biurplex, with the same departments and cities repeating across chapters. If you went through them on your own worksheet, not just by reading the descriptions, you now have a file with working formulas, a PivotTable, and a chart that you can open again as your own template the first time a similar task comes up at work.

 The One Skill Worth Remembering First

Before returning to any function from this workbook, check the alignment of the dates and numbers in the column you are about to work on. That one, first step from Chapter 2 breaks most often and most quietly, and fixing it takes a moment, compared with the time spent hunting for why a formula or a PivotTable gives a wrong result.

 Before Closing the Workbook

- ☐ I have a saved file with my own versions of at least a few exercises, not just the reading.
- ☐ I know which chapter to return to for a specific function, without browsing the whole workbook.
- ☐ I understand the difference between VLOOKUP and XLOOKUP well enough to choose between them.
- ☐ I know the order of stages from the final project: data order, formulas, summary, presentation.