

BUILDER · 5 MINUTE READ

Subagents: handing a piece of work to a helper

A main agent can hand one focused task to a helper agent with a fresh, clean context, which returns only a short result for checking.

What it is

An AI agent is an assistant that carries out a task in steps and can use tools (see guide 17). A subagent is a helper agent that the main agent starts for one focused piece of work, such as searching a large folder, reviewing a draft or comparing three options.

The helper starts with a fresh, clean context (everything the AI can see at that moment). It does not see the main conversation, only the brief it was given and the tools it was allowed. When it finishes it hands back a short result; everything it read stays with the helper.

Think of a manager who asks an assistant to "go through these 200 invoices and tell me which are over 30 days late". The manager wants the short list, not 200 invoices on their desk.

This differs from a multi-agent environment (see guide 20). There, several agents share the work as a team, each with a standing role. A subagent is temporary: the main agent stays in charge, and the helper exists for one task.

Why it matters

- The main conversation stays uncluttered: long searches and raw files stay with the helper.
- Independent tasks can run in parallel, for example three helpers each checking one supplier.
- A helper can carry specialist instructions, such as a review checklist.
- Limiting a helper's tools limits what can go wrong: a helper that can only read cannot send anything.

How to do it

1. Pick a task that stands alone, with an answer that fits in a few lines.
2. Write a brief as if for a new colleague: the goal, where to look and what you already know.
3. Define the output: format, length and what to do when unsure ("say so, do not guess").
4. Limit the tools to what the task needs, for example read-only access to one folder.
5. Ask the helper to list its sources, so the result can be checked.
6. Check the result before the main agent builds on it.
7. Save briefs that work, so you can reuse them (see guide 11).

Example

Before: you ask your agent to "find every policy document that mentions travel expenses and update our summary". It opens forty documents in the main conversation and starts losing track of your earlier instructions.

After: the main agent hands the search to a helper with this brief:

Task: Find every document in [folder name] that mentions travel expenses.
Background: We are updating our one-page staff summary of the travel policy.
You may: read files in that folder only. Do not edit, move or send anything.
Return: a list of at most 15 lines, each with the file name, the date on the document and one sentence on what it says about travel expenses.
If two documents disagree, say so. If you find nothing, say so.

The helper returns twelve lines and the main agent drafts the summary from them. You open two or three listed files to confirm the helper read them correctly, then approve the summary.

Common mistakes

- **Assuming the helper knows what you know.** It sees only the brief, so write down the background it needs.
- **Asking for "everything you found".** A long result clutters the main conversation again. Ask for a short format.
- **Trusting the summary without looking.** The helper can miss or misread things. Spot-check its sources.
- **Forgetting cost.** Each helper uses extra AI capacity, which may be paid per use. Skip helpers for one-step tasks.
- **Giving helpers more tools than needed.** A search helper does not need permission to send email (see guide 26).

Try it this week

1. Pick a question that needs reading several documents, using made-up or approved material only (3 minutes).
2. Write a brief using the steps above: goal, background, allowed tools, output format (5 minutes).
3. Open a new conversation in your approved assistant, paste the brief and the documents, and read the result (5 minutes).
4. Check two facts against the documents and note how to improve the brief (2 minutes).

A new conversation has a clean context, so it works as a test even without a subagent feature.

Going further

- Agent tools around Claude, ChatGPT, Microsoft Copilot and Gemini may let you define named subagents with their own instructions and tools; describe each clearly so the main agent knows when to use it.
- Run helpers in parallel only for independent tasks.
- Test changes to a helper's brief before relying on them (see guide 27).
- For hand-offs between agents that each keep working, see guide 22.

Key words

- **Subagent:** a helper agent started by a main agent for one focused task, which returns a short result.
- **Context:** everything the AI can see at one moment.
- **Brief:** the written instructions a helper receives; it is all the helper knows.
- **Parallel work:** several helpers working at the same time on separate tasks.
- **Tool limits:** the list of actions a helper is allowed to take.