Quickstart: your Brain in 5 minutes

A Brain is your AI's long-term memory. It remembers what earlier sessions learned and hands it back at the next task. You explain something once, and every AI in every editor knows it.

1. Set up: Claude Code

Type these two commands in Claude Code, one after the other:

```
/plugin marketplace add cachly-dev/cachly-mcp
/plugin install cachly-brain@cachly
```

Then start a new session. The first time, your browser opens once: sign in with GitHub, Google or e-mail. There is no key to copy. Check it in a terminal with `claude mcp list`.

2. Set up: Cursor, Windsurf, VS Code and others

Open a terminal in your project folder and run:

```
npx @cachly-dev/mcp-server@latest autopilot
```

Sign in once (confirm the code in your browser, free, no credit card). autopilot finds the editors in your project, writes their files and fills your Brain from your git history. Restart your editor. You need Node.js 20 or newer. In VS Code you can also use the **cachly Brain** extension from the Marketplace.

3. First test

Ask your AI in a new chat:

```
What do you know about this project?
```

Your AI should use the tool `smart_recall` and answer from your Brain. In a git project, the answer names things from your own commits.

4. The five tools that matter most

- `smart_recall` – Looks up what earlier sessions learned about the task you are starting.
- `learn_from_attempts` – Saves what worked and what did not after a fix.
- `remember_context` – Stores one fact under a name, for example how the project is built.
- `session_start` – Shows at the start of a session what happened last time.
- `session_end` – Records at the end what was done in this session.

You do not have to call them yourself: the rule files tell your AI when. You can also ask directly: “Remember that deploys go through make deploy.”

Help

- Step-by-step guide: cachly.dev/docs/setup
- Troubleshooting in your project folder: `npx @cachly-dev/mcp-server@latest doctor`
- Mail: support@cachly.dev