Before you trust an automation: the 12-check test sheet

Twelve checks to run before an automation touches real records. Tick each one only when you have seen the pass with your own test data.

Built with AI assistance. Reviewed by a person, tested by our own test suites.

☐ 1. Empty input

Do: Feed it an empty file, a blank row in the middle and a record with a required field left blank.

Pass: Blank rows are skipped. A blank required field is listed with a reason. Nothing is filled in by guess.

Fail: The run crashes, writes a zero where a value was missing, or stops without saying why.

☐ 2. Messy input

Do: Mix date formats (10/04/2025 next to 02 Oct 2025), add stray spaces, capital letters and a currency sign in a number column.

Pass: Each value is read the same way every time, or the row is set aside with its reason.

Fail: Rows disappear without a trace, or a date is read with day and month swapped.

☐ 3. Duplicate input

Do: Enter the same record twice: once as typed, once with different capital letters and a trailing space.

Pass: The first copy counts. Later copies are flagged and left out of the totals.

Fail: Totals double, or the other app gets two records for one.

☐ 4. Run it twice

Do: Run the same job again straight away, with no new data.

Pass: The second run creates nothing and changes nothing.

Fail: New copies appear, dates change, or the same customer gets a second message.

☐ 5. Break the connection halfway

Do: Stop the run in the middle (cut the network or end the process), then start it again.

Pass: After the restart, the end result equals one clean run.

Fail: Half-written records, gaps, or doubles after the restart.

☐ 6. Hit a limit

Do: Push past a rate limit, a size limit or a time limit: a 'too many requests' answer, a very large file, a very long run.

Pass: It waits and retries, or pauses and resumes from a saved point, and it says so.

Fail: It stops halfway with no note, or keeps hammering the service.

☐ 7. Count the cost of a retry

Do: Make one service fail twice before it answers, then count the calls and the time.

Pass: You know how many extra calls a retry costs and where the retries stop.

Fail: Retries with no limit, or extra calls you only notice on the bill.

☐ 8. Check who sees the data

Do: Start with test data. Read the logs and outputs for keys, passwords and personal details before real data goes in.

Pass: No keys or personal details in logs or outputs. Real data goes in last.

Fail: An access key or a customer email address printed in a log file.

☐ 9. Find the reviewer and the stop button

Do: Find where a person looks before anything hard to reverse happens, and how to pause the job.

Pass: Doubtful records go to a review list. The job can be paused without leaving half a result.

Fail: Everything is written straight through, and nobody knows how to stop it.

☐ 10. Plan the undo

Do: Before the first real run, write down how you would undo it: a copy of the file, a backup, or a list of every record the run wrote.

Pass: You can put things back as they were, from a list of what changed.

Fail: The only record of what changed is the changed data itself.

☐ 11. Read tomorrow what happened today

Do: After a run, open its log or report. Count records in, records out and records set aside.

Pass: The counts add up, and every record set aside has a reason.

Fail: No log, or counts that do not add up.

☐ 12. Change the format on purpose

Do: Rename a column, reorder the columns, or feed it a layout it has not seen before.

Pass: It stops writing and names what changed.

Fail: Values land in the wrong fields and the run still reports success.

Built with AI assistance. Reviewed by a person, tested by our own test suites.

Made-up data in our own tests. Not legal, tax or security advice. Not affiliated with any tool named.

The web version of this sheet adds notes from our own kit test runs.

breakitfirst.pages.dev | teambreakitfirst@gmail.com | YouTube @BreakItFirst | X @breakitfirstlab