

Lesson 04 — Installing the tools

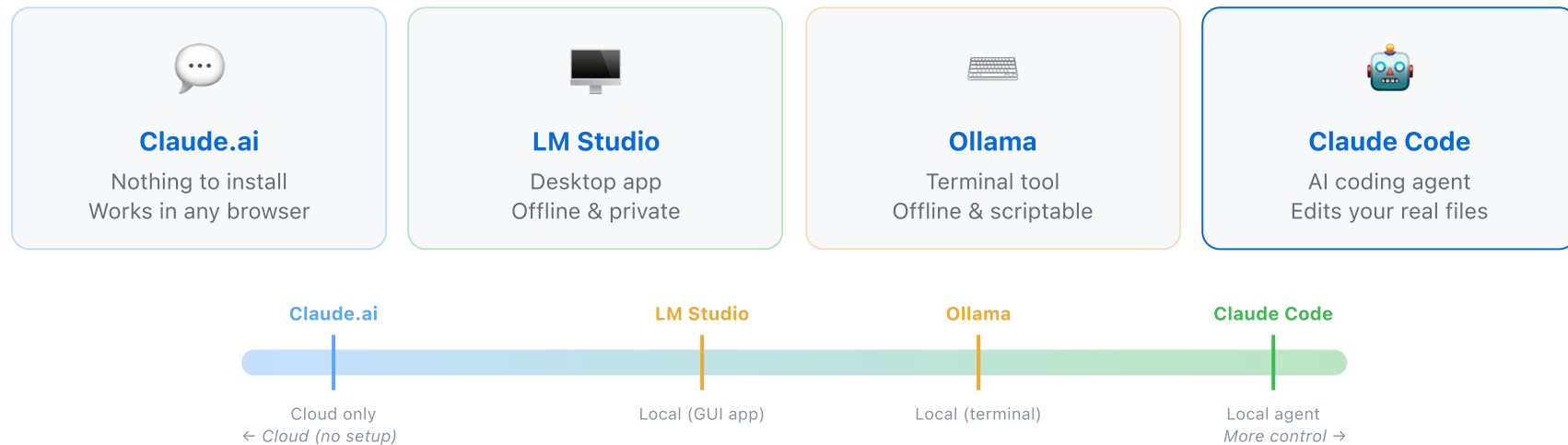
Four tools, from “no install needed” to “AI coding agent on your own machine.” Pick one, start today.

AI for Dummies

Module 4

You don't need all four. Pick the one that fits where you are *right now*. The rest will still be here when you're ready.

The four tools at a glance



Beginner path: **Claude.ai first**. Add the others only when you hit a task Chat can't do on its own.

Hardware reality check — local AI only

Cloud AI (Claude.ai, Claude Code) runs on Anthropic's servers — your machine specs don't matter. Local AI (LM Studio & Ollama) runs on *your* RAM, so this matters:

8 GB RAM

Small models (~3B params)

Works, but slowly

~4–8 tokens/sec

Good for: simple Q&A,
short drafts

Model: llama3.2:3b

16 GB RAM

Mid models (7B–13B)

Comfortable for everyday tasks

~8–20 tokens/sec

Good for: writing, summaries,
simple code

Model: llama3.2 or mistral

32 GB+ RAM

Large models (30B+)

Smooth, near-cloud quality

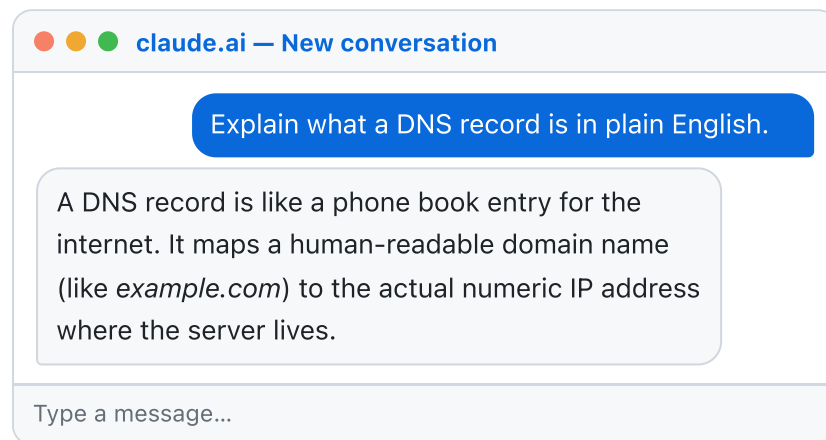
~20+ tokens/sec

Good for: anything, long docs,
complex reasoning

Model: llama3.1:70b (q4)

Apple Silicon Macs (M-series) and discrete GPUs roughly double these speeds. If your machine is modest, cloud AI is your friend — it runs on Anthropic's hardware, not yours.

Tool 1 — Claude.ai — three steps to get started



Step 1: Go to **claude.ai** in any browser — Chrome, Firefox, Safari, Edge.

Step 2: Sign up with your **email** or a Google account. Takes under a minute.

Step 3: Start typing. That's the entire setup.

Free tier is enough to learn on. Desktop & mobile apps available — same login, same experience.

Claude.ai plans — what you need

Free

- Unlimited short conversations
- Claude Sonnet (very capable)
- Works on any device
- No credit card needed
- **More than enough to learn on**

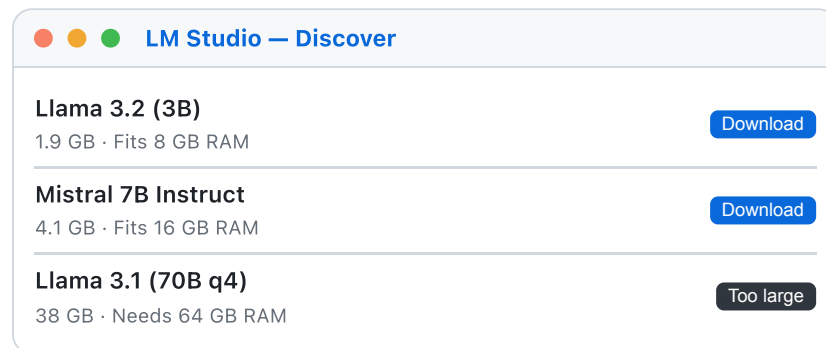
Pro & Max (paid)

- Higher usage limits
- Access to Claude Opus (most capable)
- Priority access during peak hours
- Projects, extended context
- Required for Claude Code

Recommendation: Start free. If you hit usage limits while doing real work, upgrade then — not before. The free tier handles learning perfectly.

Tool 2 — LM Studio — local AI with no terminal

The **Discover** tab — a built-in model browser:

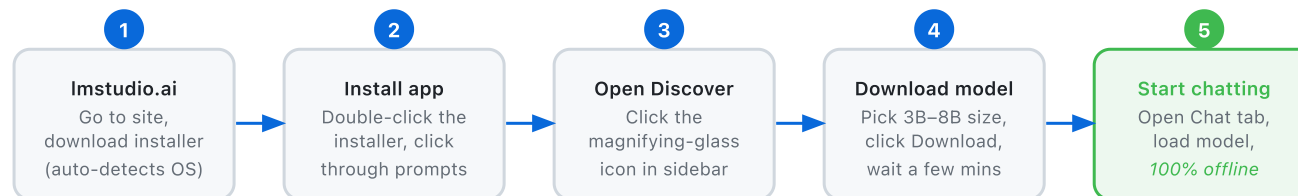


1. Download from **lmstudio.ai** — Mac, Windows, or Linux. Installs like any normal app.

2. Open the **Discover** tab. Find a 3B–8B model. Click *Download*.

3. Open **Chat**, load the model, start talking — *fully offline*.

LM Studio — install to first chat in 5 steps



Bonus — local API server: LM Studio can serve a local API on port `1234` (OpenAI-compatible format). Apps that normally call a cloud API can point at your machine instead — zero cost, zero data leaving home. You don't need this on day one; it's there in Module 6.

LM Studio vs Ollama — which should you pick?

LM Studio

- Point-and-click GUI — no terminal
- Built-in model browser with size warnings
- Visual chat interface just like Claude.ai
- Local API server toggle (one click)
- Best for: **beginners, visual people, anyone who wants no command-line**

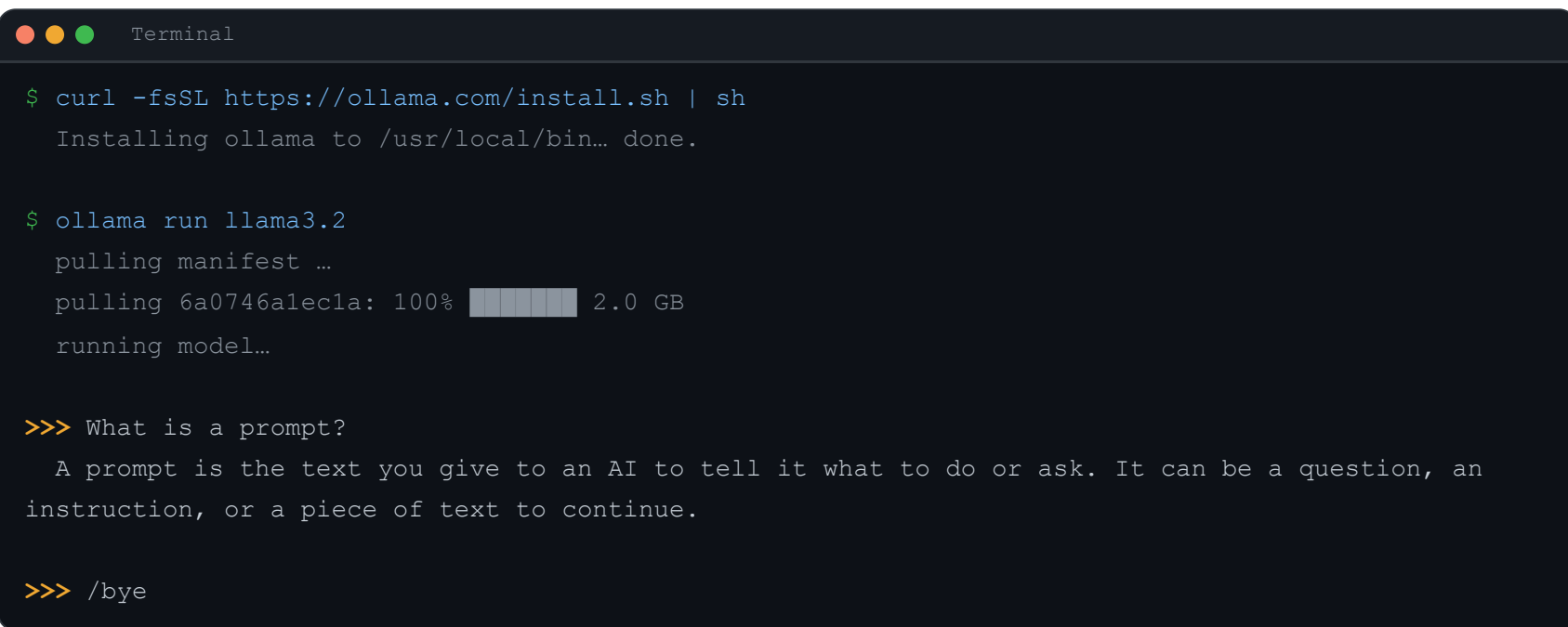
Ollama

- Tiny (~60 MB) — runs in the background
- One-command install & model download
- Scriptable: automate with a shell loop
- Local API on port 11434 (OpenAI-compatible.)
- Best for: **developers, servers, automation**

The one-line rule: LM Studio is friendlier (clicks, no terminal). Ollama is leaner and better for scripting. Most beginners try LM Studio first, then discover Ollama when they want to automate something.

Tool 3 — Ollama — AI in your terminal

Install once, then run any open model with a single command. First run downloads the model (a few GB); after that it loads in seconds.

A terminal window with a dark background and light text. The title bar shows three colored circles (red, yellow, green) and the word "Terminal". The terminal content shows the installation of Ollama using curl, followed by running the llama3.2 model. It shows progress bars for pulling the manifest and the model itself (2.0 GB). Finally, it shows a prompt being asked and answered, and then the user typing /bye to exit.

```
$ curl -fsSL https://ollama.com/install.sh | sh
Installing ollama to /usr/local/bin... done.

$ ollama run llama3.2
pulling manifest ...
pulling 6a0746a1ec1a: 100% ██████████ 2.0 GB
running model...

>>> What is a prompt?
A prompt is the text you give to an AI to tell it what to do or ask. It can be a question, an
instruction, or a piece of text to continue.

>>> /bye
```

Type `/bye` to exit. On Linux use the one-liner above. On Mac/Windows download from ollama.com/download.

Ollama — commands you'll actually use

Memorize these four and you'll cover 90% of what Ollama can do:

```
ollama run llama3.2      # download (first time) + start a chat
ollama list              # see every model you've downloaded
ollama pull mistral      # download a model without running it
ollama rm llama3.2       # delete a model to free up disk space
ollama serve             # run Ollama as a background API server
```

Popular starter models:

`llama3.2` — Meta's fast 3B model

`mistral` — Mistral 7B, excellent balance

`phi3` — Microsoft's small but smart model

Local API: Ollama automatically exposes an **OpenAI-compatible API on port 11434**. Your own scripts can call it just like the cloud API — free, no data leaving home. See Module 6.

Tool 4 — Claude Code — AI that acts, not just answers

Claude Code is a **coding agent**. You run it inside a project folder and describe what you want in plain English. It reads your files, writes or edits code, and runs commands on your behalf.

What it can do:

- Read your entire project codebase
- Write new files or edit existing ones
- Run terminal commands and see output
- Fix bugs you describe in plain English
- Scaffold a new project from scratch

What you need first:

- A paid Claude plan (Pro, Max, Team, or Enterprise)
- — or — an Anthropic Console account with billing
- macOS, Linux, or Windows (WSL)
- ~4 GB RAM (very light)

The free Claude.ai tier does not include Claude Code. If you're just starting out, set it aside and come back when you have a coding project to work on.

Claude Code — two install options

Option A — Native installer

Simplest. No extra software needed:

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

Works on macOS and Linux out of the box.

Option B — Via npm

If you already have Node.js 18 or newer:

```
npm install -g @anthropic-ai/claude-code
```

On Windows: use inside WSL (Windows Subsystem for Linux).

After installing, `cd` into any project folder and type `claude`. It will open a browser window to authenticate on first run. That's it — you're running an AI coding agent.

Claude Code — one important rule before you start

Claude Code can edit real files and run real commands. It is not a chat window — it has actual access to your project on disk.

Do NOT start it on...

- A production system or live website
- A folder with files you haven't backed up
- Anything containing real credentials (.env)

DO start it on...

- A test or sandbox folder
- A project tracked by git (easy undo)
- A fresh directory for a new project

Think of it as a junior developer you're mentoring: talented, fast, but *you review the work before it goes live*.

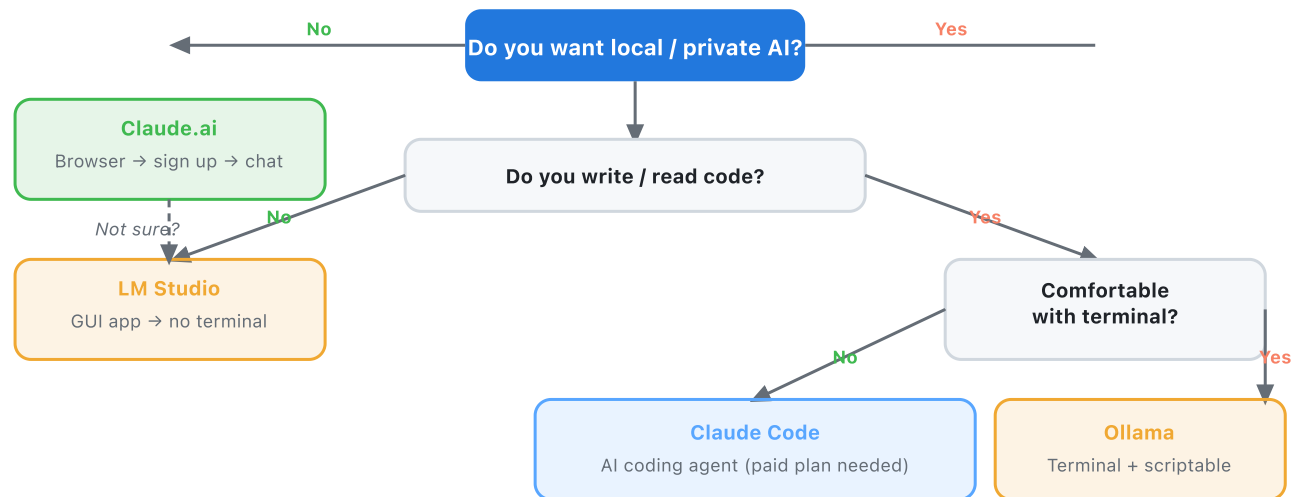
Read what it proposes before approving. More on agent safety in Module 9.

All four tools side by side

Tool	Runs where	Install?	Privacy	Best for
Claude.ai	Anthropic’s cloud	No	Data sent to Anthropic	Everyone, day one
LM Studio	Your machine (GUI)	App download	Fully offline	Private AI, no terminal
Ollama	Your machine (CLI)	Yes (one command)	Fully offline	Scripting, servers, devs
Claude Code	Your machine + cloud	Yes (curl or npm)	Code sent to Anthropic	Coding, multi-step tasks

All four install commands in one place: see the **Install Commands Gallery** in the Theory page — copy buttons included.

Pick your path — decision tree



Unsure about your situation? Use the **Tool Picker Wizard** in the Theory page — it asks 3 questions and gives you the exact install command.

A suggested 4-week path

Week 1 — Chat only. Use Claude.ai every day for real tasks. Learn what good output looks like. Practice asking — don't install anything else yet.

Week 2 — Go local. Install LM Studio, download a 3B–8B model, have the same conversation you had on Claude.ai. Compare quality.

Week 3 — Touch the API. Run the Python example from Module 6. Process a small batch of something you actually care about.

Week 4 — Build one tiny thing. Pick a starter project from Module 8 and finish it.

You don't need to rush. Each step works fine on its own. The later ones just unlock more. The single best thing you can do *today*: sign up for Claude.ai and use it for something real.

Recap — what you covered today

Four tools

- **Claude.ai** — cloud, zero install, start here
- **LM Studio** — local GUI app, offline & private
- **Ollama** — local terminal tool, scriptable
- **Claude Code** — AI coding agent, edits real files

Key install commands

```
# Ollama (Linux/WSL)
curl -fsSL https://ollama.com/install.sh | sh

# Claude Code
curl -fsSL https://claude.ai/install.sh | bash

# Or via npm
npm install -g @anthropic-ai/claude-code
```

Your homework

Install *one* tool. Use it for a real task. Take a screenshot for the homework check.

Lesson 05 →

Prompting — how to talk to AI well. The skill that improves your results more than any tool choice.

