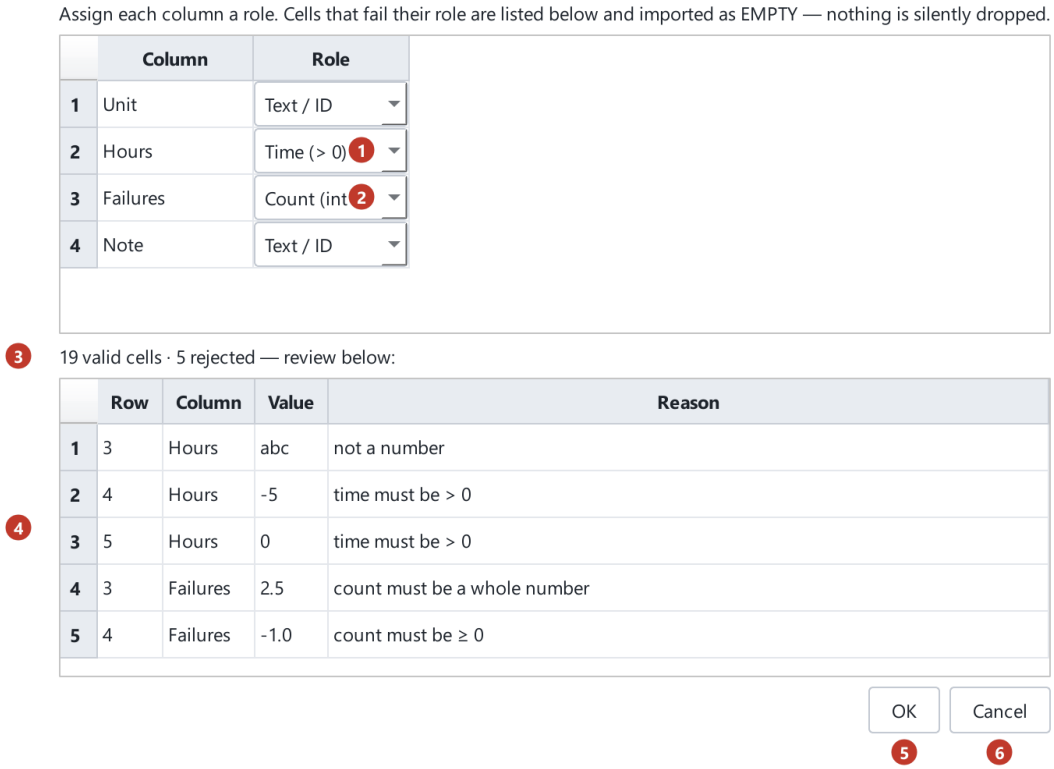
Data import: a file into the worksheet, every cell checked against its column's role

Menu: File ▸ Import Data (Excel / CSV / JSON)...

Part 1 — What the function does

What it does. File ▸ Import Data (Excel / CSV / JSON)... reads a file into the worksheet: an Excel workbook (its first sheet), a CSV file or a JSON table, with the first row as the column names. Before anything reaches the worksheet, the import wizard asks what each column holds - its role - and checks every cell against it. A cell that fails its role is listed with its row, its value and the reason, and is imported as an empty cell: nothing is guessed and nothing is dropped without a line.



The import wizard with the roles set

- ① and ② The Role of each column, chosen from a list: here Hours and Failures.
- ③ The count line: how many cells pass their roles and how many are rejected.
- ④ The table of rejected cells: Row, Column, Value, Reason.
- ⑤ OK: imports the file into the worksheet.
- ⑥ Cancel: closes the wizard and changes nothing.

Role	A cell passes when it is	The reasons for a rejected cell
Time (> 0)	a number above 0	not a number, time must be > 0, not finite
Number	any finite number	not a number, not finite
Count (int ≥ 0)	a whole number of 0 or more	not a number, count must be a whole number, count must be ≥ 0
Text / ID	anything	—
Ignore column	the column is not imported	—

From which data. A file with one header row and one column per quantity: `.xlsx` or `.xls`, `.csv`, or `.json` (a list of records).

When an engineer uses it. When the data already sits in a spreadsheet, an export of a maintenance system or a log file, so it is read in once instead of typed.

When not to use it.

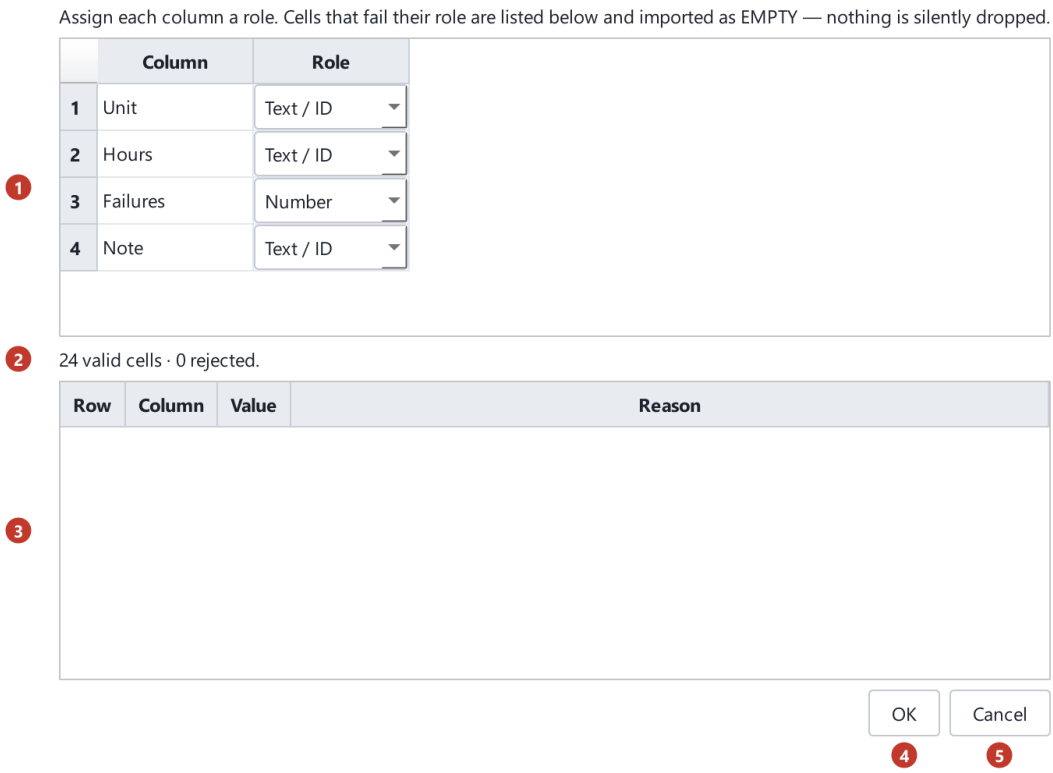
Situation	Use instead
The data sits in a database	File ▸ Import from SQL Database…: the result of one SELECT query goes through the same wizard
A few values only	Type or paste them into the worksheet (chapter 1.2)

Part 2 — The example's input data

Setting. A maintenance clerk exports the run log of six pumps from a spreadsheet as a CSV file. Some cells were typed carelessly.

Input	Value
File in the worked-example zip	`1-Data-Import/Example-1.csv`
Its columns	`Unit` (U1 … U6), `Hours` (120, 250.5, abc, -5, 0, 400), `Failures` (3, 0, 2.5, -1, 7, 1), `Note` (n1 … n6)
The worksheet before	empty

The file as the wizard opens it:

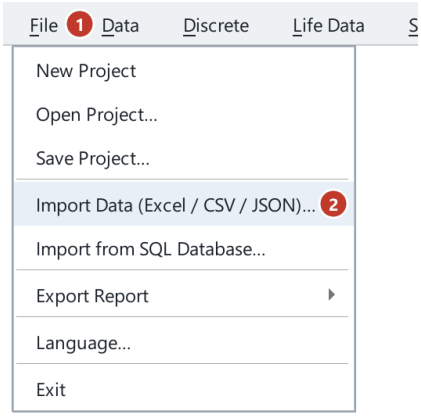


The import wizard as it opens

- ① The file's four columns, each with the role ReliaEn guessed: Text / ID for Unit, Hours and Note, Number for Failures. A column with one cell that is not a number is guessed as text.
- ② The count line for the guessed roles: 24 valid cells · 0 rejected. - text accepts anything, so nothing is rejected yet.
- ③ The table of rejected cells, empty.
- ④ OK.
- ⑤ Cancel.

Part 3 — How to use it, step by step

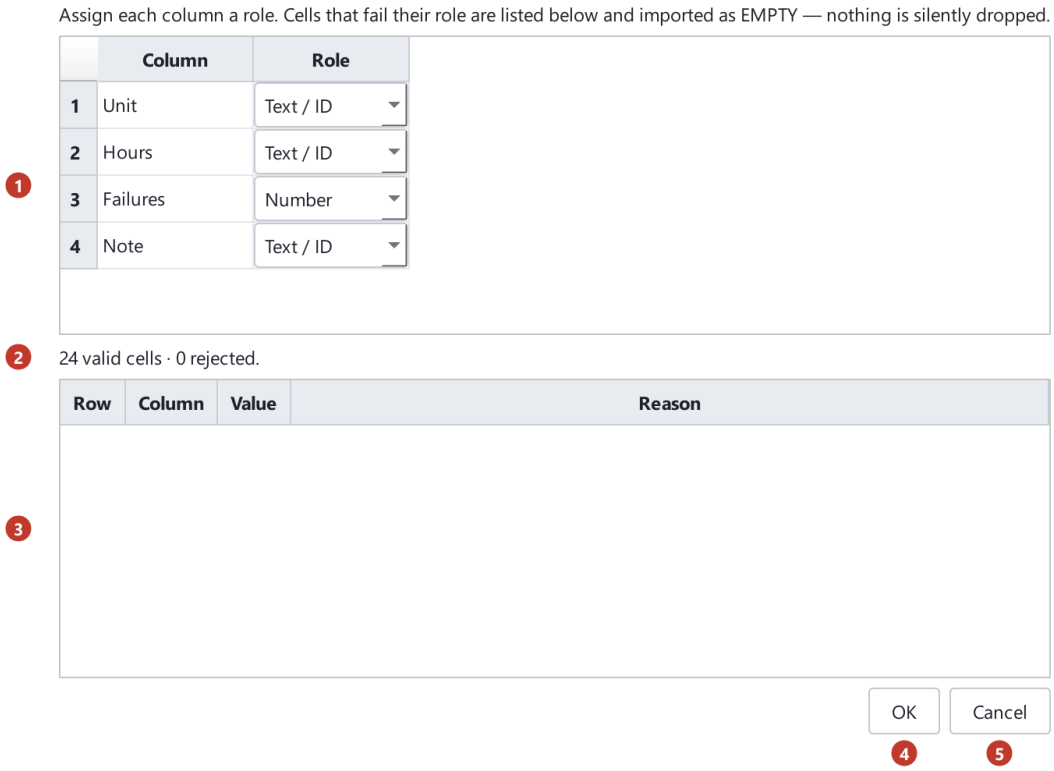
1. **Choose File ▸ Import Data (Excel / CSV / JSON)…** in the menu bar.



Step 1 — the menu path

- ① File: the menu.
- ② Import Data (Excel / CSV / JSON)…: the entry to click.

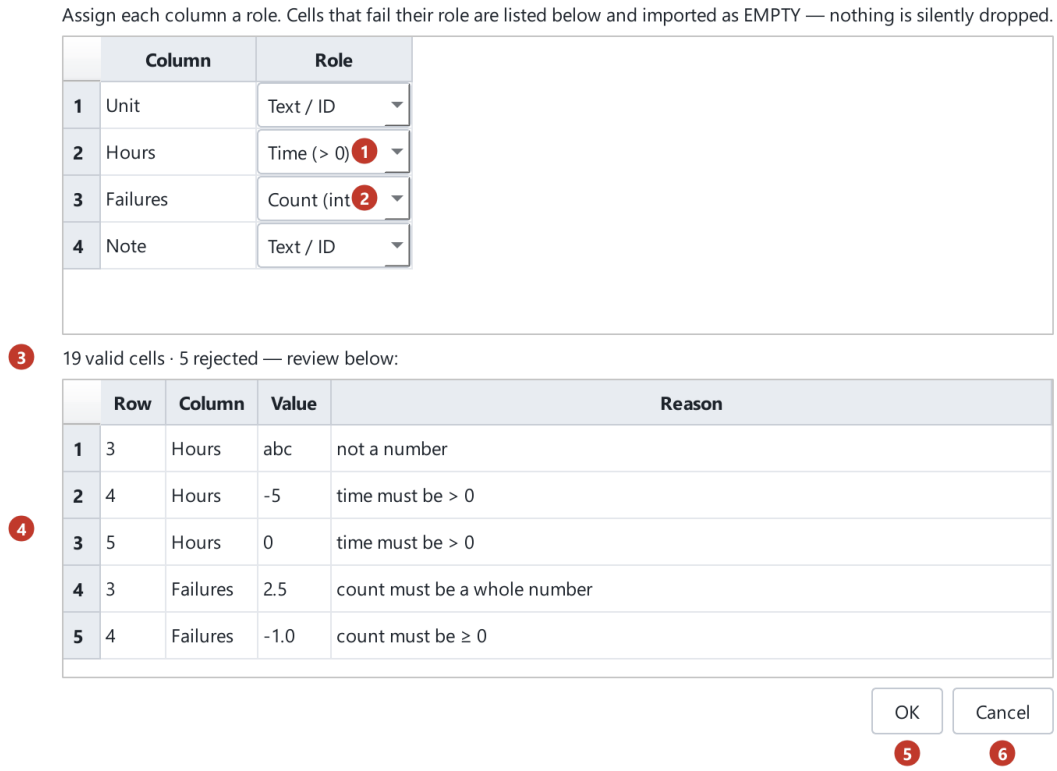
2. **Choose Example-1.csv in the file chooser Import data and click Open.** The wizard Import wizard — map columns opens with the guessed roles, as in the picture of Part 2.



Step 2 — the wizard as it opens

- ① The four columns with their guessed roles.
- ② The count line for the guessed roles: 24 valid cells · 0 rejected. - text accepts anything, so nothing is rejected yet.
- ③ The table of rejected cells, empty.
- ④ OK.
- ⑤ Cancel.

3. **Set the role of Hours to Time (> 0) and of Failures to Count (int ≥ 0).** Leave Unit and Note as Text / ID. The count line and the table follow each change at once.



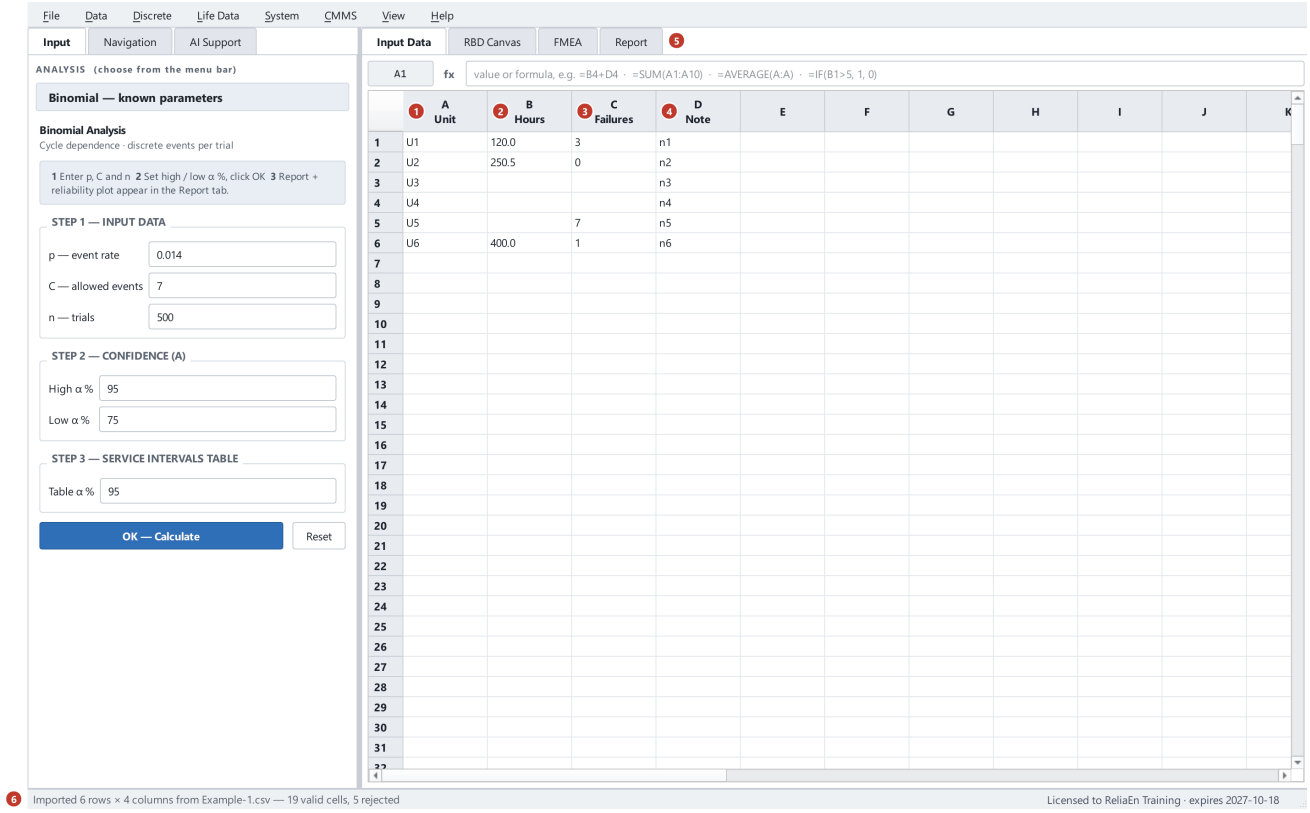
Step 3 — the roles set

- ① Hours: Time (> 0).
- ② Failures: Count (int ≥ 0).
- ③ The count line: 19 valid cells · 5 rejected — review below:.
- ④ The five rejected cells.
- ⑤ OK.
- ⑥ Cancel.

4. ③ ④ **Read the count line and the table before you import.** Row 3 Hours abc - not a number; rows 4 and 5 Hours -5 and 0 - time must be > 0; row 3 Failures 2.5 - count must be a whole number; row 4 Failures -1 - count must be ≥ 0.

5. ⑤ **Click OK.**

Part 4 — The result



The worksheet after OK

- ① to ④ Columns A to D, named Unit, Hours, Failures and Note, 6 rows each. The five rejected cells are empty: Hours of U3, U4 and U5, Failures of U3 and U4.
- ⑤ The Input Data tab, brought to the front.
- ⑥ The status line: Imported 6 rows × 4 columns from Example-1.csv — 19 valid cells, 5 rejected. It stays until the next status message.

Unit	Hours	Failures	Note
U1	120.0	3	n1
U2	250.5	0	n2
U3	(empty)	(empty)	n3
U4	(empty)	(empty)	n4
U5	(empty)	7	n5
U6	400.0	1	n6

The project now has unsaved changes (chapter 1.4).

Part 5 — How to interpret the results

What the counts mean. The file has 6 rows of 4 columns, 24 cells. Five fail their roles, so 19 are imported with a value and the five are imported empty. ReliaEn never changes a value to make it pass: -5 is not read as 5, and 2.5 is not rounded to 2 or 3.

What the empty cells do in an analysis. An analysis that reads Hours sees three hours and three blanks, never the bad values. Correct the cells at the source - the row numbers in the wizard are the file's data rows - and import again, or type the corrected values into the worksheet.

The import replaces the worksheet. When the worksheet already holds data, the wizard says so in one line above its buttons, The current worksheet will be replaced., and OK replaces every column. Cancel changes nothing. The project's reports stay either way.

The same data in other files. An Excel workbook or a JSON table with the same columns gives the same worksheet and the same counts. The worksheet holds 500 rows: a longer file imports its first 500, and ReliaEn names the rows it did not import in a box.

From a database. File ▸ Import from SQL Database... takes a database path or address and one query; only a query that starts with SELECT (or WITH) runs, and a SQLite file is opened read-only. The result goes through the same wizard.