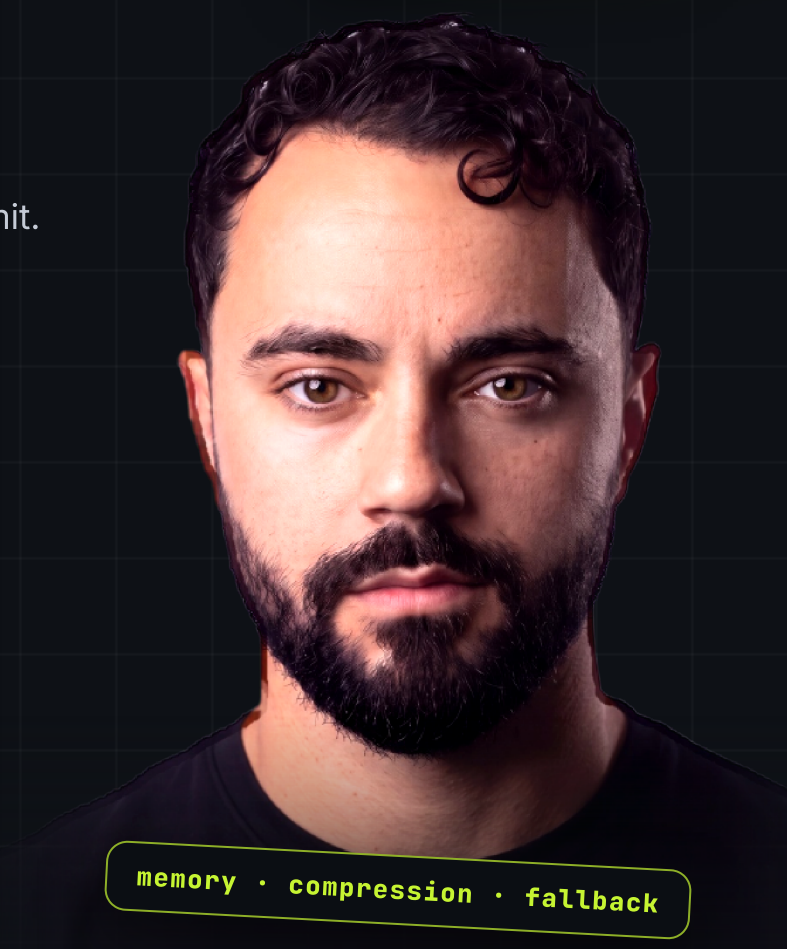
// FREE GUIDE BY @AVIGRONDIN

5 Claude Code Plugins

SETUP GUIDE

/plugin install



memory · compression · fallback

Use fewer tokens, keep memory between sessions, and keep coding when you hit a limit.

FREE | OPEN SOURCE | 5 TOOLS

You commented TOKENS. Here is every link plus the setup: what each tool does, the exact install steps from its own docs, how to check it works, and what to watch out for. I didn't build any of them. Each one is credited to the people who did.

// 02 · WHAT'S INSIDE

Five add-ons. Five different jobs.

Each one fixes a different leak: a session that forgets, a prompt stuffed with logs, a limit that ends your day, a setup you never tuned, and skills that never get better.

#	TOOL	WHAT IT DOES	WHAT IT IS	INSTALL
01	**OmniRoute**	Keep coding when a limit hits. One local address, hundreds of AI providers, and it moves to the next model when one runs out.	**AI gateway**	`npm`
02	**Claude-Mem**	Remembers what happened in past sessions and loads the useful parts into new ones.	**Claude Code plugin**	`/plugin or npx`
03	**Headroom**	Shrinks tool output, logs and files on your computer before they reach the model.	**Local proxy**	`uv or pip`
04	**Claude Code Setup**	Scans your project and recommends hooks, skills, MCP servers and subagents. Read-only.	**Official Anthropic plugin**	`/plugin`
05	**Task Observer**	Logs your corrections and proposes skill improvements. You approve every change.	**Skill**	`npx skills`

WHO IT'S FOR

Daily users

You hit your usage limit in the middle of real work.

Long sessions

Big repos, big logs, a context window that fills up fast.

Skill builders

You wrote skills once and they never improved.

New to Claude Code

You don't know which hooks or MCP servers to set up.

HONEST NOTE

Only two of the five install from Claude Code's `/plugin` menu (Claude-Mem and Claude Code Setup). OmniRoute and Headroom run next to Claude Code on your computer, and Task Observer is a skill. All five work with Claude Code. Each page uses the install route its own docs give.

// 03 · WHAT YOU NEED

Claude Code first. Then the /plugin menu.

Everything here runs in Claude Code, Anthropic's coding agent, in your terminal or the desktop app. The regular chat on claude.ai can't run any of it.

1 · INSTALL CLAUDE CODE

Mac or Linux, in your terminal:

```
curl -fsSL https://claude.ai/install.sh | bash
```

Windows, in PowerShell:

```
irm https://claude.ai/install.ps1 | iex
```

Then open a project folder and type `claude`. Full steps: code.claude.com/docs/en/setup

2 · ALSO ON YOUR COMPUTER

- **Node.js 22 or 24** for OmniRoute (22.22.2 or newer). Claude-Mem needs 20+.
- **uv and Python 3.13** for Headroom. uv can fetch Python for you.
- **An account with an AI provider** for OmniRoute. You bring your own.

Get them at nodejs.org and docs.astral.sh/uv

3 · HOW PLUGINS WORK

`/plugin`	Opens the plugin menu: Discover, Installed and Marketplaces tabs.
`claude-plugins-official`	Anthropic's marketplace. Claude Code adds it for you the first time you start it.
`/plugin marketplace add owner/repo`	Adds someone else's marketplace, straight from their GitHub repo.
`/plugin install name@marketplace`	Opens the plugin's details so you can review it and pick a scope.
`/reload-plugins`	Loads a plugin change into the session you're already in.
`claude plugin list`	In your shell: everything installed, with its scope.

USER SCOPE
Just you, every project
Saved in ~/.claude/settings.json.

PROJECT SCOPE
Everyone in this repo
Saved in .claude/settings.json, which you commit.

LOCAL SCOPE
Just you, this repo
Saved in .claude/settings.local.json.

// 04 · TOOL 1 · OMNIROUTE

Hit a limit? Switch models, keep going.

OmniRoute is a free AI gateway that runs on your computer. You connect your own AI provider accounts, point Claude Code at one local address, and when a model hits its limit, it moves to the next one.

TYPE	RUNS	NEEDS	LICENCE	REPO
AI gateway	localhost:20128	Node.js 22 or 24	MIT	OmniRoute →

INSTALL IT

01 Install

```
npm install -g omniroute
```

From the repo's Quick Start. Peer-dependency warnings during install are harmless, per the README.

02 Start it

```
omniroute
```

The dashboard opens at localhost:20128. The API lives at localhost:20128/v1.

03 Connect a provider

Dashboard → Providers. Add an API key or sign in to an account you own, then test the connection.

04 Make a key

Dashboard → API Keys. This key lets your tools talk to OmniRoute, not to the providers. Save it: it only shows once.

CHECK IT WORKS

```
curl http://localhost:20128/v1/models -H "Authorization: Bearer YOUR_KEY"
```

You should see the models from the providers you connected. No models? Connect a provider first (step 3).

CATALOG
375+ providers
OpenAI, Anthropic, Gemini, DeepSeek, Groq and hundreds more.

FREE TIERS
150+ marked free
As the repo counts them, Nov 3, 2026. Each one has its own limits.

MODEL
"auto"
Lets OmniRoute pick a model from your connected providers.

// 05 · TOOL 1 · OMNIROUTE

Point Claude Code at OmniRoute.

The easy way writes no config: OmniRoute starts Claude Code for you with the right settings. Keep OmniRoute running in another terminal.

`omniroute launch`	Starts Claude Code against your local OmniRoute. Nothing is written to disk.
`omniroute setup-claude`	Writes one Claude Code profile per model it finds, in ~/.claude/profiles/.
`omniroute launch --profile glm52`	Starts Claude Code with one of those profiles (glm52 is the docs' example name).
`omniroute doctor`	Checks your providers, ports and setup when something's off.

THE MODEL NAMES TO KNOW

`auto`	Balanced default. Sticks with the last provider that worked.
`auto/coding`	Quality first, weighted for writing code.
`auto/fast`	Lowest latency first.
`auto/cheap`	Cheapest per token first.
`auto/subscription`	Only quota you already pay for. Never a billable fallback.

To confirm Claude Code goes through OmniRoute, watch each request in Dashboard → Monitoring / Logs.

GOOD TO KNOW

- **It isn't Claude anymore.** Claude Code runs on the model OmniRoute picks. Quality changes with it.
- **Bring your own accounts.** Each free tier has its provider's limits and terms. Some need a card.
- **Not every provider is in auto.** Ones marked `tos: avoid` are left out.

IF IT BREAKS

- **npm 11 blocks the install?** Use the `--allow-scripts` line in the README.
- **Setting it by hand?** Base URL `http://localhost:20128`, no /v1.
- **Full guide:** CLAUDE-CODE-CONFIGURATION.md

// 06 · TOOL 2 · CLAUDE-MEM

Claude forgets. This remembers.

Claude-Mem records what Claude does in each session, compresses it into short notes, and loads the useful parts into your next one. It runs in the background while you work.

TYPE	STORES	NEEDS	LICENCE	REPO
Claude Code plugin	~/.claude-mem/	Node.js 20+	Apache-2.0	claude-mem →

OPTION A · INSIDE CLAUDE CODE

```
/plugin marketplace add
thedotmack/claude-mem
/plugin install claude-mem
```

Then restart Claude Code.

OPTION B · IN YOUR TERMINAL

```
npx claude-mem install
```

Its docs say to run this once even after option A: it puts the launcher the hooks need in place, so your first session isn't missing memory.

THE SIGN-IN QUESTION

The npx installer asks you to sign in and pre-selects CMEM Pro, their hosted memory (30-day free trial). You don't have to take it. To skip sign-in and use your Claude plan, run `npx claude-mem install --provider claude`. Your own OpenRouter or Gemini key also works.

THEN JUST ASK

- `What bugs did we fix last session?`
- `How did we implement authentication?`
- `Show me recent work on this project`

CHECK IT WORKS

- **Open a new session.** Notes from earlier sessions show up at the start.
- **Run** `npx claude-mem doctor` to check the hook launcher and your Claude Code version.

GOOD TO KNOW

- **Not** `npm install -g claude-mem`. That installs the library only. No hooks, no memory.
- **Needs Claude Code 2.1.139+.** Older versions capture nothing. Run `claude update`.
- **Keep secrets out.** Wrap them in `<private>` tags and they're not stored.
- **Memory costs tokens.** On the Claude plan option, writing the notes uses your plan.
- **Crypto token?** The README mentions one (CMEM). You don't need it to use the plugin.

// 07 · TOOL 3 · HEADROOM

Shrink what Claude reads before you pay for it.

Headroom sits between Claude Code and the model and compresses tool output, logs, search results and files, on your computer. In its own demo a 10,144-token log came down to 1,260 tokens and the error line was still found.

TYPE	RUNS	NEEDS	LICENCE	REPO
Local proxy	localhost:8787	uv + Python 3.13	Apache-2.0	headroom →

INSTALL AND USE IT

01 Install the CLI

```
uv tool install --python 3.13 "headroom-ai[all]"
```

Puts the `headroom` command on your computer in its own environment.

02 Start Claude Code with it

```
headroom wrap claude
```

Starts the local proxy and launches Claude Code through it. Start Claude Code this way each time.

03 Watch the savings

```
headroom dashboard
```

Opens localhost:8787/dashboard while the proxy is running.

CHECK IT WORKS

- `headroom doctor` health check. Confirms routing works.
- `headroom savings` your own number, from your own traffic.
- **Their benchmark:** 21% to 57% fewer tokens across four tool-heavy tests. Prose barely shrinks.

GOOD TO KNOW

- **Install `headroom-ai`.** The PyPI package called `headroom` is a different project.
- **The npm package has no CLI.** It's a code library only.
- `wrap` **also adds Serena,** a code-navigation MCP server. Skip it with `--code-memory none`.
- **Undo it any time:** `headroom unwrap claude`

WHERE IT PAYS OFF

Long sessions with heavy tool output: test runs, logs, big JSON, search results. Short chats see little or no change. Docs: docs.headroomlabs.ai/docs

// 08 · TOOL 4 · CLAUDE CODE SETUP

Not sure what to set up? Ask the official plugin.

Claude Code Setup is Anthropic's own plugin. It scans your project and recommends the top one or two automations in each category. It's read-only: it suggests, it never changes your files.

TYPE	BY	CHANGES FILES	LICENCE	REPO
Official plugin	Anthropic	Never	Apache-2.0	claude-code-setup →

TYPE THIS IN CLAUDE CODE · INSTALL

```
/plugin install claude-code-setup@claude-plugins-official
```

Or open `/plugin`, go to Discover and search for it. Review the details, pick a scope, done. No marketplace listed? Run `claude plugin marketplace add anthropics/claude-plugins-official` in your shell first.

WHAT IT RECOMMENDS

MCP servers
Outside tools, like context7 for docs or Playwright for front ends.

Skills
Packaged know-how for jobs you repeat.

Hooks
Auto-format, auto-lint, block edits to sensitive files.

Subagents
Reviewers for security, performance, accessibility.

Slash commands
Quick workflows like /test or /pr-review.

SAY THIS IN CLAUDE CODE

- `recommend automations for this project`
- `help me set up Claude Code`
- `what hooks should I use?`

MY TIPS

- **Run it first in every new project.** Then ask Claude to build only the ones you'll really use.
- **Ask for one type** and it goes deeper: 3 to 5 picks instead of 1 or 2.
- **Check it's there:** `/plugin` → Installed, or `claude plugin list`.

WHAT IT READS

Your project files, with Claude's Read, Glob, Grep and Bash tools. It may also search the web for tools that fit your stack. It writes nothing. Building what it recommends is a separate ask.

// 09 · TOOL 5 · TASK OBSERVER

Your skills learn from your corrections.

Task Observer is a meta-skill. It watches your sessions, logs each time you correct Claude, and turns those notes into suggested skill updates and new skill ideas. You review them.

TYPE	BY	INSTALLS WITH	LICENCE	REPO
Skill	Eoghan Henn	Vercel's skills CLI	CC BY 4.0	one-skill-to-rule-them-all →

INSTALL AND TURN IT ON

01 Install it

```
npx skills add rebelytics/one-skill-to-rule-them-all --skill task-observer -g
```

Needs Node.js. The `-g` installs it for all your projects.

02 Activate it

Add the activation block from references/environments.md to your CLAUDE.md. Installing alone isn't enough: the README says it under-triggers without it.

03 Test in a NEW session

Start with normal work that doesn't mention it. After a few sessions, a `skill-observations/observation-log/` folder should exist. No folder means it never turned on.

THE HONEST VERSION

You may have heard it "rewrites your other skills". It doesn't do that on its own. It proposes changes, you approve each one in a review, and updates wait in `skill-updates/` until you install them.

SAY THIS IN CLAUDE CODE

```
Use the task-observer skill and help me add its activation block to my CLAUDE.md
```

```
Any observations logged?
```

```
Review my open observations and show me the proposed skill updates
```

GOOD TO KNOW

- **It costs tokens every session.** It loads its own instructions at the start of each one.
- **Ask at the end of a session.** "Any observations logged?" is the habit the author recommends.
- **Best with many skills.** With a handful, editing them by hand may be enough.

// 10 · THE FULL STACK

The full stack, in the right order.

Start with the safe one. Add the two that sit in the middle of your traffic last, and run one of them at a time. (The order is my suggestion, not the projects'.)

01	**Claude Code Setup**	Official and read-only. See what your project needs before you add anything.
02	**Claude-Mem**	Memory starts building from your next session.
03	**Task Observer**	Starts logging corrections now, so your first review has something in it.
04	**Headroom**	Add it when your sessions get long and tool-heavy.
05	**OmniRoute**	Add it when you keep hitting limits and want other models to fall back to.

ONE PROXY PER SESSION

STAYING ON YOUR CLAUDE PLAN

```
headroom wrap claude
```

Claude models, smaller prompts.

USING OTHER PROVIDERS

```
omniroute launch
```

Other models, with fallback. OmniRoute has its own compression engines built in (one is inspired by Headroom).

WHY NOT BOTH?

Both work by changing where Claude Code sends its requests. Neither project's docs cover chaining the two, so pick one per session.

YOUR DAILY LOOP

01 Start
with `headroom wrap claude` or `omniroute launch`.

→

02 Work
Claude-Mem and Task Observer run in the background.

→

03 Ask
"Any observations logged?" before you close.

→

04 Review
the proposed skill updates once a week.

// 11 · TROUBLESHOOTING

When it breaks, check these first.

These fixes come from each project's own troubleshooting docs. Not listed here? Tell Claude what you see and paste the error.

"memory hooks are not active"	Claude-Mem: run `npx claude-mem install`, restart your terminal, then Claude Code.
Claude-Mem saves nothing	Check `claude --version`. It needs 2.1.139 or later: `claude update`.
headroom: command not found	Run `uv tool update-shell`, then open a new terminal.
Context fills up faster on a proxy	`headroom wrap claude` sets `ENABLE_TOOL_SEARCH` for you. Running `claude` by hand? Set it yourself.
/remote-control is missing	Claude Code only offers it when talking to api.anthropic.com. Use plain `claude` for those sessions.
OmniRoute's auto returns nothing	No eligible provider yet. Connect one in Dashboard → Providers and test it.
Plugin installed, not working	Run `/reload-plugins`, or restart Claude Code.
Task Observer never logs	The activation block is missing from CLAUDE.md. Add it, then test in a new session.

TO REMOVE ONE

`/plugin uninstall`	Claude-Mem or Claude Code Setup. Or use the Installed tab in `/plugin`.
`npx claude-mem uninstall`	Removes Claude-Mem's hook launcher too.
`headroom unwrap claude`	Puts Claude Code back the way it was before Headroom.
`omniroute launch`	Writes nothing. Just start plain `claude` again to stop using it.

MY TIP

Install one tool, check it works, then add the next. If something breaks, you know exactly which one did it.

// 12 · SAFETY

Third-party code. Know what it can see.

Four of the five come from outside Anthropic. Anthropic's own docs say a plugin can run code on your computer with your permissions. Here is what each one touches.

TOOL	WHAT IT SEES	WHAT IT STORES	WHERE DATA GOES
OmniRoute	Every prompt and file Claude Code sends	Your provider keys and request logs, on your computer	Your requests go to the providers you connect
Claude-Mem	Every tool call: files read, edits, commands	Notes in ~/.claude-mem/ on your computer	CMEM Pro: notes compressed through cmem.ai. Claude plan option: one numbers-only summary at signup
Headroom	Everything between Claude Code and the model	Uncompressed originals, cached on your computer	Compression runs on your computer. Its docs say nothing is sent out to be compressed
Claude Code Setup	Your project files, read-only	Nothing	It may search the web for recommendations
Task Observer	Your sessions and your corrections	Logs in skill-observations/, updates in skill-updates/	No service of its own. It runs inside Claude Code

API KEYS (OMNIROUTE)

- **Provider keys go in the dashboard,** never in the chat.
- **Your OmniRoute key** only unlocks OmniRoute. Treat it like a password anyway.
- **Read each provider's terms** before you connect it. Free tiers have rules.

BEFORE YOU INSTALL ANYTHING

- **Read the details pane** `/plugin install` opens it before anything installs.
- **Updates change the code.** Auto-update can replace the files you reviewed.
- **Remove what you don't use.** Page 11 has the commands.

Anthropic's plugin safety page: code.claude.com/docs/en/plugins/security

// 13 · CREDITS

None of these are mine. Credit to the builders.

All five are free, open-source projects built by other people. Here's who made each one, the licence it's under, and where to find it.

TOOL	WHERE	BUILT BY	LICENCE	STARS
OmniRoute	`github.com/cheaper-inference/OmniRoute`	Diego Souza and contributors	MIT	84,000+
Claude-Mem	`github.com/thedotmack/claude-mem`	Alex Newman (@thedotmack)	Apache-2.0	111,000+
Headroom	`github.com/headroomlabs-ai/headroom`	Headroom Contributors	Apache-2.0	78,000+
Claude Code Setup	`anthropics/claude-plugins-official`	Anthropic (Isabella He)	Apache-2.0	45,000+ *
Task Observer	`github.com/rebelytics/one-skill-to-rule-them-all`	Eoghan Henn (rebelytics.com)	CC BY 4.0	3,700+

Stars checked on GitHub, Nov 3, 2026. * Stars for Anthropic's whole plugin directory, not this one plugin.

HOW THIS GUIDE WAS MADE

Every install command, setting and number in this guide comes from each project's own README and docs, plus Anthropic's plugin docs, read on Nov 3, 2026. I did not run these installs for this guide. These projects change fast: if a project's docs say something different from this page, follow the docs.

THE DOCS I USED

- **OmniRoute:** Free Tiers Guide and Claude Code setup
- **Claude-Mem:** docs.claude-mem.ai/installation
- **Headroom:** docs.headroomlabs.ai/docs
- **Plugins:** code.claude.com/docs/en/discover-plugins

SAY THANKS

- **Star the repo** of each one you use. It's free and it helps.
- **Report bugs** on each project's GitHub Issues page, not to me.
- **Task Observer is CC BY 4.0:** share it, but credit Eoghan Henn and link the repo.

// 14 · THE SOFTWARE I USE

My stack, in one place.

The tools behind my businesses. Start with the ones that fit what you build.

HighLevel
CRM, funnels, calendars, email and SMS in one place. What I run my funnels on.
gohighlevel.com →

HighLevel SaaS Pro
The HighLevel upgrade that lets you resell it to clients under your own brand.
gohighlevel.com/saasupgrade →

Close
A sales CRM with calling, email and SMS built in.
close.com →

n8n
Workflow automation that connects your apps, with AI steps.
n8n.io →

Higgsfield
AI image and video generation.
higgsfield.ai →

Arcads
AI video ads with AI actors, UGC style.
arcads.ai →

Foreplay
Save and organise winning ads for creative research.
foreplay.co →

Dripify
LinkedIn outreach on autopilot.
dripify.io →

Webflow
A visual website builder with hosting.
webflow.com →

PandaDoc
Proposals, contracts and e-signatures.
pandadoc.com →

monday.com
Projects, tasks and team workflows.
monday.com →

OnlineJobs.ph
Hire full-time remote team members from the Philippines.
onlinejobs.ph →

Coach Financing
Offer your clients financing for coaching and programs.
coachfinancing.com →

Some of these are affiliate links: if you sign up through them I may earn a commission, at no extra cost to you. I only list tools I use or have used in my businesses.

// 15 · WHAT NEXT

Your Claude Code is set up. I build businesses with AI.

I use AI to build and run lead generation businesses. If you want to do the same, there are two ways to work with me.

OPTION 1 · THE COMMUNITY

Join my Skool community

Pay Per Lead Academy

Stop chasing clients. Start building a lead generation business.

- Pick a niche
- Run Meta and Google ads
- Find lead buyers and price your leads
- Automate delivery and tracking

Join on Skool →

skool.com/pay-per-lead-academy-3593

OPTION 2 · DIRECT

Book a call with me

1 on 1 with Avi Grondin · 60 minutes

One hour. Your business. My full attention.

- A deep dive into your specific business
- Your niche, your ads, your lead buyers
- A straight read on your numbers
- Your next best moves

Book your call →

Tap the button or scan the code

MORE LIKE THIS

Follow @avigrondin

More free AI guides, explained so anyone can run them.

