

Four Claude Code Plugins Guide

Install Ponytail, OmniRoute, Graphify, and Agent Skills and cut your Claude Code token usage, with every command and config file.

Why I run these four

Vanilla Claude Code wastes tokens in four predictable ways. It writes more code than the task needs. It stops working the moment you hit a usage limit on one provider. It rereads the same files every session because it does not remember your project. And it improvises long, branching conversations instead of following a tested procedure. Each of these four plugins closes one of those gaps, and they are all free and open source.

Ponytail makes the agent write less code. OmniRoute reroutes you to free providers when a limit hits. Graphify builds a map of your codebase so the agent stops rereading files. Agent Skills gives it 25 tested workflows for the full build lifecycle.

You do not need all four to see a difference. If you only install one, install Ponytail. It is two commands and it changes the next prompt you send.

01

How it actually works

1. Ponytail loads a rule set that runs before the agent writes code. It walks a decision ladder: does this need to exist, is it already in the codebase, does the standard library or platform cover it, is it a one-liner. The agent only writes new code after all of those come back no. Less generated code means fewer output tokens now and fewer input tokens every time that file gets reread.
2. OmniRoute runs as a local gateway on your machine. Claude Code talks to it instead of talking to one provider directly. When a provider's free quota runs out or a request fails, OmniRoute falls back to the next provider in line, so your session keeps going instead of stopping at a limit.
3. Graphify parses your code locally with tree-sitter and stores the structure as a knowledge graph. When Claude Code needs to find where something lives, it queries the graph instead of opening ten files to work it out. The code parsing never leaves your machine and costs no API credits.

02

The one thing almost everyone gets wrong

People install OmniRoute and expect Claude Code to keep using Claude for everything, just with more headroom. That is not what happens.

WATCH OUT

When OmniRoute falls back, it routes you to a different model from a different provider. That model may be smaller or slower than Claude. For throwaway work like scaffolding, tests, or a quick rename, that is fine. For hard reasoning, it is worth stopping and waiting for your Claude limit to reset. Treat OmniRoute as a way to keep moving on easy work, not a free replacement for the model you actually want.

03

The plugins and the commands

SOURCE CODE

Ponytail, by DietrichGebert. Install marketplace and plugin from this repo:

<https://github.com/DietrichGebert/ponytail>

SOURCE CODE

OmniRoute, by diegosouzapw. Installed from npm, source here:

<https://github.com/diegosouzapw/OmniRoute>

SOURCE CODE

Graphify, by Graphify Labs. PyPI package name is graphifyy with a double y:

<https://github.com/Graphify-Labs/graphify>

SOURCE CODE

Agent Skills, by Addy Osmani with Federico Bartoli and Joan Leon:

<https://github.com/addyosmani/agent-skills>

INSTALL COMMANDS (RUN INSIDE CLAUDE CODE)

```
/plugin marketplace add DietrichGebert/ponytail
/plugin install ponytail@ponytail

/plugin marketplace add addyosmani/agent-skills
/plugin install agent-skills@addy-agent-skills
```

INSTALL COMMANDS (RUN IN YOUR TERMINAL)

```
# OmniRoute
npm install -g omniroute
omniroute # starts the gateway at http://localhost:20128/v1
omniroute configure claude-code

# Graphify
uv tool install graphifyy
graphify install
graphify claude install # makes Claude Code check the graph first
```

~/.CONFIG/PONYTAIL/CONFIG.JSON

```
{
  "defaultMode": "full"
}
```

.GRAPHIFYIGNORE (IN YOUR PROJECT ROOT)

```
node_modules/
dist/
build/
.venv/
*.generated.*
```

Set it up in Claude Code or Codex

1. Check your tooling. Run `node --version` and `npm --version`. Ponytail and OmniRoute both need Node.js on your PATH. If you plan to use Graphify, install `uv` as well.
2. Get the source if you want to read or fork it. `git clone https://github.com/DietrichGebert/ponytail`, `git clone https://github.com/diegousouzapw/OmniRoute`, `git clone https://github.com/Graphify-Labs/graphify`, `git clone https://github.com/addyosmani/agent-skills`. You do not need to clone these to use them through the marketplace or npm, but the repos are the source of truth for options.
3. Install the two Claude Code plugins. Open Claude Code in any project and run the four marketplace and install lines from the code block above. Restart Claude Code once so the slash commands register.
4. Codex users: Ponytail also ships a Codex path. Run `codex plugin marketplace add` for the same repo, then `codex plugin add ponytail@ponytail`. For Agent Skills on Codex, clone the repo into your project or into `.codex/skills/` and add a pointer line in `AGENTS.md`. The folder name must match the name field in each `SKILL.md`.
5. Install OmniRoute. Run `npm install -g omniroute`. Start it with `omniroute` in its own terminal tab and leave it running. Then run `omniroute configure claude-code` so Claude Code sends requests through it. The auto model works with no API key.
6. Install Graphify. Run `uv tool install graphify`, then `graphify install`, then `graphify claude install`. Open your project and run `/graphify .` to build the first graph. Re-run `/graphify . --update` after big changes.
7. First test. In a real project, ask Claude Code to add a small feature and tell it to follow Ponytail. Then run `/ponytail-gain` to see the benchmark scorecard. A successful setup shows the slash commands autocompleting and the graph query returning file paths instead of the agent opening files blind.

04

The commands I actually run

START A FEATURE THE LAZY WAY

Add [FEATURE] to [FILE]. Follow Ponytail: reuse what is already in the codebase, use the standard library or a platform feature if one fits, and keep it to the smallest change that works.

CHECK THE DIFF BEFORE I COMMIT

Run `/ponytail-review` on the current diff. Tell me which parts are over-built and cut anything that is not doing real work.

USE THE MAP INSTEAD OF READING FILES

Before you open any files, run `/graphify query "where is [THING] handled?"` and start from what the graph returns.

KEEP MOVING AFTER A LIMIT

I just hit my usage limit. Switch to the omniroute auto model and keep going on [TASK]. Flag anything that needs the stronger model so I can redo it later.

RUN A LIFECYCLE SKILL

/plan [FEATURE], then /build with per-task verification. Stop and show me the plan before you start building.

When it breaks

ERROR	FIX
/plugin marketplace add fails	Update Claude Code to the latest version and retry. The plugin marketplace needs a recent build.
Ponytail slash commands do nothing	Node.js is not on your PATH in this shell. Run <code>node --version</code> , fix the PATH, restart Claude Code.
omniroute: command not found	The global npm bin folder is not on your PATH. Run <code>npm config get prefix</code> and add its bin subfolder to your shell profile.
OmniRoute auto model returns errors	The current free provider is rate limited. Wait a minute, or pick a different routing strategy in the OmniRoute dashboard.
graphify not found after install	The uv tool bin is not on your PATH. Run <code>uv tool update-shell</code> and open a new terminal.
/graphify . is slow or the graph is huge	Add build and dependency folders to <code>.graphifyignore</code> (<code>node_modules/</code> , <code>dist/</code> , <code>.venv/</code>) and rerun.
Claude Code ignores the graph	Run <code>graphify claude install</code> once, then restart Claude Code so it loads the config.
Agent Skills commands missing	Check the install id: <code>/plugin install agent-skills@addy-agent-skills</code> . The marketplace id has the addy- prefix.

Ready to go deeper?

This playbook is the short version. Inside the hub you get the full build walkthroughs, the prompt library, and the systems I use to ship this in a weekend.

Join the Build →

<https://hub.digicuratoragency.com/join>