

The CRM cleanup checklist

Twelve checks that find the duplicate records, dead deals and silent data rot costing you work you already won. Run it in an afternoon, on any CRM.

How to use this

1. **Export first.** One CSV or spreadsheet of contacts, one of companies, one of open deals. Work from the export, not the live system, so nothing is changed by accident.
2. **Work top down.** The checks are ordered by damage found per minute spent. If you have one hour rather than an afternoon, the first four are the hour.
3. **Write the number down.** Each check has a line for what you found. The counts are the argument for fixing it properly, and the baseline you measure the next pass against.
4. **Fix formatting before merging.** Standardising phone numbers and company names first will surface duplicates that a straight de-duplication pass cannot see.

Every check here is one you can run by hand with filters and a sort. Nothing in this document requires our product, and the counts you write in it are your own.

☐ 01

Exact-match duplicates

Sort by email, then by phone. Anything appearing twice is a merge candidate. Check which record holds the real history before you pick a survivor: the newer row is often the emptier one.

Why it hurts: two half-histories of one customer, so the follow-up and the invoice chase different addresses.

FOUND

☐ 02

Near-match duplicates

The ones that survive an exact-match pass. Sort alphabetically by company and read: Acme Denta1, Acme Denta1 Ltd and ACME DENTAL are one account in three rows.

Why it hurts: account-level reporting splits three ways and every total is quietly wrong.

FOUND

☐ 03

Records with no owner

Filter for an empty owner or assignee. Anything unassigned is nobody's job, which is usually why it has not moved since the day it was created.

Why it hurts: unowned records do not appear in anyone's pipeline review, so nobody ever decides about them.

FOUND

☐ 04

Deals with no activity in 30 days

Not to delete: to decide. Every one is either a follow-up you owe someone or a loss you have not admitted yet. Both are worth knowing before the forecast goes out.

Why it hurts: a pipeline padded with stalled deals makes every forecast optimistic and every review longer.

FOUND

☐ 05

Missing fields that you actually act on

Pick the three fields you genuinely use to make a decision, and count the records that lack them. Ignore the rest. A column being empty is only a problem if something depends on it.

Why it hurts: segments and automations silently skip the records with gaps, so the gap compounds.

FOUND

☐ 06

Broken email formats

Filter for addresses with no @, two of them, or no dot after the domain. This is a syntax check, not a deliverability check: it finds typing mistakes, not dead mailboxes.

Why it hurts: far cheaper to find with a filter than with a bounce and a damaged sending reputation.

FOUND

☐ 07

Role addresses standing in for people

Count the info@, sales@ and office@ addresses sitting on a contact record. They are not wrong, but they are not a person, and treating them as one skews everything you conclude about engagement.

Why it hurts: a named contact leaves and nobody notices, because the shared mailbox keeps answering.

FOUND

☐ 08

Inconsistent phone and address formats

Count the distinct shapes in the phone column. Standardising costs nothing and makes every later de-duplication pass more accurate, which is why it belongs before the merging and not after.

Why it hurts: the same number stored four ways defeats matching, so duplicates survive the cleanup.

FOUND

☐ 09

Contacts with no company link

Orphaned people are invisible to any account-level view: they will not appear under the company, in its history, or in anything you filter by account.

Why it hurts: you call an account thinking you have one contact there and find three you forgot.

FOUND

☐ 10

Closed-won deals with no invoice

Cross-reference won deals against what was actually billed. Do this one even if you skip everything else on the page.

Why it hurts: this is work you delivered and never charged for. It pays for the afternoon on its own.

FOUND

☐ 11

Stages and fields that no longer match how you sell

If half the pipeline sits in one stage, that stage is doing no work. If a custom field is empty on most records, it is a tax on every record created. Delete both.

Why it hurts: every optional field makes the form longer, and a longer form gets filled in less honestly.

FOUND

☐ 12

Permissions you have outgrown

Who can export the entire database? Who still has an account from a role they left, or a job they no longer do? Answer it deliberately rather than by accident.

Why it hurts: the cost of this one is not felt gradually. It is felt all at once, on the day it matters.

FOUND

Before you merge anything

1. **Keep the export.** It is your evidence of what the data looked like before you touched it.
2. **Standardise first, merge second.** Then re-run checks 1 and 2. The second pass almost always finds records the first could not.
3. **Decide the survivor deliberately.** Most complete, most recent activity, or the one holding the invoice history. Pick a rule and apply it consistently.

The second cleanup is the one worth avoiding.

Every check in this document is a rule a machine can run continuously. That is the whole difference between a cleanup and a system that does not need one. The rot restarts the Monday after you finish, because nothing about how the data got in has changed.

Validate at the point of entry

A malformed address caught on the form never becomes a bounce, a segment gap, or a row on next quarter's version of this checklist.

Match before the record is created

A duplicate is cheap to stop and expensive to merge. The moment to catch it is while somebody is still typing.

Fill gaps from what you already hold

Most missing fields are not unknown. They are recorded somewhere else in the business and never made it onto the record.

Flag stale on day 30, not day 300

A follow-up you owe is worth something for weeks and nothing after a year. Automation is just the thing that remembers.

Want to run these checks against your own export, right now? The companion skills pack does the counting for you: it runs locally with Claude Code or any coding agent, reads CSV and multi-sheet Excel files, and uses the same matching rules described here. Your file never leaves your machine.