

kv — Setup Guide

What kv is: your own private knowledge vault. You feed it YouTube videos and articles; it transcribes and organizes them on *your* computer, and then you can ask your AI assistant questions answered from your own collection — not the open internet.

The big idea behind this guide: you will **not** type technical commands. You will open this project in an AI coding assistant and **paste the short prompts below, one at a time**. The assistant does the actual work and asks you a few simple questions along the way.

Tip: keep this guide open beside your assistant. Copy a prompt, paste it, let it finish, then move to the next one.

1. What you need first (one-time, about 10 minutes)

You only have to set these up once.

Docker Desktop — the engine that runs kv on your machine. Download, install, and **open it** (leave it running): <https://www.docker.com/products/docker-desktop/>

An AI assistant that can work in a folder. Easiest for non-technical users: **Cursor** — a normal app you download, install, and open: <https://cursor.com> *Advanced option: Claude Code* (<https://claude.com/claude-code>), *a command-line tool — only pick this if terminals don't scare you. The Claude Desktop app can't run the setup itself; we connect it at the end so you can chat with your vault.*

(Optional, only to chat from the Claude Desktop app later) **Node.js**: <https://nodejs.org>

Which computers? Mac and Linux work as-is. On **Windows**, install **WSL** (Windows Subsystem for Linux) first — when you install Docker Desktop it offers to set WSL up for you; accept that. Then do everything inside WSL.

That's the only by-hand part. (Your assistant installs a small helper called `uv` automatically — you don't need to do anything for it.)

2. Open the project

You'll have received **kv-setup.zip** (or a download link) from whoever set this up for you. Unzip it somewhere easy to find — your Documents folder is fine — then open that folder in your assistant:

- **Cursor (recommended):** File → Open Folder → pick the unzipped `kv-setup` folder.

- **Claude Code (advanced):** open a terminal in the folder and run `claude`.

Then paste:

```
Open this kv-setup project so we can set it up together. I'm not technical – please do the work and explain things simply.
```

(You don't need GitHub, git, or any account — just the unzipped folder.)

3. Run the setup

This checks that everything's ready, asks you a few questions (which topics you care about, how often to update, etc.), starts kv, and connects it to your AI.

What's a "domain"? A domain is just a topic area you want to keep separate — like *fitness*, *finance*, or *cooking*. Each gets its own folder and its own search. During setup you'll pick a few (or make your own), and you can tell kv whether **newer** material should rank higher (great for news/markets) or **everything** should count equally (great for books and durable knowledge). When in doubt, take the defaults — you can change this later.

If you're using Claude Code, paste:

```
/kv-setup
```

Any other assistant (Cursor, etc.), paste this instead:

```
Run the kv first-time setup for me. I'm not technical, so do the work and explain simply. Steps:
1. Run ./kv doctor and make sure Docker is installed and running. If not, tell me exactly what to install.
2. Ask me which topics I want (and any preferences), then configure kv non-interactively: run "./kv init --print-schema", write my answers to a JSON file, and run "./kv init --answers <that file>". Do NOT run a bare "./kv init" — its question prompts need a real terminal and will error.
3. Run ./kv up and wait for it to finish (it can take 10–15 minutes the first time — that's normal).
4. Run ./kv status and confirm everything is healthy.
If anything fails because a service is down, run ./kv repair and try again.
```

The first start is slow — it downloads a few gigabytes and can take 10–15 minutes. The screen may look frozen; it isn't. This only happens once. Grab a coffee.

3a. ■■ Restart your assistant (don't skip this)

When setup finishes, **fully quit your assistant and reopen it** — this is how it picks up the connection to your new vault. Without it, asking questions won't work and you'll think setup failed.

- **Fully quit** means quit the whole app, not just close the window: **Cmd+Q** on Mac, **Ctrl+Q** (or right-click the taskbar icon → Quit) on Windows/Linux.
 - Then reopen the assistant and **open the `kv-setup` folder again**.
-

4. Add your first source

Give kv something to learn from — a YouTube link or an article URL.

Claude Code:

```
/kv-add https://www.youtube.com/watch?v=XXXXXXXXX
```

Any assistant:

```
Add this link to my kv vault so it gets transcribed and indexed:  
https://www.youtube.com/watch?v=XXXXXXXXX
```

kv processes new links automatically on a schedule. If you're impatient, ask your assistant to "run the ingest now."

Other things you can add (Claude Code shortcuts shown; in any assistant, just ask in plain words):

- A **PDF or EPUB** on your computer → `/kv-file ~/Downloads/book.pdf` — it's split into chapter notes.
- A **note of your own** → `/kv-note` — your assistant asks for the title and content and saves it into the right topic.

If you have more than one topic, your assistant will ask **which domain** to file it under, so nothing lands in the wrong place. You can always see your topics by asking it to "list my kv domains."

5. Ask your vault a question

Claude Code:

```
/kv-ask What did the videos I saved say about sleep and recovery?
```

Any assistant:

```
Using my kv vault (the "kv" connection), answer this from my own saved sources and cite which notes you used: What did my saved videos say about sleep and recovery?
```

If it says it can't reach the vault, paste `/kv-doctor` (or "run kv doctor and repair, then try again").

6. (Optional) Chat with your vault from the Claude Desktop app

If you said "yes" to connecting Claude Desktop during setup:

1. Make sure **Node.js** is installed (see step 1).
 2. **Fully quit and reopen** the Claude Desktop app.
 3. Start a new chat and just ask — your vault shows up as a connected tool called **kv**.
-

7. If something ever breaks

Paste this anytime:

Claude Code:

```
/kv-doctor
```

Any assistant:

```
Something's wrong with kv. Run ./kv doctor, then ./kv repair if needed, and tell me in plain language what was wrong and whether it's fixed.
```

It safely restarts anything that stopped. It will **ask you first** before doing anything that could affect your saved data.

8. The menu — when you're not sure what to do

Claude Code: `/kv-menu`

Any assistant: "Show me the kv menu — what can I do and how do I get back to the setup guide?"

Browsing your vault (optional)

Your notes are ordinary text files in the vault folder (default `~/kv-vault`), so you can open them in any editor. For a nicer reading/editing experience, install **Obsidian** (free, <https://obsidian.md>) and choose "**Open folder as vault**" pointing at that folder. kv does **not** require Obsidian — it's just a pleasant way to browse what kv has collected.

Starting over / removing kv

If you want a clean slate (or to uninstall):

Any assistant:

```
Run ./kv uninstall — show me what it will remove first, and only run it with
--yes after I confirm.
```

`./kv uninstall` stops and removes kv's containers, indexes, and downloaded models, and removes the background indexer jobs. **It does not touch your vault** (your notes), so you can reinstall later with `/kv-setup` and pick up where you left off. To just pause kv without removing anything, use `./kv down`.

Tidiness note: ingested PDFs/EPUBs keep their original file in the vault's `_inbox/` folder. To reclaim that space, ask your assistant to run `./kv inbox clean` (the notes stay; only the original files are removed).

Good to know

- **Your data stays on your computer.** kv runs locally; nothing is uploaded to a third party.
- **Reminders are quiet.** kv won't pop up notifications. Review nudges and health checks are written into a digest inside your vault — ask your assistant to "show my kv reminders" or run `./kv status`.
- **Leave Docker Desktop running** while you use kv. If you restart your computer, just reopen Docker Desktop and your assistant can run `/kv-doctor` to bring kv back up.
- **Stuck?** `/kv-doctor` fixes most things; `/kv-menu` reorients you; this guide is always here.