

COMPANION TO MODULE 01 · PERSONAL KNOWLEDGE DATABASE

# Set up your **Second Brain** in any AI you already have

Click-by-click for six tools across three paths. The Module 01 prompt works in any AI tool with file context – only the setup steps differ. If your tool isn't listed, find the closest one; the workflow is identical, only the labels shift.

## THE THREE PATHS

### A On your computer

Agent reads your folder and writes the three context files plus a CLAUDE.md / AGENTS.md wrapper at the project root. Future sessions auto-load.

Claude Code · GitHub Copilot Agent · Cursor · Windsurf · Codex CLI · Gemini CLI · Aider

### B In a cloud Project

Upload your docs to a Project once; chat against them across many sessions. You save the three files yourself.

Claude Projects · ChatGPT Projects · Gemini Gems · Custom GPTs · Notebook LM

### C In any chat

Attach files at session start, run the prompt, save outputs. Works everywhere; you'll re-attach next session.

Microsoft 365 Copilot · ChatGPT · Claude.ai · Gemini · Perplexity

## BEFORE YOU START

- 1. Files.** 3-5 documents from one project – briefs, notes, decision threads, schemas. (Detail in Module 01.)
- 2. An AI tool with file context.** Pick A, B, or C above based on what your tool can do.
- 3. ~10 minutes.** Setup is one-time. Daily load is < 60 seconds after that.

Full prompt + module page // [pickbits.ai/pkd](https://pickbits.ai/pkd)

PICKBITS.AI // SETUP COMPANION // 1 / 5

## A On your computer

The agent runs the silent inventory, asks 3-5 questions, then writes the three context files into `context-stack/` plus a thin `CLAUDE.md` + `AGENTS.md` wrapper at the project root. The wrapper auto-loads the context files at every future session – daily load becomes *"open and ask where I left off."*

### Claude Code

1. Open a terminal in the folder with your project source docs.
2. Run `claude`.
3. Paste the prompt (from [pickbits.ai/pkd](https://pickbits.ai/pkd)).
4. Answer the 3-5 questions.
5. Approve writes – limited to `context-stack/` plus the wrappers (`CLAUDE.md`, `AGENTS.md`) at the root.
6. The debrief lists what was built. Future sessions auto-load via `CLAUDE.md`.

**Also works in:** *Codex CLI, Aider, Gemini CLI – any terminal agent in a workspace.*

### GitHub Copilot Agent

1. Open your project folder in VS Code.
2. Open Copilot Chat. Switch the mode dropdown from *Ask* to *Agent*.
3. Paste the prompt.
4. Approve tool calls when prompted (`readFile`, `writeFile`).
5. Answer the 3-5 questions.
6. Files appear in `context-stack/`. The wrapper at `AGENTS.md` auto-loads them next time – Copilot reads it as the cross-tool agent-config convention.

**Also works in:** *Cursor, Windsurf, Cline, Continue – any IDE-embedded agent with workspace write.*

**Daily load on Path A:** open a fresh chat in the same folder. Ask *"where did I leave off?"* The wrapper tells the agent which context files to load. ~30 seconds, no manual references needed.

## B In a cloud Project

Drop your 3-5 source docs into a Project once and they're attached to every chat in that space – closest thing to "wakes up knowing" without folder write. The agent outputs the three context files as *markdown blocks*; you save them locally (and optionally re-upload to the Project so future chats inherit them).

### Claude Projects

1. Sign in at [claude.ai](https://claude.ai). Click *Projects* in the sidebar.
2. Click *Create project*. Name it (a short slug, e.g. **feedrunner**).
3. Open *Project knowledge*; upload your 3-5 source docs.
4. Open a new chat *inside the project*.
5. Paste the prompt. Answer the 3-5 questions.
6. Save the three returned markdown blocks to a local **context-stack/**. Re-upload them to the Project so future chats inherit them.

**Also works in:** *Notebook LM*, any "create a workspace, attach files, chat" surface.

### ChatGPT Projects

1. Sign in at [chatgpt.com](https://chatgpt.com). Click *+ New Project* (Plus / Team / Enterprise).
2. Name the project. Open *Files*; upload your 3-5 source docs.
3. (Optional) add custom instructions: "Treat attached docs as my Context Stack source."
4. Open a new chat *inside the project*.
5. Paste the prompt. Answer the 3-5 questions.
6. Save the three markdown blocks to a local **context-stack/**. Re-upload them to the Project so future chats inherit them.

**Also works in:** *Gemini Gems*, *Custom GPTs*, *Microsoft 365 Copilot Agents*.

**Daily load on Path B:** open a chat in the project (the three context files are already attached). Ask "*load identity.md + the current project-\*.md + state.md, tell me where I left off.*" Update **state.md** at session end and re-upload.

## C In any chat

Works anywhere, but you'll feel the friction by week two – every new chat means re-attaching files. The fix is to graduate to Path B once you know the workflow. Until then, this works fine; just budget 90 seconds for daily load instead of 60.

### Microsoft 365 Copilot

1. Open Copilot – copilot.cloud.microsoft, the Edge sidebar, or in-app (Word, Outlook, Teams).
2. Toggle to *Work* scope so it can see your tenant's SharePoint / OneDrive.
3. Either reference docs by name in the prompt ("read the FeedReader brief in OneDrive/Projects/...") or attach them with the paperclip.
4. Paste the Module 01 prompt. Answer the 3-5 questions.
5. Copy each of the three markdown blocks. Save into a Word doc each, or as a local `.md` file.
6. Save to OneDrive in `context-stack/` – keeps them portable across devices.

**Also works in:** Copilot in Word, Outlook, or Teams; Copilot Pages – UI labels shift, the workflow is identical.

### ChatGPT (no project)

1. Sign in at chatgpt.com. Start a new chat.
2. Click the paperclip. Upload your 3-5 source docs.
3. Paste the prompt. Answer the 3-5 questions.
4. Copy the three markdown blocks. Save to a local `context-stack/` folder.
5. Next session: re-attach the three context files *and* any updated source docs.
6. When tired of re-attaching, graduate to Path B (ChatGPT Projects on Plus / Team / Enterprise).

**Also works in:** Gemini app, Claude.ai chat, Perplexity Pro – any chat surface with file upload.

**Daily load on Path C:** attach `identity.md` + the relevant `project-*.md` + `state.md` to a fresh chat, ask "load these, tell me where I left off;" –90 seconds – friction is the cost of universality.

## THE PROMPT – ANNOTATED DIGEST

context-stack-builder.md

Full prompt at [pickbits.ai/pkd](https://pickbits.ai/pkd)

"Act as my Context Stack builder."	One role, one job. Keeps the AI from drifting to generic-assistant mode.
"Read only the folder I share. Write only to context-stack/ and the Path A wrappers."	Safety contract. Scope-limited – no surprise writes elsewhere on your machine.
"Skip credentials, .env, anything secret. Note skips in debrief."	Visible omissions. You see what didn't go in, not just what did.
"Scan silently. Then ask 3–5 short questions in one message."	Inventory before noise; clarification over fabrication.
"Mark uncertain spots [TBD]. Debrief what you skipped."	Visible gaps, not invented confidence. You can trust what's there.

## THE DRILL – SETUP IS ONE-TIME, THESE TWO ARE FOREVER

**Daily Load**

~30s (A) · 60–90s (B/C)

1. Open your AI. On Path A, the wrapper auto-loads. On B/C, attach the three files.
2. Ask: *"where did I leave off?"*
3. Work. At session end, update `state.md`.

**Monday Reset**

~10 min

1. Update each `project-*.md` with last week's decisions and status.
2. Archive last week's `state.md`. Start a fresh one.
3. Prune `patterns/` – delete anything unused in 30 days.

**The failure mode – name it so you avoid it.** Skip the Monday Reset for two weeks and `state.md` becomes a lie. Your AI will load it confidently and answer wrong. The whole system is downstream of this 10 minutes – protect it on your calendar.

1:1 coaching [pickbits.ai/coaching](https://pickbits.ai/coaching) // team training [pickbits.ai/consulting](https://pickbits.ai/consulting) // open source [github.com/pickbitsai/context-stack](https://github.com/pickbitsai/context-stack)

PICKBITS.AI // SETUP COMPANION // 5 / 5