VIDEO TO COURSE · SAMPLE OUTPUT

Excel Formulas for Beginners

Clean, label, total and count your data with 19 everyday functions

4 modules | 12 lessons | quizzes + answer keys | assignments | action plan

Based on “Formulas in Excel | Excel Tutorials for Beginners” by Alex The Analyst · youtube.com/watch?v=XRPyj7cKVsQ · Creative Commons Attribution licence (CC BY)
Built with the Video to Course Prompt Studio. Lesson text as the AI returned it; only the layout was added.

Contents

Course Map

Welcome: Excel Formulas for Beginners

Module 1: Write your first formulas

Find the Highest and Lowest Values with MAX and MIN

Total and Count a Column with SUM and COUNT

Label Every Row with IF

Module 2: Make formulas check conditions

Sort Rows into Several Labels with IFS

Total and Count with One Condition: SUMIF and COUNTIF

Total and Count with Several Conditions: SUMIFS and COUNTIFS

Module 3: Clean and build text

Check and Clean a Text Column with LEN and TRIM

Pull Characters from Either End with LEFT and RIGHT

Join Names and Build Emails with CONCATENATE

Module 4: Fix and use dates

Turn Dates into Text with TEXT and Pull Out the Year

Swap Date Separators with SUBSTITUTE

Count Calendar and Working Days with DAYS and NETWORKDAYS

Action Plan

Listing And Launch Notes

FROM SECTION A

Course Map

Module	Lesson	Objective	Skills covered	Minutes
0. Welcome	m0-l0 Welcome	Decide whether the course fits you and build the practice sheet every lesson uses	none	3
1. Write your first formulas	m1-l1 Find the highest and lowest values with MAX and MIN	Write MAX and MIN formulas on a date column and a number column and check each result against the data	1, 7, 2, 3	5
1. Write your first formulas	m1-l2 Total and count a column with SUM and COUNT	Total a number column with SUM, count it with COUNT, and confirm the count against the bottom of the window	18, 21	4
1. Write your first formulas	m1-l3 Label every row with IF	Write an IF formula that labels every row one of two ways, with the boundary value chosen on purpose	4	5
2. Make formulas check conditions	m2-l1 Sort rows into several labels with IFS	Write an IFS formula that labels each category in a column, and handle rows that match nothing	5	5
2. Make formulas check conditions	m2-l2 Total and count with one condition: SUMIF and COUNTIF	Answer "how much" and "how many" questions with one condition using SUMIF and COUNTIF	19, 22	5
2. Make formulas check conditions	m2-l3 Total and count with several conditions: SUMIFS and COUNTIFS	Answer questions with two or more conditions using SUMIFS and COUNTIFS in the correct argument order	20, 23	6
3. Clean and build text	m3-l1 Check and clean a text column with LEN and TRIM	Flag entries of the wrong length with LEN and strip stray spaces with TRIM	6, 13	5
3. Clean and build text	m3-l2 Pull characters from either end with LEFT and RIGHT	Extract a set number of characters from the start or end of a value	8, 9	5
3. Clean and build text	m3-l3 Join names and build emails with CONCATENATE	Join columns into a full name and a patterned email address	14, 15	5
4. Fix and use dates	m4-l1 Turn dates into text with TEXT and pull out the year	Tell a real date from a text date, convert dates with TEXT, keep the values, and take the year with RIGHT	11, 12, 10	6
4. Fix and use dates	m4-l2 Swap date separators with SUBSTITUTE	Replace every separator, or only a chosen one, in a column of text dates	16, 17	5
4. Fix and use dates	m4-l3 Count calendar and working days with DAYS and NETWORKDAYS	Count the calendar days and working days between a start and end date, with your own holiday list	24, 25	5
After the course	Action plan	Clean and summarize a new list from start to finish without reopening the lessons	all, in order	n/a

Every skill (1 to 25) appears exactly once. The first lesson gives an instant, checkable answer from a single formula. The hardest work, multi-condition totals with a new argument order and the date conversion, sits in the middle and second half, after the learner has written a dozen simpler formulas. Total: about 62 minutes of lessons plus assignments.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Welcome: Excel Formulas for Beginners

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Decide whether this course fits you, and build the practice sheet that every lesson uses.

2. Why it matters

A formula only makes sense when you can see it working on rows in front of you. Ten minutes spent building a small practice sheet now means every lesson ends with a result you can check, not a formula you hope is right.

3. The teaching

This course is built on a tutorial by Alex The Analyst called "Formulas in Excel | Excel Tutorials for Beginners". Alex is a data analyst who teaches on YouTube, and he is clear about what his tutorial is and isn't. He doesn't try to show every formula. He shows, in his words, "some of my favorites and the ones that i found the most useful", and he says that almost all of them are ones he "learned while i was on the job". Everything you learn here is Alex's method, taught by me with permission, and his name is on every idea that came from him.

Alex works on a small table of employees whose names you may recognize from the TV series The Office: Michael Scott, Jim Halpert, Pam Beesly, Toby Flenderson, Stanley and Kevin. Each person has an ID, a first and last name, an age, a gender, a job title, a salary, a start date and an end date. He keeps one sheet tab per function and works through 19 of them.

Who this is for. You already use Excel to look at lists: staff, customers, orders, dates. You can open a workbook, type in a cell, switch between sheet tabs and select a range by dragging across it. You have written few formulas, or none. You might be aiming at data analyst work, or you might just be tired of doing by hand what a formula could do in a second.

What you will be able to do. By the end you can clean a messy list in Excel and answer questions about it with 19 everyday functions: find the highest and lowest values, label rows with IF and IFS, total and count only the rows that meet your conditions, fix stray spaces, split and join text, convert and reshape dates, and count the days or working days between two dates.

How the course is laid out. Four modules, twelve lessons, about five minutes each.

- Module 1, Write your first formulas: MAX, MIN, SUM, COUNT and IF.
- Module 2, Make formulas check conditions: IFS, SUMIF, COUNTIF, SUMIFS and COUNTIFS.
- Module 3, Clean and build text: LEN, TRIM, LEFT, RIGHT and CONCATENATE.
- Module 4, Fix and use dates: TEXT, SUBSTITUTE, DAYS and NETWORKDAYS.

The order is not the order Alex used. It is the order you need: each lesson only uses what came before it. After the last lesson there is one action plan you can follow without opening the lessons again.

How to use it. Build the practice sheet in the assignment below, then do the lessons in order and type every formula yourself. Alex's own advice is the same: "i highly recommend trying these formulas out for yourself so you can get a feel for how they work". He mentions saving his workbook to GitHub for download. That link isn't part of this course, so your practice sheet stands in for it.

Cell addresses in the lessons are illustrative. Alex selects ranges by dragging and never reads out column letters, so the course uses one layout for your practice sheet: A ID, B First Name, C Last Name, D Age, E Gender, F Job Title, G Salary, H Start Date, I End Date, headers in row 1 and data in rows 2 to 10.

You will see two labels. **[General knowledge, not from the video. Check before publishing.]** marks Excel facts I have added because Alex doesn't say them, such as where the quotation marks go. **[Instructor's method, not from the video.]** marks a checking or practice technique I teach on top of Alex's demonstration. Everything without a label comes from Alex.

[General knowledge, not from the video. Check before publishing.] Alex never says which version of Excel he uses. When he hands a whole column to a function such as IF or LEN, he gets an answer in every row. That is how current Microsoft 365 Excel behaves. In older versions you write the formula for the first row and copy it down, and each lesson shows that route too. IFS needs Excel 2019 or later, and DAYS needs Excel 2013 or later.

What it does not cover. Every other Excel function, including lookups such as VLOOKUP, averages, sorting, filtering, charts and pivot tables. Alex's tutorial doesn't teach them, so neither does this course. It makes no promise about jobs or pay. Alex's video was sponsored by Udemy; that message is not part of this course.

4. Worked example

Illustrative. Dana runs the staff schedule for a twelve-person clinic and keeps staff details in a spreadsheet. Every month she counts by eye how many people earn over a pay threshold and retypes start dates that arrive in three different formats. She checks this page: she can open workbooks and select ranges, she has never written a formula, and the course covers counting with conditions and fixing dates. She builds the practice sheet with nine made-up staff and the four traps listed below. The one thing that would have gone wrong: she nearly practiced on the live staff file. One mistyped formula pasted over a column would have changed real records, so she works on a copy.

5. Common mistake

Watching the formulas instead of typing them. Formulas look obvious when someone else types them and then fall apart on your own data, usually at a comma or a quotation mark. Do instead: type every formula in every lesson into your practice sheet, and check each result against the rows before moving on.

6. Summary

Alex The Analyst's tutorial shows 19 Excel functions he uses as a data analyst, on a small employee table. This course teaches them in four modules: first formulas, conditions, text, then dates. Each lesson takes about five minutes and ends with a task on your own practice sheet. Cell addresses are illustrative. Labeled passages mark added Excel facts and the instructor's methods; check those before relying on them.

7. Quiz

1. Which learner is this course built for? A. Someone who has never opened Excel and needs to learn what a cell is B. Someone who uses Excel lists and wants to start answering questions with formulas C. An experienced analyst looking for advanced lookups and pivot tables D. Someone who wants to learn SQL and Python

2. A lesson shows `=IF(D2>30,"Old","Young")` but your ages are in column F. What do you do? A. Move your ages into column D before you start B. Type the formula with F2 in place of D2, because the addresses are illustrative C. Skip the lesson, because it was written for different data D. Leave the formula as written and accept a wrong answer

3. You see "[General knowledge, not from the video. Check before publishing.]" above a paragraph. What does that tell you? A. Alex said it and you can rely on it as written B. The paragraph is optional and can be skipped C. It is an added Excel fact that should be checked against your own version D. It is out of date and should be ignored

Answer key

1. **B.** That is the viewer Alex speaks to. A is wrong: the course assumes you can already open a workbook and select a range. C is wrong: lookups and pivot tables aren't covered. D is wrong: Alex mentions SQL and Python only in passing.
2. **B.** The addresses follow one illustrative layout; swap in your own columns. A is wrong: you can, but there's no need. C is wrong: the method works on any column. D is wrong: D2 would point at the wrong data.
3. **C.** It marks a fact Alex didn't state. A is wrong: the label exists because Alex didn't say it. B is wrong: these are often the details that make a formula work. D is wrong: it may well be accurate; the point is to check.

8. Assignment: build your practice sheet

Time: 15 minutes.

Make a copy-safe workbook called "Formula practice". On the first sheet, type the nine headers in row 1 (A ID, B First Name, C Last Name, D Age, E Gender, F Job Title, G Salary, H Start Date, I End Date) and nine rows of made-up people in rows 2 to 10. Build in four traps that later lessons rely on:

1. One person aged exactly 30.
2. One person on a salary of exactly 50,000.
3. One last name typed with a space after it.
4. At least four different job titles.

Use IDs from 1001 to 1009, and type the dates as ordinary dates (for example 3/14/2012).

[Instructor's method, not from the video.] The traps mirror the edge cases in Alex's own table: his IF example leaves out age 30, his SUMIF example leaves out a salary of exactly 50,000, and his TRIM example cleans names with stray spaces.

Done checklist

- [ ] Nine headers in row 1, in the order given
- [ ] Nine rows of data in rows 2 to 10, IDs 1001 to 1009
- [ ] One age of exactly 30 and one salary of exactly 50,000
- [ ] One last name with a trailing space (click the cell and check the end of the text)
- [ ] At least four different job titles, and every start date earlier than its end date

Sample answer (illustrative; every name and value invented)

ID	First Name	Last Name	Age	Gender	Job Title	Salary	Start Date	End Date
1001	Rosa	Delgado	41	Female	Manager	68000	3/14/2009	6/30/2021
1002	Tom	Reyes	30	Male	Salesman	50000	8/2/2015	1/15/2022
1003	Nadia	Park␣	27	Female	HR	47000	11/2/2016	4/1/2023

(␣ marks the trailing space. The other six rows follow the same pattern.)

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Find the Highest and Lowest Values with MAX and MIN

Module 1: Write your first formulas. By the end of this module you can write a formula from scratch, summarize a column four ways, and label every row with a rule.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Write MAX and MIN formulas on a date column and a number column, and check each result against the data.

2. Why it matters

"Who started first?" and "what's the top salary?" are the questions people ask about a list before any other. A formula answers them in a second, on nine rows or nine thousand, and the steps you learn here are the same steps for every formula in the course.

3. The teaching

Alex starts with MAX and MIN because, as he puts it, they are "super self-explanatory but again one that you're going to use all the time". That makes them the right place to learn the mechanics, because you can see straight away whether the answer is right.

Step 1: type an equals sign. Click an empty cell and type `=`. Alex's explanation is the whole rule: "equal means i am now creating a formula". Without it, Excel treats what you type as ordinary text.

Step 2: name the function and press Tab. Type `max`. A list of matching functions appears. Alex presses Tab, and Excel fills in the name and the opening parenthesis for you: `=MAX(`.

Step 3: read the hint. As soon as the parenthesis opens, Excel shows the inputs it expects. For MAX it reads "number1, number2" and so on. You will use this hint for every new function in the course, so get in the habit of glancing at it. Inputs shown in square brackets are optional, a point Alex makes later when he reaches SUBSTITUTE.

Step 4: give it a range. Alex doesn't type numbers in. He drags from the first cell of the column to the last, and Excel writes the range for him. On your practice sheet the start dates are in H2 to H10:

```
=MAX(H2:H10)
```

Step 5: close and enter. "You don't have to close the parentheses but you can," Alex says, and he does. Press Enter.

[General knowledge, not from the video. Check before publishing.] Excel adds a missing closing parenthesis for you in a simple formula like this. Closing it yourself is still the better habit, because longer formulas later in the course have several parentheses and Excel can't always guess where one belongs.

What MAX and MIN return. MAX gives the largest value in the range and MIN gives the smallest. On dates, the largest is the latest. Alex runs MAX on his start dates and checks it by eye: "2013 was the last year and this one is actually

the latest in that year". Then MIN on the same range returns December 7, 1995. "We can see that that is correct," he says, "so michael scott started in 1995 the earliest of all the employees."

Then he does the same on salaries. MAX shows that the highest salary is 65,000, and the person on it is Michael Scott. MIN shows that Pam Beesly earns the least.

Notice what Alex did after each formula: he looked at the data to confirm it. Notice too what the formula didn't tell him. MAX returns the value, not the person. Alex finds the name by looking along the row.

When the answer looks wrong. On the salary MAX, Alex's result came out looking odd, and his first thought was that he had picked the wrong range. He hadn't. The cell he typed into was formatted as a date, so the salary was being displayed as one. In his words, "this column was a date range or a date column for whatever reason let me get rid of that". Once he cleared the format, the right number showed.

[General knowledge, not from the video. Check before publishing.] To clear a cell's format, select it and set Home > Number Format to General or Number. The formula itself doesn't change.

Undo. Later in the tutorial Alex makes a slip and fixes it with Ctrl+Z: "i did ctrl z to undo that if you didn't know how to do that". Use it freely while you practice.

[Instructor's method, not from the video.] Check every new formula once by eye, the way Alex does: find the value it returned in the column, and make sure nothing above or below it in the list beats it.

4. Worked example

Drawn from the source. Alex wants to know who has been with the company longest and who earns most. In an empty cell he types `=MIN(`, drags down the start date column, closes the parenthesis and presses Enter. The cell shows December 7, 1995. He scans the column, finds it on Michael Scott's row, and confirms it is the earliest. Next he types MAX over the salary column. The result looks wrong, so he checks the range, finds it is fine, and sees the real problem: the output cell is formatted as a date. He clears the format and the cell shows 65,000, Michael Scott's salary.

The one thing that would have gone wrong: had he trusted his first instinct, he would have rewritten a correct formula. The formula was right; the display was wrong.

5. Common mistake

Rewriting a formula because its result looks strange. A salary that shows as a date, or a date that shows as a five-digit number, is usually a format problem in the cell, not a formula problem. Do instead: check the range first, then the cell's format, and only then the formula.

6. Summary

Start every formula with "=", type the function name, press Tab, read the input hint, drag across the range and press Enter. MAX returns the largest value in a range and MIN the smallest; on dates that means the latest and earliest. Alex's MIN on start dates returned December 7, 1995 (Michael Scott), and his MAX on salaries returned 65,000. If a result looks odd, check the cell's format before rewriting anything. Ctrl+Z undoes a slip.

7. Quiz

1. Your start dates are in H2:H10. Which formula returns the most recent start date? A. `=MIN(H2:H10)` B. `=MAX(H2:H10)` C. `MAX(H2:H10)` typed without the equals sign D. `=MAX(H2)`

2. You type `=MAX(G2:G10)` over a salary column and the result looks like a date. The range is correct. What do you do first? A. Change MAX to MIN B. Set the result cell's format to General or Number C. Delete the salary column and retype it D. Add a second closing parenthesis

3. You want to know which employee has the lowest salary. You have typed `=MIN(G2:G10)` and it returned a number. What is the next step? A. Nothing; MIN has already told you the person's name B. Find that value in the salary column and read the name on the same row C. Change the formula to `=MAX(G2:G10)` D. Press Ctrl+Z to see the name

Answer key

1. **B.** The latest date is the largest value. A is wrong: MIN returns the earliest date. C is wrong: without "=" Excel stores it as text. D is wrong: it looks at one cell, not the column.
2. **B.** A correct formula showing an odd value points to the cell's format. A is wrong: it answers a different question. C is wrong: the data is fine. D is wrong: an extra parenthesis would cause an error, not fix the display.
3. **B.** MIN and MAX return values, not names. A is wrong: the formula returns only the number. C is wrong: MAX returns the highest. D is wrong: Ctrl+Z undoes your last action.

8. Assignment: your summary block

Time: 15 minutes.

On your practice sheet, start a summary block under the data, in row 13. In A13 to A16, type four labels: Latest start, Earliest start, Highest salary, Lowest salary. Next to each, in column B, write the matching MAX or MIN formula. In column C, write the name of the person the value belongs to, found by looking along the row. Later lessons add more rows to this block.

Done checklist

- [ ] Four formulas, each starting with "=" and using the whole column range (rows 2 to 10)
- [ ] The two date results display as dates and the two salary results as numbers
- [ ] Each result has been found in the data and a name written beside it
- [ ] The person who started latest is not also listed as starting earliest

Sample answer (illustrative; values from the sample practice sheet)

> Latest start | `=MAX(H2:H10)` | 4/18/2019 | Lena Ford Earliest start | `=MIN(H2:H10)` | 3/14/2009 | Rosa Delgado Highest salary | `=MAX(G2:G10)` | 68000 | Rosa Delgado Lowest salary | `=MIN(G2:G10)` | 41000 | Sam Oduya

---

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Total and Count a Column with SUM and COUNT

Module 1: Write your first formulas.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Total a number column with SUM, count it with COUNT, and confirm the count against the figure at the bottom of the Excel window.

2. Why it matters

"How much in total?" and "how many?" sit underneath almost every report you will ever be asked for. These two formulas are also the base for SUMIF and COUNTIF in Module 2, which only total or count the rows you choose.

3. The teaching

You already know the mechanics from MAX and MIN: equals sign, function name, Tab, range, Enter. SUM and COUNT use exactly the same steps. Alex calls them equally easy: "count and sum are kind of on the same level of difficulty they're both pretty beginner". He moves through them quickly, so this lesson is short too.

SUM. "Sum means add," as Alex says, and SUM "just adds up all the numbers in range". He runs it over the salary column to get what all the salaries come to together. On your practice sheet:

```
=SUM(G2:G10)
```

That's the whole formula. Alex calls SUM "probably the most basic formulas that you can do", and he's right. The reason it gets its own lesson is what comes next: in Module 2 you will add a condition to it.

COUNT. COUNT, in Alex's words, "is just going to give you a count of how many cells are there", and "it's not going to add it". Alex selects a range and gets 9.

```
=COUNT(G2:G10)
```

Check the count against the window. Alex then does something worth copying. He highlights the same cells with the mouse and reads the count Excel shows at the bottom of the window. It says 9, the same as his formula. That's a ten-second check that your formula covered the cells you meant it to.

[General knowledge, not from the video. Check before publishing.] Two details matter here, and Alex doesn't mention either.

First, COUNT only counts cells that contain numbers. Dates count, because Excel stores them as numbers underneath, as you'll see in Module 4. Text does not. Run COUNT over a column of names and you get 0. To count every non-empty cell, text included, use COUNTA, which works the same way: `=COUNTA(B2:B10)`.

Second, the "Count" at the bottom of the window counts every non-empty cell, not just numbers. On a clean salary column the two agree. If someone has typed "n/a" or "TBC" into a salary cell, COUNT will come out one lower than the

figure at the bottom. When they disagree, that disagreement is useful: it tells you a number column contains something that isn't a number.

[Instructor's method, not from the video.] Once you have a total and a count in the same summary block, you can divide one by the other in a third cell to get an average salary, for example `=B17/B18` if SUM is in B17 and COUNT in B18. It's a quick check on whether the total looks sensible.

4. Worked example

Drawn from the source, then extended (illustrative). Alex runs COUNT over his range and gets 9. He highlights the same cells, reads the count at the bottom of the window, and sees 9 there too. The formula and the data agree.

Now suppose, as an illustration, that one salary on your practice sheet had been typed as "TBC" while waiting for HR. COUNT would return 8, and the bottom of the window would still say 9. SUM would quietly leave the "TBC" out of the total.

The one thing that would have gone wrong: without the check, the total would look fine and be short by one person's salary, and nobody would know why.

5. Common mistake

Using COUNT to count names or job titles. COUNT returns 0 or a number that's too low, because it ignores text. Do instead: use COUNT for number and date columns, use COUNTA for text, and check either one against the count at the bottom of the window.

6. Summary

SUM adds every number in a range: `=SUM(G2:G10)`. COUNT counts the cells in a range that hold numbers, dates included; it adds nothing up. Alex checks his COUNT result, 9, by highlighting the same cells and reading the count at the bottom of the window. That count includes text, so if it is higher than COUNT, a number column has text in it. For text columns, use COUNTA.

7. Quiz

1. Your salaries are in G2:G10. Which formula gives the total salary bill? A. `=COUNT(G2:G10)` B. `=SUM(G2:G10)` C. `=MAX(G2:G10)` D. `=SUM(G2)`

2. `=COUNT(G2:G10)` returns 8, but when you highlight G2:G10 the bottom of the window says Count: 9. What does that most likely mean? A. COUNT is broken and should be retyped B. One cell in the range holds text instead of a number C. One salary is zero D. The range is missing a row

3. You need the number of employees, and you decide to count the Last Name column, B2:B10. Which formula works? A. `=COUNT(B2:B10)` B. `=SUM(B2:B10)` C. `=COUNTA(B2:B10)` D. `=MAX(B2:B10)`

Answer key

1. **B.** SUM adds the values. A is wrong: it counts cells instead of adding them. C is wrong: it returns only the highest salary. D is wrong: it adds a single cell.

2. **B.** The window count includes text and COUNT doesn't, so one cell isn't a number. A is wrong: COUNT is doing exactly what it should. C is wrong: zero is a number, so COUNT would include it. D is wrong: both counts use the same highlighted range.
3. **C.** COUNTA counts non-empty cells, text included. A is wrong: COUNT ignores text and would return 0. B is wrong: SUM adds numbers and names have none. D is wrong: MAX returns a value, not a count.

8. Assignment: total and count the salaries

Time: 10 minutes.

Add three rows to your summary block: Total salary (`SUM`), Number of salaries (`COUNT`), and Average salary (the total divided by the count). Then type "TBC" over one salary, note what changes, and press Ctrl+Z to put the salary back.

Done checklist

- [ ] SUM and COUNT both cover rows 2 to 10 of the salary column
- [ ] The COUNT result matches the count at the bottom of the window when the column is highlighted
- [ ] The average cell refers to the total and count cells, not to typed numbers
- [ ] A note records what COUNT and SUM showed while "TBC" was in the column

Sample answer (illustrative)

> Total salary | `=SUM(G2:G10)` | 459000 Number of salaries | `=COUNT(G2:G10)` | 9 Average salary | `=B17/B18` | 51000 With "TBC" in G4: COUNT 8, window count 9, total fell by 47,000. Undone with Ctrl+Z.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Label Every Row with IF

Module 1: Write your first formulas.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Write an IF formula that labels every row in a column one of two ways, with the boundary value placed on purpose.

2. Why it matters

Most lists need sorting into two groups before anyone can use them: over or under a limit, in or out, yes or no. IF does that for every row at once, and it's the idea behind SUMIF, COUNTIF and IFS later in the course.

3. The teaching

Alex sums IF up in four words: "if this then that". Every IF formula asks one question about a cell, gives one answer if the question is true and another if it's false.

Open the parenthesis after `=IF` and the hint shows the three inputs Alex walks through:

1. **logical_test**: the question, a comparison such as "equal to" or "greater than".
2. **value_if_true**: what to show when the answer is yes.
3. **value_if_false**: what to show when the answer is no.

Commas separate the three.

Alex's example. He takes the age column and asks whether each person is over 30. "If they're greater than 30 we're going to call them old and then if it is false so if they're younger than 30 ... we're going to say young." On your practice sheet, with ages in column D, the formula for the first person is:

```
=IF(D2>30,"Old","Young")
```

Every row gets one of the two labels. That's the defining feature of IF: there's always an answer, because the false value catches everything the test doesn't.

[General knowledge, not from the video. Check before publishing.] Three things the captions can't show you:

- Text answers go in quotation marks: `"Old"`, `"Young"`. Numbers don't need them.
- The comparison signs are `>` (greater than), `>=` (greater than or equal to), `<` (less than), `<=` (less than or equal to), `=` (equal to) and `<>` (not equal to).
- Alex selects the whole age range inside his IF and gets a label in every row at once. That works in current Microsoft 365 Excel, where the results "spill" down the column, and it needs empty cells below to spill into. In other versions, write the formula for row 2 only, then copy it down: select the cell and double-click the small square at its bottom-right corner, or drag that square down to row 10. Excel moves D2 to D3, D4 and so on for you.

The boundary. Here is the point Alex stops to make, and it's the one that catches people. "This is something where you need to specify if you want 30 and over or over 30. We chose over 30. So 30 is not included in that." Someone aged exactly 30 is not greater than 30, so the test is false, and they are labeled "Young".

Neither choice is wrong. What's wrong is not choosing. Before you type the sign, say the rule out loud with the boundary in it: "over 30 is Old" or "30 and over is Old". Then pick the sign that matches:

- "over 30" is `D2>30`
- "30 and over" is `D2>=30`

[Instructor's method, not from the video.] Once the labels are in, find the boundary row on purpose. Your practice sheet has someone aged exactly 30 for this reason. Check that their label matches the rule you said out loud. If it doesn't, change the sign, not the data.

You don't need a second IF to do the opposite test. As Alex says after building his, "we don't actually need two of these that's pretty self-explanatory": one IF already sorts every row into one group or the other.

4. Worked example

Drawn from the source. Alex labels his employees by age. He types `=IF(`, selects the age column, types `>30`, a comma, "old", a comma, "young", and closes the parenthesis. Everyone over 30 shows "old" and everyone else shows "young". He then points out that 30 itself falls into "young", because he chose "over 30" and not "30 and over".

The one thing that would have gone wrong: if the rule he had been given was "30 and over", one person aged 30 would carry the wrong label, and every count built on those labels later would be off by one. The fix is a single character: `>=` instead of `>`.

5. Common mistake

Typing `>` when the rule means "and over", or `<` when it means "and under". The formula works, the labels look right, and only the boundary row is wrong. Do instead: say the rule aloud with the boundary in it, pick the sign to match, and check the one row that sits exactly on the boundary.

6. Summary

IF asks one question of a cell and returns one answer if it's true and another if it's false: `=IF(D2>30,"Old","Young")`. Text answers go in quotation marks. Every row gets a label. Decide whether the boundary is included before choosing the sign: `>30` leaves 30 out, `>=30` keeps it in. Alex's example labels anyone over 30 "old", so a 30-year-old is "young". In versions without spilling, write the formula for row 2 and copy it down.

7. Quiz

1. Your rule is "a salary of 50,000 or more is High; anything else is Standard". Salaries are in G. Which formula is correct for row 2? A. `=IF(G2>50000,"High","Standard")` B. `=IF(G2>=50000,"High","Standard")` C. `=IF(G2<50000,"High","Standard")` D. `=IF(G2>=50000,High,Standard)`

2. Using `=IF(D2>30,"Old","Young")`, what label does a person aged exactly 30 get? A. Old B. Young C. An error, because 30 is not covered D. Nothing; the cell stays empty

3. You copy `=IF(D2>30,"Old","Young")` from row 2 down to row 10. What does the formula in row 7 test? A. D2 B. D7 C. D10 D. The whole column D

Answer key

1. **B.** "Or more" includes 50,000, so the sign is `>=`. A is wrong: it leaves exactly 50,000 out. C is wrong: it labels the low salaries High. D is wrong: text answers need quotation marks.
2. **B.** 30 is not greater than 30, so the test is false. A is wrong: that would need `>=30`. C is wrong: IF always returns one of its two values. D is wrong: the false value fills the cell.
3. **B.** Copying down moves the reference one row at a time. A is wrong: the reference isn't fixed. C is wrong: row 7 points at row 7. D is wrong: the copied formula looks at one cell.

8. Assignment: an Age Group column

Time: 15 minutes.

Add a header "Age Group" in J1. Write the rule in words in a note beside the sheet, including the boundary. Then write the IF formula for the whole column, either as one spilled formula over D2:D10 or as a formula in J2 copied down to J10. Find the person aged exactly 30 and check their label against your written rule.

Done checklist

- [ ] The rule is written in words, including whether 30 is in or out
- [ ] Every row from 2 to 10 shows one of the two labels
- [ ] The sign in the formula matches the written rule
- [ ] The person aged exactly 30 carries the label the rule says they should

Sample answer (illustrative)

> Rule: 30 and over is "Senior staff"; under 30 is "Junior staff". 30 is in. J2: `=IF(D2>=30,"Senior staff","Junior staff")`, copied to J10. Check: Tom Reyes, 30, shows "Senior staff". Correct.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Sort Rows into Several Labels with IFS

Module 2: Make formulas check conditions. By the end of this module you can label rows by several categories and total or count only the rows that meet one condition or several.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Write an IFS formula that gives each category in a column its own label, and make sure rows that match no category still get a label.

2. Why it matters

Real columns rarely split into just two groups. Job titles, regions and product types have four, six or twelve values, and IFS lets one formula label all of them without stacking IF inside IF.

3. The teaching

IF gave you one test with a true answer and a false answer. Alex describes IFS as "a little bit different": "you can put multiple conditions". Open the parenthesis after `=IFS` and the hint shows the pattern: logical_test1, value_if_true1, logical_test2, value_if_true2, and so on.

So you write pairs. Each pair is a question and the answer to give if that question is true. As Alex puts it, "you're going to see this pattern right if this is our conditional or logical test so if this is true this is what's going to be returned".

What IFS doesn't have. IF always had a false value. IFS doesn't. In Alex's words, "ifs does not have that ... you can't say if this one's false you're just going to have multiple conditions". Hold on to that, because it explains the one surprise in this lesson.

Alex's example. He works on the job title column and makes up labels for fun ("we can just do random things"):

- if the job title equals Salesman, show "Sales"
- if it equals HR, show "Fire Immediately"
- a third condition shows "Give Christmas Bonus" (which job title it tests can't be made out in the recording)

On your practice sheet, with job titles in column F, the first two pairs look like this:

```
=IFS(F2="Salesman","Sales",F2="HR","Fire Immediately")
```

The surprise: #N/A. When Alex runs it, some rows show #N/A. He explains: "when that happens it just means nothing met that condition". His table has Supplier Relations and Accountant job titles, and "we never said anything about supplier relations we never said anything about accountants". Those rows had no matching pair, and with no false value to fall back on, IFS returned #N/A.

So #N/A from IFS isn't a broken formula. It's a list of the rows you forgot to describe.

[General knowledge, not from the video. Check before publishing.] Three details that make IFS reliable:

- **Catch everything else.** Add a last pair whose test is simply `TRUE` and whose answer is your fallback label. TRUE is always true, so any row that reaches it gets that label instead of #N/A:
  `=IFS(F2="Salesman","Sales",F2="HR","People",TRUE,"Other")`
- **Order matters.** IFS checks the pairs from left to right and returns the answer for the first test that's true. It ignores the rest. With numbers, put the narrowest test first: `=IFS(D2>50,"50 plus",D2>30,"31 to 50",TRUE,"30 and under")`. Reverse the first two and nobody would ever get "50 plus", because anyone over 50 is also over 30.
- **Text in quotation marks.** Both the job title you test for and the label you return go in quotation marks. The text match ignores capital letters, so "hr" matches "HR".

IFS needs Excel 2019 or later. In older versions, you'd nest one IF inside another, which this course doesn't cover.

[Instructor's method, not from the video.] Before you write the formula, list every distinct value in the column. Click the header, turn on a filter (Data > Filter), and open the drop-down: it shows each job title once. Write one pair for each value you want to label, then decide what the TRUE fallback should say.

4. Worked example

Drawn from the source. Alex writes an IFS on his job title column with three pairs: Salesman returns "Sales", HR returns "Fire Immediately", and a third title returns "Give Christmas Bonus". He presses Enter. The Salesman and HR rows show their labels. The Supplier Relations and Accountant rows show #N/A, because no pair mentioned them.

The one thing that would have gone wrong: if those #N/A cells had been fed into a count or a report, they would have broken it. Adding `TRUE,"Other"` as a final pair (general knowledge) gives every row a label.

5. Common mistake

Treating #N/A from IFS as a formula error and rewriting the conditions. The conditions were fine; there just weren't enough of them. Do instead: list every value in the column first, write a pair for each one you care about, and end with a `TRUE` pair for everything else.

6. Summary

IFS takes pairs: a test, then the answer if that test is true, repeated as many times as you need. It has no false value, so any row that matches no test shows #N/A, as Alex's Supplier Relations and Accountant rows did. End with `TRUE,"Other"` to catch the rest. IFS returns the first true test, so put narrow tests before broad ones. Text goes in quotation marks.

7. Quiz

1. You write `=IFS(F2="Salesman","Sales",F2="HR","People")`. A row with the job title "Accountant" shows #N/A. Why? A. The formula has a typing error B. No pair tests for Accountant, and IFS has no false value C. IFS only works on numbers D. Accountant is spelled with a capital letter

2. You want every job title other than Salesman and HR to show "Other". Which formula does that? A. `=IFS(F2="Salesman","Sales",F2="HR","People","Other")` B. `=IFS(F2="Salesman","Sales",F2="HR","People",TRUE,"Other")` C. `=IF(F2="Salesman","Sales","HR","People")` D. `=IFS(TRUE,"Other",F2="Salesman","Sales",F2="HR","People")`

3. A person is 55. What does `=IFS(D2>30,"Over 30",D2>50,"Over 50",TRUE,"30 or under")` show? A. Over 50 B. Over 30 C. 30 or under D. #N/A

Answer key

1. **B.** IFS returns #N/A when no test is true. A is wrong: nothing is mistyped; a pair is missing. C is wrong: IFS tests text too, as the job titles show. D is wrong: the match ignores capitals.
2. **B.** A final `TRUE` pair catches everything else. A is wrong: a lone answer with no test isn't a valid pair. C is wrong: IF takes one test and two answers, not this pattern. D is wrong: TRUE first would label every row "Other".
3. **B.** IFS stops at the first true test, and 55 is over 30. A is wrong: that test is never reached. C is wrong: an earlier test was true. D is wrong: a test matched.

8. Assignment: a Department column

Time: 15 minutes.

Use the filter drop-down to list every job title on your practice sheet. In K1 type "Department". Write an IFS formula that gives each job title a department name, and end with a `TRUE` pair for anything new that might be added later. Then type a job title you didn't plan for into one row, check that it shows your fallback label, and undo it.

Done checklist

- [ ] A written list of every distinct job title in the column
- [ ] One pair for each job title, with text in quotation marks
- [ ] A final `TRUE` pair, and no #N/A anywhere in rows 2 to 10
- [ ] An unplanned job title was tested and showed the fallback label

Sample answer (illustrative)

> Titles: Manager, Salesman, HR, Accountant, Receptionist. K2: `=IFS(F2="Salesman","Sales",F2="HR","People",F2="Accountant","Finance",F2="Manager","Management",TRUE,"Other")`, copied to K10. Receptionist shows "Other". Test with "Driver" in F6: "Other". Undone.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Total and Count with One Condition: SUMIF and COUNTIF

Module 2: Make formulas check conditions.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Answer a "how much" question and a "how many" question, each with one condition, using SUMIF and COUNTIF.

2. Why it matters

"What do we pay everyone above the threshold?" and "how many people earn over it?" come up far more often than a plain total. SUMIF and COUNTIF answer them straight from the list, with no sorting, filtering or copying rows out.

3. The teaching

You've used SUM and COUNT, and you've written a test with IF. SUMIF and COUNTIF put the two together. Alex describes SUMIF this way: "you can add an if statement and then add it if it meets a certain criteria".

SUMIF's inputs. Open the parenthesis after `=SUMIF` and you need, in Alex's words, a range and criteria, and "you can include a sum range if you would like":

1. **range**: the cells to test.
2. **criteria**: the condition each cell must meet.
3. **sum_range** (optional, shown in square brackets): the cells to add, if they're different from the ones you're testing.

Alex's SUMIF. He uses the salary column as the range, and his condition is a salary greater than 50,000. On your practice sheet:

```
=SUMIF(G2:G10,">50000")
```

Alex's result is 128,000, and he checks it out loud: "now his is fifty thousand exactly so that won't count but we have sixty three and sixty five thousand which does equal 128 thousand". Two salaries were over 50,000, and one person on exactly 50,000 was left out. That's the same boundary you met with IF: "greater than" doesn't include the number itself.

COUNTIF. COUNTIF takes a range and criteria, "exact same as we did before", Alex says. The difference is what it does with the matches. SUMIF adds the values. COUNTIF counts how many cells meet the condition. Alex asks how many people earn over 45,000:

```
=COUNTIF(G2:G10,">45000")
```

His answer is 5. As he puts it, the SUMIF "adds everything together", while "the count is just going to count the amount of cells that meet that criteria".

[General knowledge, not from the video. Check before publishing.] The details that make these work:

- **Quotation marks around the condition.** When the condition includes a comparison sign, the whole thing goes in quotation marks: `">50000"`, `">=50000"`, `"<>HR"`. A plain number can go without them.
- **Text conditions.** To match text, put the text in quotation marks: `=COUNTIF(E2:E10,"Female")`.
- **Testing one column, adding another.** This is what sum_range is for. To total the salaries of everyone whose gender is Female, test column E and add column G: `=SUMIF(E2:E10,"Female",G2:G10)`. If you leave sum_range out, SUMIF adds the tested cells themselves, which is what Alex did, because he tested and added the same salary column.
- **Including the boundary.** "50,000 or more" is `">=50000"`.

Choosing between them. Ask what the question wants back. "How much?" or "what's the total?" is SUMIF. "How many?" is COUNTIF. The condition is written the same way in both.

[Instructor's method, not from the video.] Check a SUMIF by running the matching COUNTIF next to it. Alex's numbers show why that helps: his SUMIF over 50,000 added two salaries (63,000 and 65,000), so a COUNTIF with the same condition should return 2. If the count surprises you, the total is wrong too.

4. Worked example

Drawn from the source. Alex wants the combined salary of everyone earning more than 50,000. He writes SUMIF over the salary column with the condition greater than 50,000 and gets 128,000. He checks it: 63,000 plus 65,000 is 128,000, and the person on exactly 50,000 isn't included. He then asks a "how many" question with COUNTIF, over 45,000, and gets 5.

The one thing that would have gone wrong: if his manager had asked for "everyone on 50,000 or more", 128,000 would have been 50,000 short. The condition needed to be `">=50000"`.

5. Common mistake

Writing the condition without quotation marks, as in `=SUMIF(G2:G10,>50000)`. Excel rejects the formula. Do instead: put any condition with a sign in quotation marks, and decide whether the boundary is in or out before you type the sign.

6. Summary

SUMIF adds only the values that meet one condition; COUNTIF counts how many cells meet it. Both take a range and a condition, and SUMIF can take a third range to add when you test one column and total another. Conditions with a sign go in quotation marks: `">50000"`. Alex's SUMIF over 50,000 returned 128,000 (63,000 plus 65,000; exactly 50,000 left out), and his COUNTIF over 45,000 returned 5.

7. Quiz

1. You need the total salary of everyone earning 50,000 or more. Salaries are in G2:G10. Which formula? A. `=SUMIF(G2:G10,">50000")` B. `=SUMIF(G2:G10,">=50000")` C. `=COUNTIF(G2:G10,">=50000")` D. `=SUMIF(G2:G10,>=50000)`

2. Your manager asks how many staff are in HR. Job titles are in F2:F10. Which formula answers that? A. `=SUMIF(F2:F10,"HR")` B. `=COUNTIF(F2:F10,"HR")` C. `=COUNT(F2:F10)` D. `=COUNTIF(F2:F10,HR,G2:G10)`

3. You need the total salary paid to Female staff. Gender is in E2:E10 and salary in G2:G10. Which formula? A. `=SUMIF(G2:G10,"Female")` B. `=SUMIF(E2:E10,"Female",G2:G10)` C. `=SUMIF(G2:G10,"Female",E2:E10)` D. `=COUNTIF(E2:E10,"Female")`

Answer key

1. **B.** "Or more" includes 50,000, and the condition is in quotation marks. A is wrong: it leaves out exactly 50,000. C is wrong: it counts people rather than adding salaries. D is wrong: the condition has no quotation marks, so Excel rejects it.
2. **B.** "How many" with one condition is COUNTIF. A is wrong: job titles are text, so there's nothing to add. C is wrong: COUNT ignores text and has no condition. D is wrong: the text needs quotation marks, and COUNTIF takes no third range.
3. **B.** Test the gender column, add the salary column. A is wrong: it tests salaries for the word Female. C is wrong: the ranges are swapped. D is wrong: it counts people instead of adding pay.

8. Assignment: four one-condition answers

Time: 15 minutes.

Add four rows to your summary block and answer each with one formula: (1) total salary of people earning over 50,000, (2) how many people earn over 50,000, (3) the same two again with "50,000 or more", (4) total salary of one gender of your choice. Your practice sheet has one salary of exactly 50,000, so (1) and (3) should differ by exactly 50,000.

Done checklist

- [ ] Every condition with a sign is in quotation marks
- [ ] The "over" and "or more" totals differ by exactly 50,000
- [ ] Each SUMIF has a COUNTIF beside it with the same condition
- [ ] The gender total tests the gender column and adds the salary column

Sample answer (illustrative)

> Over 50,000: `=SUMIF(G2:G10,">50000")` 228000 | `=COUNTIF(G2:G10,">50000")` 4 50,000 or more: `=SUMIF(G2:G10,">=50000")` 278000 | `=COUNTIF(G2:G10,">=50000")` 5 Female total: `=SUMIF(E2:E10,"Female",G2:G10)` 223000

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Total and Count with Several Conditions: SUMIFS and COUNTIFS

Module 2: Make formulas check conditions.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Answer "how much" and "how many" questions with two or more conditions using SUMIFS and COUNTIFS, with the inputs in the right order.

2. Why it matters

Real questions usually come with more than one "where": female staff over 30, sales in March in the North region. SUMIFS and COUNTIFS take as many conditions as you need, and they're among the most useful formulas in day-to-day reporting.

3. The teaching

SUMIFS is to SUMIF what IFS was to IF. Alex makes the link himself: "there's the if and the ifs so the ifs is going to be if it has it meets multiple conditions". The tooltip describes SUMIFS as adding "the cells specified by a given set of conditions or criteria".

The order changes. Alex warns you before he starts: "the syntax or this one's going to be a little bit different". In SUMIF, the range you add came last and was optional. In SUMIFS it comes first and is required:

1. **sum_range**: the cells to add. Alex: "this is the area that's going to be added after all these if statements are done".
2. **criteria_range1**, **criteria1**: the first column to test and its condition.
3. **criteria_range2**, **criteria2**: the second column and its condition, and so on.

After the sum range, everything comes in pairs: a column, then the condition for that column.

Alex's SUMIFS. He adds salaries for people who are female and over 30. On your practice sheet, salaries in G, gender in E and age in D:

```
=SUMIFS(G2:G10,E2:E10,"Female",D2:D10,">30")
```

His result is 88,000. He checks it by finding the matching rows: "female female there's one two right here", two people who meet both conditions, and their salaries add to 88,000.

Every condition must be true. A row is added only if it passes every pair. A woman aged 28 fails the age test, and a man aged 45 fails the gender test, so neither is added.

COUNTIFS uses the same pairs, with no sum range in front, because it counts rows rather than adding them. Alex counts people whose ID is greater than 1005 and who are male. "It can be any range," he notes, and this time he tests the

ID column. On your practice sheet, IDs in A and gender in E:

```
=COUNTIFS(A2:A10,">1005",E2:E10,"Male")
```

His answer is 3, and he names them: "michael stanley and kevin". Alex adds that this is one he genuinely uses a lot, "very useful to give quick numbers like this".

[General knowledge, not from the video. Check before publishing.]

- The quotation-mark rules from SUMIF still apply: `">30"`, `"Female"`.
- Every range must be the same size. If the sum range is G2:G10, each criteria range must also run from row 2 to row 10. Ranges of different sizes return #VALUE!.
- SUMIFS and COUNTIFS only do "and". For "or" (female or over 30), write two formulas and think carefully about rows counted twice.
- With one condition, SUMIFS works too: `=SUMIFS(G2:G10,G2:G10,">50000")` gives the same answer as SUMIF. Some people use SUMIFS for everything so they only learn one order.

[Instructor's method, not from the video.] Build a multi-condition formula one condition at a time. Write the COUNTIFS with the first pair, check the count makes sense, add the second pair, check again. When the count is right, copy the pairs into a SUMIFS and put the sum range at the front. It's much easier to find a mistake when only one thing changed.

4. Worked example

Drawn from the source. Alex wants the combined salary of female employees over 30. He types `=SUMIFS(`, selects the salary column as the sum range, then the gender column with "female", then a second range with "greater than 30". The result is 88,000. He finds the two rows that meet both conditions and confirms they make up the total. Then he switches to COUNTIFS: ID greater than 1005 and male. Three people match: Michael, Stanley and Kevin.

The one thing that would have gone wrong: if he'd carried over the SUMIF habit and put the salary range at the end, Excel would have read the salary column as a condition range and returned a wrong figure or an error.

5. Common mistake

Putting the sum range last in SUMIFS, the way SUMIF does it. Do instead: say the order to yourself as you type, "what to add, then where to look and what to look for", and read the tooltip, which shows sum_range first.

6. Summary

SUMIFS adds values that meet every condition you give it; COUNTIFS counts the rows that do. SUMIFS puts the range to add first, then pairs of column and condition. COUNTIFS has only the pairs. All ranges must be the same size, and a row must pass every condition. Alex's SUMIFS for female staff over 30 returned 88,000; his COUNTIFS for ID over 1005 and male returned 3: Michael, Stanley and Kevin.

7. Quiz

1. Salaries are in G2:G10, gender in E2:E10 and age in D2:D10. Which formula totals the salaries of male staff aged 40 or over? A. `=SUMIFS(E2:E10,"Male",D2:D10,">=40",G2:G10)` B.

`=SUMIFS(G2:G10,E2:E10,"Male",D2:D10,">=40")` C. `=SUMIF(G2:G10,"Male",">=40")` D. `=SUMIFS(G2:G10,E2:E10,"Male",D2:D10,">40")`

2. `=COUNTIFS(A2:A10,">1005",E2:E10,"Male")`**. A woman has ID 1008. Is she counted?** A. Yes, because her ID is over 1005 B. No, because she fails one of the two conditions C. Yes, because COUNTIFS counts a row if any condition is true D. Only if she is over 30

3. You write `=SUMIFS(G2:G10,E2:E9,"Female")` **and get #VALUE!. What's the fix?** A. Put the condition before the range B. Make the criteria range run from row 2 to row 10 like the sum range C. Switch to COUNTIFS D. Remove the quotation marks from "Female"

Answer key

1. **B.** Sum range first, then column and condition pairs, with `>=` for "40 or over". A is wrong: the sum range is at the end, SUMIF style. C is wrong: SUMIF takes one condition and this mixes the inputs. D is wrong: `>40` leaves out people aged exactly 40.
2. **B.** Every pair must be true, and she fails the gender test. A is wrong: one condition isn't enough. C is wrong: COUNTIFS needs all conditions, not any. D is wrong: age isn't in this formula.
3. **B.** The ranges must be the same size. A is wrong: pairs always go range, then condition. C is wrong: COUNTIFS has the same rule. D is wrong: text conditions need quotation marks.

8. Assignment: three multi-condition answers

Time: 20 minutes.

Write three questions about your practice sheet that each have two conditions, then answer each with a COUNTIFS and a SUMIFS. Build each COUNTIFS one condition at a time and record the count after each step. Check each final count by finding the matching rows by eye.

Done checklist

- [ ] Three questions written out in words, each with two conditions
- [ ] Each SUMIFS starts with the salary range
- [ ] All ranges in each formula run from row 2 to row 10
- [ ] The count after the first condition and after both conditions is recorded for each question
- [ ] Each final count matches the rows found by eye

Sample answer (illustrative)

> Q1: Female staff aged 30 or over. Step 1 `=COUNTIFS(E2:E10,"Female")` 4. Step 2 `=COUNTIFS(E2:E10,"Female",D2:D10,">=30")` 2. `=SUMIFS(G2:G10,E2:E10,"Female",D2:D10,">=30")` 121000. Rows found by eye: Rosa Delgado, Ana Silva. Match.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Check and Clean a Text Column with LEN and TRIM

Module 3: Clean and build text. By the end of this module you can find and fix messy text, pull pieces out of a value, and join columns into new ones.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Flag entries of the wrong length with LEN, and strip stray spaces from a text column with TRIM.

2. Why it matters

Data typed by people arrives with wrong-length IDs and invisible spaces, and both quietly break counts, matches and joins. Checking and cleaning a column first takes two formulas and saves you from reporting numbers built on bad entries.

3. The teaching

LEN counts characters. The tooltip says LEN "returns the number of characters in a text string". Alex runs it on the last name column: "for halpert it's seven characters for flenderson it's ten characters". On your practice sheet:

```
=LEN(C2)
```

That on its own isn't very interesting. What makes LEN useful is knowing how long a value *should* be. Alex's example comes from his own work. "I used to work with a lot of customer data or patient data they had like social security numbers." When a number had the wrong length, "let's say 10 numbers or 11 numbers where it should only be nine", he knew it was wrong and could "get rid of that or discard it from our results". He gives a second case: phone numbers. "If you look at the length of the phone numbers and there's ones that are like 12 numbers long you know those might not be ones that are accurate and you need to go look at them."

So LEN is a check: decide the correct length, then find every entry that doesn't have it. Combine it with IF from Module 1 and the flag writes itself. If social security numbers sat in column L, stored as nine digits:

```
=IF(LEN(L2)<>9,"Check","OK")
```

[General knowledge, not from the video. Check before publishing.] LEN counts every character, including spaces, dashes and brackets. Alex says he thinks a social security number is nine digits, and that is correct for the digits, but written with dashes (123-45-6789) it is 11 characters. Decide whether your column stores the dashes before choosing the correct length. The same goes for phone numbers with spaces or brackets.

TRIM removes stray spaces. Alex deliberately messed up a last name column for this one, adding random spaces at the end of names. His reason is worth hearing: "when you're working with real data you're going to get data like this it's messy it's dirty". It happens because "sometimes you're going to be working with data that is inputted by a user it's not like a drop down option". Someone types a name, hits the space bar or Enter by accident, submits it, and that's how it sits in the database. "I guarantee you if you're working as a data analyst you're going to see stuff like this," he says, and not just in names.

The tooltip explains what TRIM does: it "removes all spaces from [a] text string except for a single space between words". So "Jim Halpert" keeps the space in the middle, because, as Alex says, TRIM "understands that in normal language space is supposed to be there". The extra spaces at either end go.

```
=TRIM(C2)
```

Alex hands it the whole column and gets clean names, "nice and clean much more usable".

[Instructor's method, not from the video.] You usually can't see a trailing space, so use LEN to find them. Compare the length before and after trimming:

```
=LEN(C2)-LEN(TRIM(C2))
```

Any result above 0 is the number of extra spaces in that cell. Run it down the column before you clean, so you know how many entries were affected.

[General knowledge, not from the video. Check before publishing.] TRIM gives you a new, clean column; the messy one is still there. To replace it, copy the trimmed column and use Paste Special > Values over the original, or keep both and use the clean one from now on. Alex's whole-column formula spills in Microsoft 365; in older versions, write it for row 2 and copy it down.

4. Worked example

Drawn from the source, with an illustrative check added. Alex has a last name column with random spaces typed after some names. He writes `=TRIM(` over the column and closes it. The new column shows every name with the extra spaces gone and the names otherwise unchanged.

Illustrative. On a practice sheet, "Park " (with a trailing space) and "Park" look identical. `=LEN(C4)` returns 5 instead of 4, and `=LEN(C4)-LEN(TRIM(C4))` returns 1. After TRIM, a COUNTIF for "Park" finds her.

The one thing that would have gone wrong: without trimming, `=COUNTIF(C2:C10,"Park")` would have missed her, because "Park " with a space isn't the same text as "Park".

5. Common mistake

Picking the "correct" length without checking how the data is written. A social security number column stored with dashes would flag every single row as wrong at 9 characters. Do instead: look at a few real entries, count what's actually in them, spaces and dashes included, and set the length from that.

6. Summary

LEN returns how many characters a cell holds; Alex's "Halpert" is 7 and "Flenderson" is 10. Use it as a check: decide the right length, then flag the rest with a formula such as `=IF(LEN(L2)<>9,"Check","OK")`. LEN counts spaces and dashes too. TRIM removes every space except single spaces between words, which cleans names typed with stray spaces. `LEN(C2)-LEN(TRIM(C2))` shows how many extra spaces a cell had.

7. Quiz

1. Phone numbers in column M should be exactly 10 digits, stored without spaces. Which formula flags the wrong ones? A. `=LEN(M2)` B. `=IF(LEN(M2)<>10,"Check","OK")` C. `=TRIM(M2)` D. `=IF(LEN(M2)=10,"Check","OK")`

2. `=LEN(C5)` returns 8 for the name "Delgado", which has 7 letters. What's the likely cause, and the fix? A. A trailing space; clean it with TRIM B. LEN counts letters twice; use COUNT instead C. The name is misspelled; retype it D. The cell is formatted as a date; change it to General

3. TRIM is applied to " Jim Halpert " (with extra spaces at both ends and three spaces in the middle). What comes back? A. "JimHalpert" B. "Jim Halpert" C. "Jim Halpert" D. " Jim Halpert "

Answer key

1. **B.** It flags anything that isn't 10 long. A is wrong: it shows the length but flags nothing. C is wrong: TRIM removes spaces; it doesn't check length. D is wrong: it flags the correct entries instead of the wrong ones.
2. **A.** A hidden trailing space is the usual cause of one extra character. B is wrong: LEN counts each character once. C is wrong: the letters are fine. D is wrong: format doesn't change the text's length.
3. **B.** TRIM removes spaces at the ends and reduces gaps between words to one. A is wrong: TRIM keeps the single space between words. C is wrong: the extra middle spaces are removed. D is wrong: the end spaces are removed.

8. Assignment: a cleaning log for the last name column

Time: 15 minutes.

On your practice sheet, add three columns: LEN of the last name, extra spaces (`LEN(C2)-LEN(TRIM(C2))`), and the trimmed last name. Note how many rows had extra spaces. Then paste the trimmed names over the originals as values, and confirm that the extra-spaces column now shows 0 everywhere.

Done checklist

- [ ] The extra-spaces column shows a number for every row, 2 to 10
- [ ] The row with the trailing space you planted shows 1 or more
- [ ] A note records how many rows had extra spaces before cleaning
- [ ] After pasting values, every row in the extra-spaces column shows 0
- [ ] The last name column holds plain text, not formulas

Sample answer (illustrative)

> Before: 1 row with extra spaces (row 4, "Park ", LEN 5, extra 1). Trimmed with `=TRIM(C2)` copied to row 10, pasted as values over C2:C10. After: extra spaces 0 on all nine rows.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Pull Characters from Either End with LEFT and RIGHT

Module 3: Clean and build text.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Extract a set number of characters from the start or the end of a value with LEFT and RIGHT.

2. Why it matters

Codes, IDs and dates often pack several pieces of information into one cell: a prefix, a branch number, a year. LEFT and RIGHT pull out the piece you need for every row at once, so you can count, sort or label by it.

3. The teaching

If you've used SQL, Alex has a shortcut for you: "left and right are kind of like sub strings". In SQL a substring lets you "choose a certain part of the text string and you can extract data from that", usually by giving a number of characters. Excel, he points out, has no function called SUBSTRING: "there's substitute but there's no substring left and right is really the closest thing that we have". If you've never touched SQL, ignore the comparison. The idea is simple: take some characters from one end.

LEFT. The tooltip says LEFT returns the specified number of characters "from the start of a text string". Alex's description: "we're starting from the very far left". It takes two inputs:

1. **text**: the cell to take characters from.
2. **num_chars**: how many characters to take.

Alex starts simple. He picks a text column and asks for three characters, "just choose three as an example", and "you can see that it takes the first three characters from every single thing". On your practice sheet, to take the first three letters of each first name:

```
=LEFT(B2,3)
```

"Rosa" gives "Ros". "Tom" gives "Tom", because it only has three letters.

Numbers work too. "It doesn't just have to be you know name with with actual words or letters," Alex says. "You can do the exact same thing" with numbers.

RIGHT. RIGHT is the mirror image: "this one is going to start from the very far right and go over one character". Alex runs it on his ID column, where "all of them start with 100", and asks for just the last character:

```
=RIGHT(A2,1)
```

On an ID of 1007 (an illustrative value), that gives 7. "Starting from the very far right we go over one character and that's what we take."

The same pattern answers the opposite question. To take the "100" prefix off the front of each ID, use LEFT with 3: `=LEFT(A2,3)`.

[General knowledge, not from the video. Check before publishing.] Three details worth knowing:

- **The result is text.** Even when you take characters from a number, LEFT and RIGHT hand back text. "7" from RIGHT is not the number 7, so it won't add up with SUM or match a number test. Wrap it in VALUE to turn it back into a number: `=VALUE(RIGHT(A2,1))`.
- **num_chars is optional.** Leave it out and you get one character.
- **The middle.** Alex says LEFT and RIGHT are the closest thing to a substring. Excel does also have MID, which takes characters from the middle: `=MID(text, start position, number of characters)`. It isn't covered in this course.

As with TRIM, Alex gives LEFT and RIGHT a whole column at once. In versions without spilling, write the formula for row 2 and copy it down.

Alex has one more use of RIGHT, the one he says he has used "countless times" on the job: pulling the year off the end of a date. It needs one extra step first, because of how Excel stores dates, so it waits until Module 4.

4. Worked example

Drawn from the source, with illustrative IDs. Alex's IDs all start with 100. He writes RIGHT over the ID column with 1 character and gets the last digit of every ID. On a practice sheet with IDs 1001 to 1009, `=RIGHT(A2,1)` returns 1 for ID 1001 and 7 for ID 1007.

Illustrative. Next, with the last digits in column L, you label IDs ending above 5 as a "late batch": `=IF(L2>5,"Late batch","Early batch")`. Every row comes back "Late batch", even ID 1001. The digits are text, and Excel ranks any text above any number in a comparison, so "1" counts as bigger than 5.

The one thing that would have gone wrong: nothing looks broken, so the labels could have gone into a report unchecked. The fix is to turn the digit back into a number first, `=IF(VALUE(L2)>5,"Late batch","Early batch")` (general knowledge).

5. Common mistake

Running LEFT or RIGHT on a column you haven't cleaned. If a value has a trailing space, `=RIGHT(C2,3)` takes the space as one of its three characters, and you get two letters and a blank. Do instead: TRIM the column first, as in the last lesson, then extract.

6. Summary

LEFT takes a set number of characters from the start of a value, RIGHT from the end. Both take the cell and the number of characters: `=LEFT(B2,3)`, `=RIGHT(A2,1)`. Alex took the first three characters of a text column, and the last digit of IDs that all start with 100. They work on numbers too, but the result is text. Clean stray spaces with TRIM before extracting, or the space becomes one of your characters.

7. Quiz

1. Product codes in column A look like "NY-4471". You need the two-letter city code at the front. Which formula? A. `=RIGHT(A2,2)` B. `=LEFT(A2,2)` C. `=LEFT(A2,4)` D. `=LEN(A2,2)`

2. Codes in column A look like "NY-4471". You need the last four digits. Which formula? A. `=LEFT(A2,4)` B. `=RIGHT(A2,4)` C. `=RIGHT(A2,3)` D. `=TRIM(A2,4)`

3. `=RIGHT(C2,3)` on the last name "Reyes" returns "es" plus a blank. What should you do? A. Change it to `=RIGHT(C2,2)` B. TRIM the last names first, then run RIGHT on the clean text C. Use LEFT instead D. Delete the row

Answer key

1. **B.** The city code is the first two characters. A is wrong: it takes the last two digits. C is wrong: it takes "NY-4", including the dash and a digit. D is wrong: LEN counts characters and takes one input.
2. **B.** The digits are the last four characters. A is wrong: it takes "NY-4" from the front. C is wrong: it misses one digit. D is wrong: TRIM removes spaces; it doesn't extract.
3. **B.** The blank is a trailing space, so clean before extracting. A is wrong: it hides the problem in this row and breaks the clean rows. C is wrong: LEFT takes from the other end. D is wrong: the data is fine once trimmed.

8. Assignment: an initials and branch column

Time: 15 minutes.

On your practice sheet, add two columns. In the first, take the first letter of each first name and the first letter of each last name in two separate LEFT formulas. In the second, take the last digit of each ID with RIGHT. Then label each ID "Late batch" if its last digit is above 5 and "Early batch" otherwise, turning the digit back into a number first.

Done checklist

- [ ] The LEFT formulas use 1 as the number of characters
- [ ] The RIGHT formula returns a single digit for every ID from row 2 to row 10
- [ ] The last name column was trimmed before taking its first letter
- [ ] The batch label uses VALUE, and ID 1001 shows "Early batch"

Sample answer (illustrative)

> First initial `=LEFT(B2,1)` R | Last initial `=LEFT(C2,1)` D Last ID digit `=RIGHT(A2,1)` 1 (ID 1001) Batch `=IF(VALUE(N2)>5,"Late batch","Early batch")` Early batch (ID 1001); Late batch for 1006 to 1009.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Join Names and Build Emails with CONCATENATE

Module 3: Clean and build text.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Join columns into a full name, and into a patterned email address, with CONCATENATE.

2. Why it matters

Lists usually store pieces separately (first name, last name, domain, code), and mail merges, labels and logins need them joined. One CONCATENATE formula builds the joined version for every row instead of retyping it.

3. The teaching

Alex calls CONCATENATE "one that i have used just way way way too many times". The tooltip says it all: it "joins two or more text strings into one string". As Alex puts it, "it basically joins things together and adds them together".

First try: two columns. Alex joins first name and last name:

```
=CONCATENATE(B2,C2)
```

The result, in his words: "it says jim halpert but it's all connected and that's typically not how people write their names". CONCATENATE joins exactly what you give it, and you didn't give it a space.

Adding a space. So Alex goes back into the formula and adds another input between the two names. A pair of quotation marks holds whatever text you want to drop in, and "we can include anything in here we can include a dash we can include an exclamation point or we can just include a space". A space between quotation marks:

```
=CONCATENATE(B2," ",C2)
```

"And just like that it works perfectly." Now each row shows a proper full name.

That's the whole technique: the inputs are joined in order, left to right, and anything you want that isn't already in a cell goes in quotation marks as its own input.

Building an email. Alex's second use is "something like generating an email". He replaces the space with a dot, and adds the domain at the end. On your practice sheet:

```
=CONCATENATE(B2,".",C2,"@gmail.com")
```

"And now i've created emails for all of these people." Rosa Delgado becomes Rosa.Delgado@gmail.com. The exact pieces Alex types are on screen rather than spoken, so the order shown here (first name, dot, last name, domain) is the pattern his description points to.

Alex adds that this example shows up in demonstrations "almost everywhere because honestly it gets used a lot um by data analysts".

[Instructor's method, not from the video.] Join clean text. Any stray space left in a name ends up inside the joined result, and in an email address that space makes it unusable. If you skipped cleaning, trim inside the formula: `=CONCATENATE(TRIM(B2),".",TRIM(C2),"@gmail.com")`.

[Instructor's method, not from the video.] A formula like this builds addresses that follow a pattern. It doesn't make them real. Don't send anything to generated addresses until each one has been confirmed.

[General knowledge, not from the video. Check before publishing.] Newer Excel versions also have CONCAT, which works the same way, and the ampersand joins text without a function at all: `=B2&" "&C2` gives the same full name. CONCATENATE still works in current versions. As with the other text functions, Alex's whole-column version spills in Microsoft 365; elsewhere, write row 2 and copy it down.

4. Worked example

Drawn from the source. Alex joins the first and last name columns and gets "JimHalpert", run together. He edits the formula to put a space in quotation marks between the two cells, and every row now reads as a full name: "Jim Halpert". He then swaps the space for a dot and adds "@gmail.com" on the end, and every row has an email address built from the person's name.

Illustrative. On a practice sheet where "Park" still has a trailing space, the email formula returns "Nadia.Park @gmail.com", with a space before the @.

The one thing that would have gone wrong: that address would fail wherever it was used. Cleaning the last name column with TRIM first, or trimming inside the formula, prevents it.

5. Common mistake

Forgetting the separator, then trying to fix the result by hand. You end up with "JimHalpert" in nine rows and nine manual edits. Do instead: decide the separator first (a space for names, a dot for emails) and put it in quotation marks as its own input between the cells.

6. Summary

CONCATENATE joins its inputs, in order, into one piece of text. Cells go in as references; anything else goes in quotation marks as its own input. `=CONCATENATE(B2," ",C2)` makes a full name; leave out the `" "` and you get Alex's "JimHalpert". Swap the space for a dot and add `"@gmail.com"` to build a patterned email. Trim names first, because stray spaces end up in the result.

7. Quiz

1. First names are in B and last names in C. Which formula produces "Halpert, Jim"? A. `=CONCATENATE(B2,", ",C2)` B. `=CONCATENATE(C2,", ",B2)` C. `=CONCATENATE(C2,B2)` D. `=CONCATENATE("C2, B2")`

2. You need staff logins in the form first initial plus last name, such as "RDelgado". Which formula builds it? A. `=CONCATENATE(LEFT(B2,1),C2)` B. `=CONCATENATE(B2,C2)` C. `=LEFT(CONCATENATE(B2,C2),1)` D. `=CONCATENATE(RIGHT(B2,1),C2)`

3. Your email formula returns "Nadia.Park @company.com". What's the cause? A. CONCATENATE always adds a space before text in quotation marks B. The last name cell has a trailing space C. The dot needs to be outside the quotation marks D. The domain should be in its own cell

Answer key

1. **B.** Inputs join in order, so last name first, then comma and space, then first name. A is wrong: it gives "Jim, Halpert". C is wrong: it has no separator. D is wrong: the cell addresses are inside quotation marks, so they print as text.
2. **A.** LEFT takes the first letter, then CONCATENATE adds the last name. B is wrong: it uses the whole first name. C is wrong: it returns just one letter. D is wrong: RIGHT takes the last letter of the first name.
3. **B.** CONCATENATE copies the space it finds in the cell. A is wrong: it adds nothing you didn't give it. C is wrong: the dot is fine as quoted text. D is wrong: a quoted domain works.

8. Assignment: full name and email columns

Time: 15 minutes.

On your practice sheet, add a "Full Name" column and an "Email" column. Use a company domain of your choice for the email. Check every result for spaces in the wrong place, and fix any you find by trimming rather than retyping.

Done checklist

- [ ] Full Name shows first name, one space, last name, on every row from 2 to 10
- [ ] Email follows one pattern on every row, with the domain in quotation marks
- [ ] No email contains a space
- [ ] Any trailing space was fixed with TRIM, not by editing the result

Sample answer (illustrative)

> Full Name: `=CONCATENATE(B2," ",C2)` Rosa Delgado Email: `=CONCATENATE(TRIM(B2),".",TRIM(C2),"@example.com")` Rosa.Delgado@example.com Row 4 checked: Nadia.Park@example.com, no space.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Turn Dates into Text with TEXT and Pull Out the Year

Module 4: Fix and use dates. By the end of this module you can tell real dates from text dates, convert and reshape them, and count the days and working days between two dates.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Tell a real date from a text date, convert a date column to text with TEXT, keep the result as fixed values, and take the year from it with RIGHT.

2. Why it matters

Dates are where text formulas most often give answers that look like nonsense, and the cause is invisible until someone shows you. Once you can see what a date really is, you can reshape a whole column in a minute instead of retyping it.

3. The teaching

In the last module you used RIGHT to take characters from the end of a value. Alex's favorite use of it is pulling the year off a date, one he has used "countless times in actually using this in my job". But he hits a wall first, and the wall is the lesson.

The problem. Alex runs RIGHT with 4 characters on a column of end dates. "It's giving us completely random numbers. Why is that? Because underneath the date range there are um numbers." To prove it, he changes the cells' format to General: "it's going to have the numbers", and the last four characters of those numbers are exactly what RIGHT returned. "So it's doing what it's supposed to do but it's not doing what we actually want."

[General knowledge, not from the video. Check before publishing.] Excel stores a real date as a serial number: the count of days since January 1, 1900. What you see is formatting laid over that number. RIGHT reads the number, not the formatting. That's also why MAX and MIN worked on dates in Module 1, and why COUNT counts them.

Real date or text date? Alex shows that typing 11-2-2001 into a cell "converts it to a date format", while the same characters can also be entered so that they "stay a text string". You can tell them apart by where they sit in the cell: "this one is ... on the right and this one's on the left that's how you can tell the difference".

[General knowledge, not from the video. Check before publishing.] By default, Excel lines real dates and numbers up on the right of a cell, and text on the left. To type a date that stays as text, start it with an apostrophe: '11-2-2001. The apostrophe doesn't show in the cell.

The fix: TEXT. Rather than retyping dates "by hand completely manually and waste hours of your time", Alex uses TEXT, which "converts a value to text in a specific number format". It takes two inputs: the value, and a format code in quotation marks that tells Excel how the text should look. Alex's dates are day, then month, then four-digit year, so he types "dd slash mm yyyy":

```
=TEXT(I2,"dd/mm/yyyy")
```

He warns about the format code: "if we tell it something incorrect it's going to give us a completely terrible output or just give us an error altogether".

Keep the values. The converted column is still a formula pointing at the dates. Alex copies it "and paste[s] it right over here", and the cells now show as General, plain text.

[General knowledge, not from the video. Check before publishing.] Use Paste Special > Values for this step (on the Home tab, under Paste). An ordinary paste would copy the formulas, not the text.

Now RIGHT works. Alex runs RIGHT with 4 on the pasted text, "and there you go so now it works". Every row shows a year. His point about scale: on "10 000 rows it's gonna be a breeze ... it's going to take you two minutes or a minute".

[Instructor's method, not from the video.] Choose a format code that puts the year last. RIGHT takes from the end, so "dd/mm/yyyy" or "mm/dd/yyyy" both leave the four-digit year as the last four characters. A code like "yyyy-mm-dd" would make you use LEFT instead.

[General knowledge, not from the video. Check before publishing.] If all you need is the year of a real date, `=YEAR(I2)` returns it as a number in one step. Format code letters can vary with Excel's language settings.

4. Worked example

Drawn from the source. Alex wants the year each person left. He runs `RIGHT` with 4 on the end date column and gets four-digit numbers that aren't years. He switches the cells to General and sees the serial numbers underneath, which explains it. He converts the column with TEXT and the code "dd/mm/yyyy", copies the results and pastes them over as fixed values, and runs RIGHT with 4 again. Every row now shows the correct year.

The one thing that would have gone wrong: if he had typed a format code that didn't match what he wanted, or skipped the paste and later deleted the original date column, he would have had, in his words, "a completely terrible output", or a column of errors.

5. Common mistake

Running text formulas such as RIGHT or LEFT straight on real dates and trusting the result because it is four digits long. Do instead: check alignment or set the cells to General first. If you see five-digit numbers, they are real dates; convert with TEXT before extracting, or use YEAR.

6. Summary

A real date is a number underneath, so RIGHT on it returns what Alex calls "completely random numbers". Real dates sit on the right of a cell and text on the left. Convert a date column with `=TEXT(I2,"dd/mm/yyyy")`, paste the results into a new column as values, then RIGHT with 4 characters on that text returns the year. Use a format code that ends with the year; a wrong code gives bad output or an error.

7. Quiz

1. `=RIGHT(H2,4)` on a cell showing 3/14/2012 returns 0982. What's going on? A. The date was typed wrongly B. H2 holds a real date, and RIGHT is reading the serial number underneath C. RIGHT can only take 2 characters from a date D. The cell is text and needs to be converted to a date

2. You need the year as the last four characters of a text date. Which formula sets that up? A. `=TEXT(H2,"yyyy-mm-dd")` B. `=TEXT(H2,"dd/mm/yyyy")` C. `=TEXT(H2)` D. `=RIGHT(H2,"yyyy")`

3. You've converted dates with TEXT in column J and want to delete the original date column. What must you do first? A. Nothing, the text is already independent B. Copy column J and paste it back as values C. Change column J to a date format D. Run TRIM on column J

Answer key

1. **B.** The digits come from the serial number of a real date. A is wrong: the date displays correctly. C is wrong: RIGHT takes any number of characters. D is wrong: it's the other way round; the cell is a real date.
2. **B.** That code ends with the four-digit year. A is wrong: the year comes first, so you would need LEFT. C is wrong: TEXT needs a format code. D is wrong: RIGHT takes a number of characters, not a format.
3. **B.** Until pasted as values, column J is formulas pointing at the dates. A is wrong: deleting the source would break the formulas. C is wrong: that changes the display, not the dependency. D is wrong: there are no stray spaces to remove.

8. Assignment: a Start Year column

Time: 20 minutes.

On your practice sheet, run `=RIGHT(H2,4)` on the start dates and write down what you get. Set one date cell to General and note the number. Then convert the start dates with TEXT, paste the results as values into a new column, and take the year with RIGHT. Finally, check one row against `=YEAR(H2)`.

Done checklist

- [ ] A note records the "random" RIGHT result and the General number for the same cell
- [ ] The converted column sits on the left of its cells (text), not the right
- [ ] The converted column holds values, not formulas (click a cell and check the formula bar)
- [ ] The year column shows a four-digit year on every row from 2 to 10
- [ ] At least one row matches `=YEAR(H2)`

Sample answer (illustrative)

> H2 shows 3/14/2009. `=RIGHT(H2,4)` gave 9886; as General, H2 is 39886. `=TEXT(H2,"dd/mm/yyyy")` gave 14/03/2009, pasted as values into column O. `=RIGHT(O2,4)` gives 2009. `=YEAR(H2)` gives 2009. Match.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Swap Date Separators with SUBSTITUTE

Module 4: Fix and use dates.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Replace every separator, or only the one you choose, in a column of text dates with SUBSTITUTE.

2. Why it matters

Text dates often arrive in mixed styles, some with slashes and some with dashes, and a column in two styles can't be sorted, matched or converted cleanly. SUBSTITUTE brings the whole column to one style with a single formula.

3. The teaching

Alex sets the scene: "changing a date format changing how what it's supposed to look like is absolutely something that happens all the time". In his sheet, most dates use slashes and some use dashes. That's the mess you fix in this lesson, and it only makes sense now that you know a date can be stored as text, from the last lesson.

What SUBSTITUTE does. The tooltip: "substitute replaces existing text with new text in a text string". Open the parenthesis and Alex reads the inputs: "we take the text we have the old text we have the new text and then we have what instance".

1. **text**: the cell to work on.
2. **old_text**: the characters to replace.
3. **new_text**: what to put in their place.
4. **instance_num**: which occurrence to replace. It's shown in square brackets, and Alex explains what that means: "notice that that's in a bracket that means it's optional".

Replace every occurrence. Alex's first formula takes the dash-style dates and replaces the dash with a forward slash, with no instance number. On your practice sheet, with text dates in column P:

```
=SUBSTITUTE(P2,"-","/")
```

11-2-2001 becomes 11/2/2001. Alex is pleased: "this one is now in the correct format that we want". Then he goes the other way and turns every slash into a dash, so "now all of them are in this new format". The rule: "if you don't include an instance then it'll do it to every single one in there".

(Alex wonders aloud whether "/" is really called a forward slash. It is.)

Replace just one occurrence. Now the fourth input. Alex replaces slashes with dashes again, but "we only want to do it on the first instance of that forward slash":

```
=SUBSTITUTE(P2,"/","-",1)
```

11/2/2001 becomes 11-2/2001. "All the ones that were replaced are the very first instance whereas the second instance which is the second time it appears in the string does not get touched." Change the 1 to a 2 and "it's kind of the opposite": 11/2-2001. In his words, "we're choosing which instance or which time it shows up in that string and then it replaces it".

When to reach for it. Alex's use case: "if you want to do like a bulk replace but you only want to do it on a specific column and you just want to use a formula really quick".

[General knowledge, not from the video. Check before publishing.]

- SUBSTITUTE works on text. Run on a real date, it reads the serial number underneath, which has no slashes or dashes, and hands back that number as text: a five-digit figure such as 39886 instead of a date. Check that the column is text (sitting on the left of its cells) first.
- The old and new text go in quotation marks.
- SUBSTITUTE matches capitals exactly: replacing "hr" won't touch "HR".
- The result is a new column of formulas. Paste it as values if you want to replace the original, as in the last lesson.
- Excel's Find and Replace (Ctrl+H) can do a one-off swap too. A formula leaves the original untouched and only acts on the column you point it at.

[Instructor's method, not from the video.] Before you substitute, decide on one target style for the whole column and write it down, such as "month/day/year with slashes". Then use SUBSTITUTE to bring every row to that style, not just to "fix" whatever looks odd.

4. Worked example

Drawn from the source, with illustrative dates. Alex has a column of text dates where some use slashes and some use dashes. He writes SUBSTITUTE over the dash-style dates, with "-" as the old text and "/" as the new, and no instance number. Every dash becomes a slash and the column is consistent. Then he shows the instance number: with 1, only the first slash in each date changes; with 2, only the second.

Illustrative. With the date 11-2-2001: no instance gives 11/2/2001; turning slashes back with instance 1 gives 11-2/2001; with instance 2, 11/2-2001.

The one thing that would have gone wrong: had the column held real dates rather than text, SUBSTITUTE would have returned five-digit serial numbers instead of dates, with no error to explain why (general knowledge).

5. Common mistake

Adding an instance number when you meant to replace them all, or leaving it off when you meant just one. A date with one dash and one slash fixed is worse than either style. Do instead: leave instance_num out for a full style change, and use 1 or 2 only when you mean to touch exactly one separator.

6. Summary

SUBSTITUTE replaces old text with new text in a cell: `=SUBSTITUTE(P2,"-","/")` turns 11-2-2001 into 11/2/2001. Leave the instance number out and every occurrence changes. Add 1 or 2 and only that occurrence changes, so instance 1 on 11/2/2001 gives 11-2/2001. Square brackets in a tooltip mean the input is optional. Use it on text dates, not real ones; it matches capitals exactly.

7. Quiz

1. A text date reads 2024.03.15 and you need 2024-03-15. Which formula? A. `=SUBSTITUTE(P2,".","-",1)` B. `=SUBSTITUTE(P2,".","-")` C. `=SUBSTITUTE(P2,"-",".")` D. `=SUBSTITUTE(".","-",P2)`

2. `=SUBSTITUTE(P2,"/","-",2)` is applied to 07/04/2023. What comes back? A. 07-04-2023 B. 07-04/2023 C. 07/04-2023 D. 07/04/2023

3. You run `=SUBSTITUTE(H2,"/","-")` on a start date column and get five-digit numbers such as 39886 instead of dates. What's the most likely reason? A. SUBSTITUTE needs an instance number B. The cells hold real dates, which have no slashes underneath C. The slash needs no quotation marks D. SUBSTITUTE only replaces letters

Answer key

1. **B.** With no instance number, both dots change. A is wrong: it changes only the first dot. C is wrong: old and new text are swapped. D is wrong: the cell comes first.
2. **C.** Instance 2 replaces only the second slash. A is wrong: that needs no instance number. B is wrong: that's instance 1. D is wrong: one slash does change.
3. **B.** A real date is a number underneath, with no separators to replace, so the number comes back. A is wrong: the instance number is optional. C is wrong: text arguments need quotation marks. D is wrong: it replaces any characters, as the slashes and dashes show.

8. Assignment: one style for a mixed date column

Time: 15 minutes.

In a new column P on your practice sheet, type six text dates (start each with an apostrophe): three with dashes, three with slashes. Write down your target style. Use SUBSTITUTE to bring all six to that style, then use instance numbers to produce one example of each half-changed version.

Done checklist

- [ ] All six entries sit on the left of their cells (text)
- [ ] The target style is written down
- [ ] The cleaned column shows all six dates in the target style
- [ ] One instance-1 and one instance-2 example are shown, each with exactly one separator changed

Sample answer (illustrative)

> Target: slashes. Mixed: 11-2-2001, 4/9/2015, 6-30-2018 and three more. Q2: `=SUBSTITUTE(P2,"-","/")` 11/2/2001. All six now use slashes. Instance 1 on 4/9/2015: `=SUBSTITUTE(P3,"/","-",1)` 4-9/2015. Instance 2: 4/9-2015.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Count Calendar and Working Days with DAYS and NETWORKDAYS

Module 4: Fix and use dates.

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

1. Objective

Count the calendar days and the working days between a start date and an end date, using DAYS and NETWORKDAYS with your own holiday list.

2. Why it matters

"How long was this person with us?" and "how many working days did this project take?" are different questions with very different answers. These two formulas give you both for every row, without a calendar and a pen.

3. The teaching

Alex says this pair has "some specific use cases". Both take a start and an end date. They differ in what they count, and, annoyingly, in the order they want the dates.

DAYS: calendar days, end date first. DAYS gives "the range from this day to this day", every calendar day in between. Alex flags the order straight away: "we want to choose our end date ... that's kind of backward from what you think ... you think start date and date so you have to start with this one and then we're going to choose the start date". On your practice sheet, end dates in I and start dates in H:

```
=DAYS(I2,H2)
```

For Alex's first row, it returns 5056 days.

Alex's warning about date format. Alex runs DAYS on dates stored as text, the kind you made in the last two lessons, and says: "if you do it when it is uh in a date format it actually will not work ... you just gotta trust me".

[General knowledge, not from the video. Check before publishing.] Microsoft documents DAYS as accepting real dates, and also text that Excel can read as a date. On a standard setup, `=DAYS(I2,H2)` on real dates should work. The reason it failed for Alex isn't stated. So try it on your real dates first. If you get an error, check whether the cells are real dates or text that Excel can't read (for example a day-first date on a month-first system), and convert as in the earlier lessons.

NETWORKDAYS: working days, start date first. Alex describes NETWORKDAYS as "extremely similar except it takes out holidays and it takes out weekends". It tells you "how many working days ... has this person worked", or as he puts it, "how many days they've actually been in the office". And the order flips back: "this one's different this one says start date end date".

```
=NETWORKDAYS(H2,I2)
```

The results, Alex notes, are "dramatically lower because it's taking out weekends and holidays".

Holidays. NETWORKDAYS has a third, optional input for holidays. Alex says you "can specify ... extra holidays if you'd like but there are a already standard set holidays in there that it takes out".

[General knowledge, not from the video. Check before publishing.] This is the one point where Alex's description doesn't match Excel's documentation. NETWORKDAYS removes Saturdays and Sundays automatically. It has no built-in holiday list: it removes only the dates you supply in the third input. To leave out holidays, list them in a range, one date per cell, and point to it:

```
=NETWORKDAYS(H2,I2,$S$2:$S$12)
```

The dollar signs keep the holiday range fixed when you copy the formula down. One more difference: NETWORKDAYS counts both the start and the end date if they're working days, while DAYS gives the gap between them. A Monday-to-Friday week is 4 with DAYS and 5 with NETWORKDAYS.

[Instructor's method, not from the video.] Check both formulas on one short span you can count on your fingers, such as a single week, before you trust them on long spans. If DAYS gives a negative number, the dates are in the wrong order.

4. Worked example

Drawn from the source. Alex wants to know how long a person was at the company. He types `=DAYS(`, picks the end date first, then the start date, and gets 5056 days. He then writes NETWORKDAYS on the same two dates, start date first this time, and the result is much lower, the days actually worked once weekends and holidays are removed. The NETWORKDAYS figure itself isn't stated.

Illustrative. Run on Monday 6 January to Friday 10 January 2025: DAYS gives 4, NETWORKDAYS gives 5. Add a holiday list containing 1 January 2025 and the result stays 5, because that date falls outside the span.

The one thing that would have gone wrong: anyone relying on Alex's "standard set holidays" would report working days that still include public holidays, a few days too high for every year of employment.

5. Common mistake

Using the same date order in both formulas. `=DAYS(H2,I2)` with start first returns a negative number, and `=NETWORKDAYS(I2,H2)` with end first returns a negative count too. Do instead: remember "DAYS: end, start" and "NETWORKDAYS: start, end", and read the tooltip each time.

6. Summary

DAYS counts calendar days between two dates and takes the end date first: `=DAYS(I2,H2)`. Alex's example returned 5056. NETWORKDAYS counts working days and takes the start date first: `=NETWORKDAYS(H2,I2)`. It removes weekends; per Excel's documentation it removes only the holidays you list in its third input, not a built-in set as Alex says. Alex found DAYS worked on text dates; documentation says real dates work too.

7. Quiz

1. Start dates are in H and end dates in I. Which formula returns the calendar days between them as a positive number? A. `=DAYS(H2,I2)` B. `=DAYS(I2,H2)` C. `=NETWORKDAYS(I2,H2)` D. `=I2-DAYS(H2)`

2. You need the working days between H2 and I2, leaving out the public holidays listed in S2:S12. Which formula? A. `=NETWORKDAYS(H2,I2)` B. `=NETWORKDAYS(H2,I2,S2:S12)` C. `=DAYS(I2,H2,S2:S12)` D. `=NETWORKDAYS(I2,H2,S2:S12)`

3. A span runs from Monday to the following Monday. DAYS returns 7. What does NETWORKDAYS return, with no holidays listed? A. 7 B. 5 C. 6 D. 8

Answer key

1. **B.** DAYS takes the end date first. A is wrong: start first gives a negative number. C is wrong: NETWORKDAYS counts working days, and here the dates are reversed. D is wrong: DAYS needs two dates.
2. **B.** Start, end, then the holiday range. A is wrong: without the list, holidays are counted as working days. C is wrong: DAYS has no holiday input. D is wrong: the dates are reversed.
3. **C.** Both Mondays and the four weekdays between are working days: 6. A is wrong: that's the calendar gap. B is wrong: it misses the second Monday. D is wrong: there are only six weekdays in the span.

8. Assignment: tenure in days and working days

Time: 20 minutes.

On your practice sheet, list the public holidays where you work for the years your dates cover, one per cell, in column S. Add two columns: Days (DAYS) and Working days (NETWORKDAYS with your holiday list). Before filling them down, test both on one week you can count by hand.

Done checklist

- [ ] A holiday list in its own range, one date per cell, entered as real dates
- [ ] DAYS uses end date first and shows positive numbers on every row
- [ ] NETWORKDAYS uses start date first and points to the holiday range with dollar signs
- [ ] A one-week test is recorded with the hand count and both formula results
- [ ] Working days are lower than calendar days on every row

Sample answer (illustrative)

> Test week, Mon 6 Jan to Fri 10 Jan 2025: hand count 5 working days. DAYS 4, NETWORKDAYS 5. Correct. Row 2 (3/14/2009 to 6/30/2021): `=DAYS(I2,H2)` 4491 | `=NETWORKDAYS(H2,I2)` 3208 | `=NETWORKDAYS(H2,I2,$S$2:$S$100)` 3121, because 87 listed holidays fell on weekdays in the span.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

ACTION PLAN

Action Plan: Excel Formulas for Beginners

Instructor: [YOUR NAME] Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Use this the next time someone hands you a list and asks questions about it. Work through the steps in order: clean first, then build, then answer. Every formula you need is written out, with cell addresses you swap for your own. Anything marked "check" is a detail Alex The Analyst didn't give, so confirm it in your own version of Excel.

Before you start

Step 1. Work on a copy. Save a copy of the file and work there. Keep the original untouched. *If it goes wrong:* if you've already changed the original, use Ctrl+Z until you're back, then make the copy.

Step 2. Check how your Excel handles ranges. In an empty cell, type `=LEN(` and select a whole text column, then press Enter. *Decision point:* if a result appears in every row, your version spills, and you can hand whole columns to formulas the way Alex does. If you get one result or an error, write each formula for the first data row and copy it down (double-click the square at the cell's bottom-right corner). Check: IFS needs Excel 2019 or later; DAYS needs Excel 2013 or later.

Step 3. Write the questions down. List what you've been asked, in words, with every condition and boundary spelled out: "total salary of female staff aged 30 or over", not "female salaries". *If it goes wrong:* if a question has a boundary you can't settle ("over 30" or "30 and over"), ask before you build it. The formula can't decide for you.

Clean the text

Step 4. Find stray spaces. Next to each text column, add `=LEN(C2)-LEN(TRIM(C2))`. Anything above 0 is a cell with extra spaces. *Decision point:* if every row shows 0, skip to Step 6.

Step 5. Remove them. Add `=TRIM(C2)`, copy the result, and Paste Special > Values over the original column. Re-run Step 4: every row should now show 0. *If it goes wrong:* if a cell still shows extra characters after TRIM, it may hold a different kind of space. Check: look up non-breaking spaces for your version.

Step 6. Flag entries of the wrong length. Decide the correct length from real entries, counting dashes and spaces. Then add `=IF(LEN(L2)<>9,"Check","OK")` with your column and length. *Decision point:* for each "Check" row, decide whether to correct it, ask the owner, or leave it out of your results, as Alex did with bad social security numbers. Write down which.

Fix the dates

Step 7. Find out what your dates are. Look at the alignment: real dates sit on the right of the cell, text on the left. Or set one cell to General: a five-digit number means a real date. *Decision point:* real dates go to Step 9 if you need text, or straight to Step 13 for day counts. Text dates go to Step 8.

Step 8. Bring text dates to one style. Write down the target style. Replace every separator with `=SUBSTITUTE(P2,"-","/")`. To change only one occurrence, add the instance number: `=SUBSTITUTE(P2,"/","-",1)` for the first, 2 for the

second. Paste the results as values. *If it goes wrong:* if you get five-digit numbers instead of dates, those cells were real dates, not text. Go back to Step 7.

Step 9. Convert real dates to text when you need text. `=TEXT(H2,"dd/mm/yyyy")`, or another format code that ends with the year. Copy and Paste Special > Values into a new column. *If it goes wrong:* if the output looks wrong or errors, the format code is wrong. Check: format letters can vary with language settings.

Build new columns

Step 10. Pull pieces out. First characters: `=LEFT(B2,3)`. Last characters: `=RIGHT(A2,1)`. Year from a text date that ends in the year: `=RIGHT(O2,4)`. Or, from a real date, `=YEAR(H2)`. *If it goes wrong:* if a number test on the result misbehaves, the result is text. Wrap it: `=VALUE(RIGHT(A2,1))`.

Step 11. Join pieces together. Full name: `=CONCATENATE(B2," ",C2)`. Patterned email: `=CONCATENATE(TRIM(B2),".",TRIM(C2),"@example.com")`. *Decision point:* generated email addresses follow a pattern; they aren't confirmed addresses. Don't send to them until each is checked.

Step 12. Label rows. Two groups: `=IF(D2>=30,"30 and over","Under 30")`, with the sign matching the rule you wrote in Step 3. Several groups: list every distinct value with a filter first, then `=IFS(F2="Salesman","Sales",F2="HR","People",TRUE,"Other")`. Put narrow tests before broad ones. *If it goes wrong:* #N/A from IFS means a row matched nothing. Add or fix the `TRUE` pair.

Answer the questions

Step 13. Plain summaries. `=MAX(G2:G10)`, `=MIN(G2:G10)`, `=SUM(G2:G10)`, `=COUNT(G2:G10)`. Use COUNTA for text columns. Highlight the range and compare with the count at the bottom of the window. *If it goes wrong:* if COUNT is lower than the window count, a number column holds text. Find it and fix it. If a result looks like the wrong kind of value, check the cell's format before the formula.

Step 14. One-condition answers. How much: `=SUMIF(G2:G10,">50000")`, or to test one column and add another, `=SUMIF(E2:E10,"Female",G2:G10)`. How many: `=COUNTIF(G2:G10,">45000")`. Conditions with a sign go in quotation marks. *Decision point:* "over" is `>`; "or more" is `>=`. Check each against Step 3.

Step 15. Several-condition answers. Build `=COUNTIFS(E2:E10,"Female",D2:D10,">=30")` one pair at a time, checking the count after each. Then put the range to add in front: `=SUMIFS(G2:G10,E2:E10,"Female",D2:D10,">=30")`. *If it goes wrong:* #VALUE! means the ranges differ in size; make them all run over the same rows. A wrong total usually means the sum range was put last, SUMIF style.

Step 16. Day counts. Calendar days, end date first: `=DAYS(I2,H2)`. Working days, start date first, with your holiday list fixed in place: `=NETWORKDAYS(H2,I2,$S$2:$S$40)`. *Decision point:* NETWORKDAYS removes weekends on its own. Check: on Excel's documented behavior, it removes only the holidays you list, so build the list. *If it goes wrong:* a negative result means the dates are in the wrong order. An error means the dates aren't readable as dates; go back to Step 7.

Before you hand it over

Step 17. Check by eye. For each answer, find at least one matching row in the data and confirm it, the way Alex checks every result against his table. Test boundaries on purpose: the row exactly on 30, the salary exactly on 50,000. *If it goes wrong:* if a check fails, fix the formula, not the data, then re-run the check.

Step 18. Fix the results in place. If the file is going to someone else, paste formula columns as values where they shouldn't change, and note in a cell or comment which formulas produced them.

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

SECTION C

C. Listing and Launch Notes

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Course title and subtitle options

Ranked by clarity, most clear first.

1. **Excel Formulas for Beginners**: Clean, label, total and count your data with 19 everyday functions
2. **19 Everyday Excel Formulas**: From MAX and IF to SUMIFS, TEXT and NETWORKDAYS, step by step
3. **Excel Formulas for Working With Lists**: Answer real questions about staff, sales and dates without doing it by hand
4. **Clean Data, Clear Answers in Excel**: A beginner's course in text, date and conditional formulas
5. **The Excel Formulas Analysts Use on the Job**: A short course based on Alex The Analyst's tutorial (Ranked last: "use on the job" leans on Alex's own experience and could read as a promise about work, which the course can't make.)

Course description

Excel Formulas for Beginners: Clean, label, total and count your data with 19 everyday functions Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

Alex The Analyst, a data analyst who teaches on YouTube, picked out the Excel formulas he found most useful, almost all of them learned on the job, and demonstrated them on a small employee table. This course teaches his 19 functions in the order a beginner needs them, with every formula written out, a worked example, a quiz and a hands-on task in each lesson.

Who it's for. People who already use Excel to look at lists (staff, customers, orders, dates) and want to start answering questions with formulas instead of by eye. You should be able to open a workbook, type in cells and select a range. You don't need to have written a formula before.

What you'll be able to do. Clean a messy list in Excel and answer questions about it with 19 everyday functions: find the highest and lowest values, label rows with IF and IFS, total and count only the rows that meet your conditions, fix stray spaces, split and join text, convert and reshape dates, and count the days or working days between two dates.

What's inside. - A welcome lesson that has you build a practice sheet with the edge cases the lessons test - Module 1, Write your first formulas: MAX, MIN, SUM, COUNT and IF - Module 2, Make formulas check conditions: IFS, SUMIF, COUNTIF, SUMIFS and COUNTIFS - Module 3, Clean and build text: LEN, TRIM, LEFT, RIGHT and CONCATENATE - Module 4, Fix and use dates: TEXT, SUBSTITUTE, DAYS and NETWORKDAYS - A worked example, a three-question quiz with answer key, and a practical assignment in every lesson - One action plan for cleaning and summarizing any new list, with what to do when each step goes wrong

Who it's not for. Anyone looking for lookups (VLOOKUP, XLOOKUP), pivot tables, charts, macros or advanced analysis. Experienced Excel users. Anyone looking for a qualification or a guaranteed job outcome: the course teaches formulas, nothing more.

About 62 minutes of lessons, plus assignments.

Welcome email

Subject: Your first Excel formula takes about five minutes

Hi [STUDENT FIRST NAME],

Welcome to Excel Formulas for Beginners.

This course is built on Alex The Analyst's tutorial "Formulas in Excel | Excel Tutorials for Beginners". Alex is a data analyst, and he picked out the formulas he found most useful at work. You're going to learn all 19 of them, in an order that builds one on the next.

Start with the welcome lesson. It takes a few minutes and ends with you building a small practice sheet, nine made-up people with a few deliberate traps in the data. Every lesson after that has you type the formulas into that sheet and check the results yourself, which is the part that makes them stick.

One thing to know up front: Alex works fast and doesn't say which version of Excel he's using. Where an Excel detail matters and he doesn't mention it, such as where the quotation marks go or how to copy a formula down, the course adds it and labels it clearly.

Here's the link to the first lesson: [COURSE LINK]

[YOUR NAME]

Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst.

(Draft: [STUDENT FIRST NAME], [COURSE LINK] and [YOUR NAME] must be filled in before sending.)

Upload checklist (any course platform)

File names and load order. Load in this order, one lesson per page or unit:

1. m0-l0-welcome.md (Welcome: Excel Formulas for Beginners)
2. m1-l1-find-highest-and-lowest-with-max-and-min.md (Module 1, Lesson 1)
3. m1-l2-total-and-count-with-sum-and-count.md (Module 1, Lesson 2)
4. m1-l3-label-every-row-with-if.md (Module 1, Lesson 3)
5. m2-l1-sort-rows-into-labels-with-ifs.md (Module 2, Lesson 1)
6. m2-l2-total-and-count-with-sumif-and-countif.md (Module 2, Lesson 2)
7. m2-l3-total-and-count-with-sumifs-and-countifs.md (Module 2, Lesson 3)
8. m3-l1-check-and-clean-text-with-len-and-trim.md (Module 3, Lesson 1)
9. m3-l2-pull-characters-with-left-and-right.md (Module 3, Lesson 2)
10. m3-l3-join-names-and-emails-with-concatenate.md (Module 3, Lesson 3)
11. m4-l1-turn-dates-into-text-and-pull-the-year.md (Module 4, Lesson 1)
12. m4-l2-swap-date-separators-with-substitute.md (Module 4, Lesson 2)
13. m4-l3-count-days-with-days-and-networkdays.md (Module 4, Lesson 3)
14. action-plan.md (after the final lesson, or as a downloadable)

Module titles for the platform: - Module 1: Write your first formulas - Module 2: Make formulas check conditions - Module 3: Clean and build text - Module 4: Fix and use dates

Do not upload A-source-audit-and-course-map.md or this file. They are production notes.

Before publishing, check:

- [ ] [YOUR NAME] is replaced everywhere (search every file)
- [ ] The credit line "Based on "Formulas in Excel | Excel Tutorials for Beginners" by Alex The Analyst." appears on every lesson page and on the course sales page
- [ ] Every formula has been typed into a real workbook and returns the result the lesson says (the illustrative ones included)
- [ ] Every "[General knowledge, not from the video. Check before publishing.]" passage has been checked against Microsoft's documentation for the Excel versions your learners use, then kept with a source note or rewritten. Decide whether learners see the label.
- [ ] Every "[Instructor's method, not from the video.]" label is kept, or replaced with wording that still makes clear the method isn't Alex's
- [ ] Formulas display as code, with straight quotation marks (some platforms convert them to curly quotes, which break formulas when copied)
- [ ] Quizzes render with four options each and the answer key hidden or placed after the questions
- [ ] Checklists render as checkboxes, or as plain lists if the platform doesn't support them
- [ ] The tables in the welcome lesson and in the assignments render correctly on mobile
- [ ] The welcome email's placeholders are filled and its link works
- [ ] A link to the original tutorial (https://www.youtube.com/watch?v=XRPyj7cKVsQ) is included if Alex The Analyst's permission terms ask for it
- [ ] No screenshots or clips from the tutorial are used unless the permission covers them
- [ ] Spelling and style are consistent (US spelling throughout: summarize, favorite, labeled)

Unable to verify

Things the source states or implies that could not be confirmed from the source itself.

- The date the tutorial was published, and therefore which Excel version and behavior it reflects.
- The Excel version Alex uses. His whole-column formulas suggest Microsoft 365, unconfirmed.
- The value tested in the third IFS condition (the one that returns "Give Christmas Bonus"). Inaudible.
- The exact format code Alex typed for TEXT. Only "dd slash mm yyyy" is spoken; the course uses "dd/mm/yyyy".
- How Alex entered 11-2-2001 so that it stayed as text, and which paste option he used after TEXT.
- Which cell is on the right and which on the left in Alex's alignment comparison. The course uses Excel's default (dates right, text left).
- That the SUMIFS example's second condition ("greater than 30") was tested on the age column. Inferred from context.
- The exact email pattern Alex built (order of pieces, capitals). The course uses first name, dot, last name, "@gmail.com".
- The range used for COUNT (result 9), the full MAX date in 2013, Pam Beesly's salary, the total SUM, the LEFT results and the NETWORKDAYS result. Shown on screen, not spoken.

- Alex's statement that DAYS "will not work" on date-formatted cells. It conflicts with Microsoft's documentation of DAYS; the course flags it.
- Alex's statement that NETWORKDAYS has "standard set holidays" built in. It conflicts with Microsoft's documentation, which says it removes only the holidays you list; the course flags it.
- The length of a US social security number. Alex says "i think it's nine". The course confirms nine digits from general knowledge and notes that dashes make it 11 characters.
- The character names and spellings (Pam Beesly heard as "pam beasley"; Stanley and Kevin have no surnames in the source).
- The GitHub link for Alex's workbook.

Assumed or added

Everything here came from general knowledge or the instructor, not from the source. Each general-knowledge item is labeled in the lesson where it appears.

General knowledge added (to be checked): - Version notes: whole-range results ("spilling") in Microsoft 365, copying a formula down in other versions, IFS in Excel 2019 or later, DAYS in Excel 2013 or later. (m0-l0, and each lesson that spills) - Excel adds a missing closing parenthesis in simple formulas; clearing a format via Home > Number Format. (m1-l1) - COUNT counts only numbers (dates included); the status bar Count counts non-empty cells; COUNTA. (m1-l2) - Quotation marks around text results; the comparison operators. (m1-l3) - The `TRUE` catch-all pair in IFS; first-true-wins order; text matching ignores capitals. (m2-l1) - Quotation marks around conditions with operators; testing one column and adding another with sum_range; `>=` to include the boundary. (m2-l2) - Equal-size ranges and #VALUE!; "and" only; SUMIFS with one condition. (m2-l3) - LEN counts spaces and dashes; social security numbers with dashes are 11 characters; Paste Special > Values. (m3-l1) - LEFT and RIGHT return text; VALUE; num_chars optional; MID exists; text ranks above numbers in comparisons. (m3-l2) - CONCAT and the ampersand as alternatives. (m3-l3) - Date serial numbers from January 1, 1900; default alignment; the apostrophe for text entry; Paste Special > Values; YEAR; locale-dependent format codes. (m4-l1) - SUBSTITUTE works on text, returns a serial number from a real date, is case-sensitive; Find and Replace as an alternative; "/" is a forward slash. (m4-l2) - DAYS accepts real dates; NETWORKDAYS removes only listed holidays; NETWORKDAYS counts both ends; absolute references with dollar signs. (m4-l3) - Non-breaking spaces as a reason TRIM may not clean a cell. (action plan only)

Added by the instructor: - The practice sheet layout (columns A to I, rows 2 to 10) and its four planted traps, mirroring the edge cases in Alex's own examples. - The checking methods: check every result by eye; compare COUNT with the window count; divide SUM by COUNT for an average; say the IF rule aloud with its boundary; list distinct values with a filter before writing IFS; run a COUNTIF beside every SUMIF; build COUNTIFS one condition at a time; compare LEN before and after TRIM; trim inside CONCATENATE; choose a format code that ends in the year; set a target style before SUBSTITUTE; test DAYS and NETWORKDAYS on one countable week. - The warning not to send to generated email addresses. - All worked examples except those marked "drawn from the source", and every sample answer. The people in them (Dana, Rosa Delgado, Tom Reyes, Nadia Park, Sam Oduya, Ana Silva, Lena Ford, Ben Cho, Omar Haddad, Kai Moore) and all their figures are invented. The serial number 39886 for 3/14/2009 and the day counts in m4-l3's sample were calculated, not taken from the source. - The audience, level, outcome sentence and course size, chosen because the questionnaire said "use defaults". - The course titles, the description and the welcome email.

Still required before publishing

- **Permission on file.** Written permission from Alex The Analyst that covers selling a course built on this tutorial, with his approved credit wording. The questionnaire says permission exists. This pack hasn't seen it.
- **Instructor name.** Replace [YOUR NAME] in every file and in the welcome email.
- **Test every formula.** Type each formula in every lesson, quiz and sample answer into a real workbook, in at least Microsoft 365 and one older version if learners might use one, and confirm the results. This matters most for the DAYS and NETWORKDAYS corrections and the SUBSTITUTE-on-real-dates note.
- **Fact-check the general-knowledge passages** against Microsoft's function documentation.
- **Settle the two conflicts.** Decide how strongly to correct Alex's DAYS and NETWORKDAYS statements. The current wording attributes his claims and gives the documented behavior; if you can, ask Alex what he saw.
- **The practice file.** Either get the GitHub link to Alex's workbook (and permission to point learners at it) or supply a downloadable practice workbook built from the welcome lesson's layout.
- **Date the source.** Find the tutorial's publish date and add it to the welcome lesson's version note.
- **Decide on the labels.** Choose whether learners see the "[General knowledge…]" and "[Instructor's method…]" labels, or a softer version. Removing them entirely would present added material as Alex's.
- **Sponsor separation.** Confirm the course makes no reference to Udemy beyond the welcome lesson's one-line note and implies no endorsement.
- **Character names.** The sample data in Alex's workbook uses names from The Office. The course quotes them only as they appear in his demonstration. Confirm that's acceptable for a paid product, or remove them.
- **Fill the welcome email placeholders** and test the course link.
- **Proofread once more** for US spelling, straight quotation marks inside formulas, and stray em dashes after any edits.

What a better source would improve

Blunt version: this is a strong, honest tutorial for its length, but it's a fast screen demo, and a course can only be as precise as what's spoken.

- **The formulas themselves, spoken or shown in text.** Almost every formula is typed on screen and never read out. The course reconstructs them from context, which is why every cell address is illustrative and several details (the TEXT format code, the email pattern, the third IFS condition) are flagged.
- **The workbook.** Alex says he'll put it on GitHub, but there's no link. With it, learners could check their answers against his exact data, and the course could quote his actual results instead of saying "not stated".
- **The version.** Naming the Excel version would settle whether learners can hand whole columns to IF, LEN and TRIM, and would explain his DAYS result.
- **Correct claims on DAYS and NETWORKDAYS.** Two statements conflict with Excel's documented behavior. A source that showed the holiday input in use and ran DAYS on real dates would remove two of the course's biggest caveats.
- **The syntax details beginners trip on.** Quotation marks, comparison operators, copying formulas down, Paste Values and the meaning of #N/A are all either unspoken or mentioned in passing. Each is a general-knowledge patch in the course.
- **Depth on the short functions.** SUM and COUNT get under a minute, LEN and TRIM not much more. A source that showed one real cleaning job end to end, from raw export to report, would give the course a project instead of a set of drills.

- **More functions.** Alex invites requests for missing formulas. Lookups (VLOOKUP, XLOOKUP), MID and AVERAGE would round out a beginner course; none are in the source, so none are here.
- **No sponsor segment.** About 330 words of the opening are an advertisement, unusable for teaching.