PLUGINS

The plugin install cheat sheet

Free guide. No signup. 2 min read

Checked 2026-10-27 against code.claude.com/docs/en/discover-plugins, code.claude.com/docs/en/plugins/security and the plugin files in github.com/anthropics/claude-plugins-official.

From @learningwithcole

Three official plugins worth installing before anything else. All three live in `claude-plugins-official`, Anthropic's own marketplace, which Claude Code adds for you the first time you start it in your terminal.

1. The three install commands

Run these inside a Claude Code session:

Text

```
/plugin install frontend-design@claude-plugins-official
/plugin install code-review@claude-plugins-official
/plugin install commit-commands@claude-plugins-official
```

In a session, each command opens the plugin's details first instead of installing right away. Read the pane, then pick a scope:

- **Install for you**: every project on this machine
- **Install for all collaborators on this repository**: everyone who works in this repo
- **Install for you, in this repo only**: just you, just here

Prefer your shell? Same thing without opening a session (user scope by default):

Terminal

```
claude plugin install frontend-design@claude-plugins-official
```

To browse instead, run `/plugin` on its own. It opens on the Discover tab.

2. What each one does

frontend-design A skill Claude uses on its own whenever you ask for frontend work. It pushes Claude toward bold, distinctive design choices (type, colour, layout, motion) instead of the generic look AI pages usually have. Nothing to type: just ask for a page or component.

code-review Reviews a GitHub pull request with a team of agents running in parallel. Per the plugin's command file, five agents check: your CLAUDE.md rules, obvious bugs in the change, the git history of the touched code, comments on past PRs to those files, and code comments in the changed files. Each issue gets a confidence score, anything under 80 is dropped, and the rest is posted as a comment on the PR. It skips closed, draft and already reviewed PRs. Needs the GitHub CLI (`gh`) installed and signed in.

commit-commands Git in one command. It adds:

- `/commit-commands:commit`: drafts a message in your repo's style, stages the files and commits
- `/commit-commands:commit-push-pr`: makes a branch if you're on main, commits, pushes and opens the pull request with `gh pr create`
- `/commit-commands:clean_gone`: cleans up local branches whose remote branch is gone

After installing, type `/` and look under the plugin's name to see its commands.

3. Check the context cost

Every plugin you add costs context, and some cost it on every message.

- Open a plugin with its full install command (as above) or from the **Marketplaces** tab in `/plugin`. For official plugins the details pane shows **Context cost** with two numbers: **Every turn** (added to each message you send) and **When invoked** (added once Claude loads its skills or agents).
- The details pane you reach from the Discover list doesn't show those numbers, so use the install command to see them.
- Already installed? Run `claude plugin details <name>` in your shell. The `Always-on` line is what it adds to every session.

4. Turn off what you don't use

- In a session: `/plugin`, press **Tab** to reach **Installed**, select the plugin and press **Space** to disable it. Or run `/plugin disable`.
- In your shell: `claude plugin disable code-review@claude-plugins-official` (and `claude plugin enable …` to turn it back on).
- The Installed tab groups plugins you haven't used lately under **Not used recently**. Start there.

Good to know

- A plugin runs with your user privileges. Hooks and servers it brings can run code on your machine, so read the details pane before installing anything, official or not.
- Official marketplace plugins update automatically. After an update you'll see a prompt to run `/reload-plugins`.