Build a Family of Similar Checklists Without Starting Over

Last Modified on 09/04/2026 11:05 am EDT

Most of your checklists are one list with a few tasks swapped. Copy the one you already trust, change only what's different, and let each version carry its own trigger.

Introduction

A purchase is a purchase until it's a lead paint purchase, or an HOA purchase, or a commercial one. Same twenty tasks, three of them different. Typed from scratch each time, those twenty shared tasks end up worded twenty different ways, and the day your process changes you get to find every copy of it.

Copying is the way out of that, and Pipeline has a built-in copy on every template. What this article is really about is the part that comes before and after the click: deciding what actually deserves its own template, and making sure each copy points at a different trigger so your team doesn't end up with the same checklist twice.

This is admin work. You'll need to be a master admin, or an admin with the *Fully manage checklists and templates* permission in the template's location.

When to Use This

- You run one purchase checklist, but lead paint deals need extra forms and HOA deals need their own.
- You're about to type the same twenty tasks into a fourth template.
- Your commercial list is your residential list with half of it taken out.
- You copied a template and now the same checklist shows up twice on every transaction.
- You've built up templates for years and can't tell anymore which ones are still genuinely different from each other.

Why This Beats a Copy for Every Case

The instinct is to reach for Copy Template the moment anything differs — one per label, one per office, one per side. It's worth resisting, because every copy is another template to keep in step when the process changes, and because a lot of the copies people make aren't needed at all. A single checklist can be related to several locations and several transaction labels at once, so when the only difference is *where the list lands* rather than *what's on it*, copying is work that buys you nothing.

The copies that do earn their place come with a trap, and it's the same one every time: the shared tasks come along for the ride. One office had its closing-date task sitting on both the Pending Listing checklist and the Pending Buying checklist, so every dual-sided deal collected two of them. The fix was to stop copying that task at all — pull it into one template that applies on any side, and leave each variant carrying only what's genuinely its own.

1. Decide What's Shared and What Varies

Before you copy anything, split your tasks into two piles: the ones every version of this deal needs, and the ones only this version needs. The shared pile belongs in a single template that reaches everything. The variant carries the rest.

Done this way, a change to a shared task is one edit in one place. Done the other way — the whole list copied five times — it's five edits, and you'll miss one.

If the difference you're splitting on is the property type, look at your labels before you build anything. Labels are the thing checklists key off, so the answer to "commercial gets a different list" is often one label away.

Labels classify your transactions by property type, and they drive the checklists Pipeline auto-assigns.

See Transaction Labels

> Already carrying more templates than you can hold in your head? There's no built-in way to compare them, but you can download each one as a CSV and line them up in a spreadsheet — it's the fastest way to find the ones that stopped being different years ago. See Checklist Templates for the download.

2. Copy the Template You Already Trust

Start from your cleanest, most current list rather than whichever one is oldest. Copy Template hands you an exact duplicate, open and ready to rename.

Build a checklist once as a template, then let Pipeline apply it to every transaction that needs it.

See Checklist Templates

> Building the variant from something outside Pipeline instead — a spreadsheet, a list someone emailed you? Skip the copy and use **Bulk Add Tasks** on a new template, which takes the whole list pasted at once.

3. Trim the Copy to Its Variant

Now take out what this version doesn't need and add what only it does. One brokerage keeps a single Purchase Checklist and copies it for lead paint, HOA, and non-lead-paint deals, deleting the forms each one doesn't use and saving each under its own name.

Name them so your team can tell them apart on sight, and give some thought to the first character while you're there: checklists that assign themselves land on a transaction in alphabetical order, so a prefix — *A -*, *B -*, *C -* — puts them in the order you actually work them. Letters beat numbers, which only sort properly as far as 9.

The tasks on a checklist are the documents to collect and the steps to finish on a transaction.

See Document Names or Tasks to Track

4. Point Each Copy at Its Own Trigger

This is the step that bites, and it's the reason a family of checklists goes wrong more often than it goes right. A copy arrives carrying the original's automatic-assignment rule. Leave it alone and you now have two templates waiting for the same transactions, both of which will apply themselves — which is exactly how the same checklist ends up on a file twice.

So open the copy and set its rule to whatever makes this variant different. A template takes one rule, and that constraint is the whole reason the family exists: one template per rule, each pointed somewhere the others aren't.

> Fixing the template doesn't clean up what already landed. Duplicate checklists sitting on transactions have to be removed from those transactions — hover over the checklist's title and click the []. See Checklists Panel.