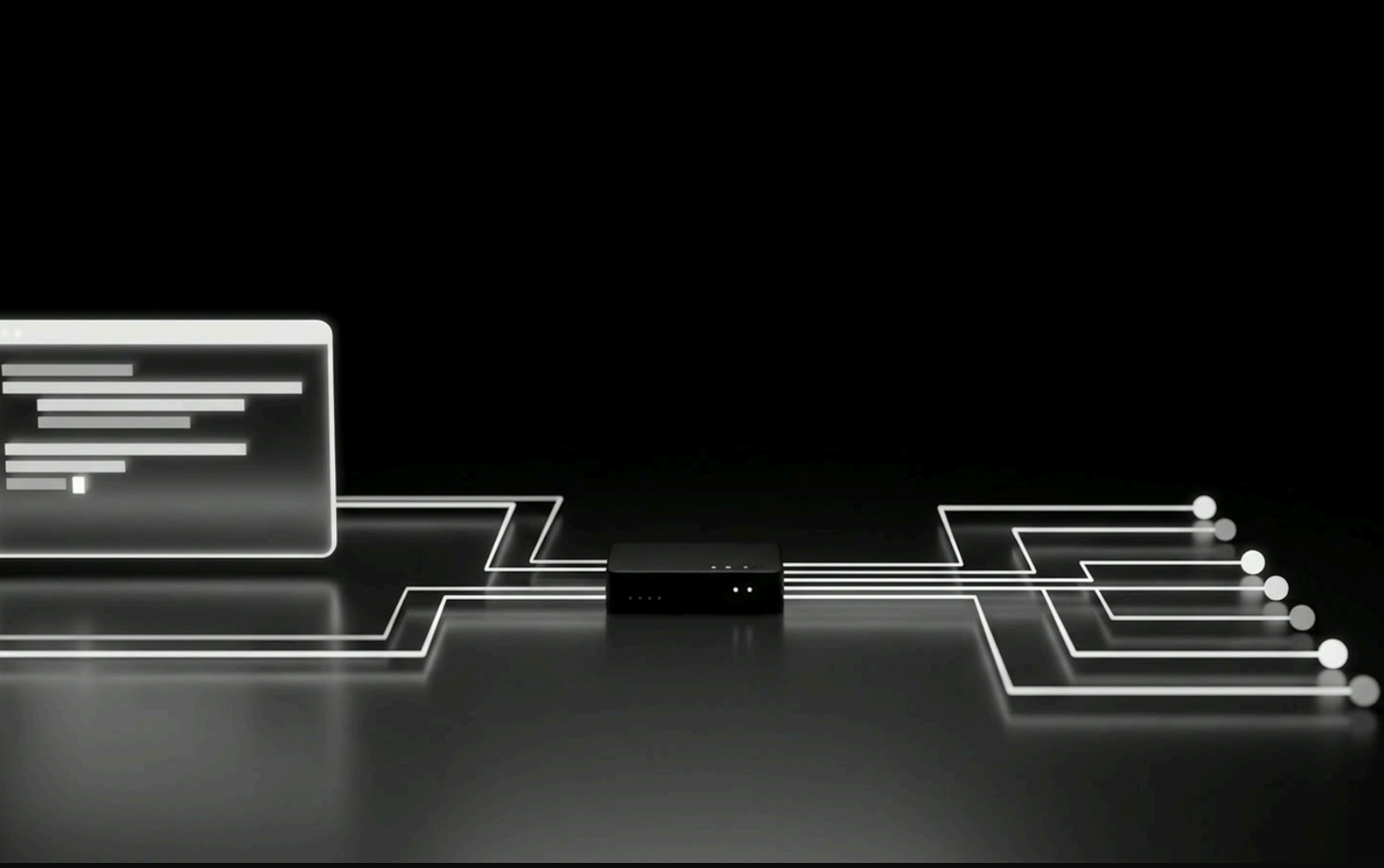


A FIELD GUIDE · KLYMENTIEV.COM

How to Run an AI Agent for Free

Claude Code, Flint Agent, a chat app and a helper script on free AI models. No subscription, no credit card, about fifteen minutes.

NOVEMBER 2026

TESTED ON A FRESH INSTALL, NO KEYS

BEFORE YOU START

One address, many free models, and an agent that never stops

AI agents need a model to think with, and the good ones charge for it. Free models exist, but each provider has its own address, key and daily limit, and any of them can run out mid-task.

Wend router sits between your agents and those providers. It gives all your agents one address on your own computer. Behind it, it keeps a list of free models that are up right now, sends each request to one that fits, and when a model fails or reaches its limit, moves the request to the next one on its own.

17	$0	19 s
free models up right away, with no keys added	spent across every test in this guide	for Claude Code to finish its first task on free models

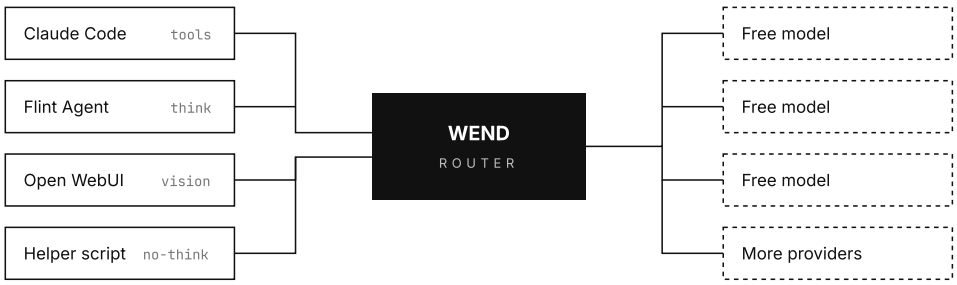


FIGURE 1 *Four agents, four profiles, one local address. The router picks a free model behind it and moves on when one runs out.*

01 Install the router page 3

02 Add providers, the suppliers of intelligence page 4

03 One profile for each kind of job page 5

04 Point four agents at four profiles pages 6 to 8

01 Install the router

One line in a terminal. No administrator rights, no account, nothing to configure yet.

Linux or macOS

```
curl -fsSL https://raw.githubusercontent.com/dklymentiev/wend-router/main/install.sh | sh
```

Windows, in PowerShell

```
irm https://raw.githubusercontent.com/dklymentiev/wend-router/main/install.ps1 | iex
```

Everything goes into one folder, `~/wend` or `%LOCALAPPDATA%\wend` on Windows. The script starts the router, sends one test request and opens the dashboard at `http://127.0.0.1:8888/dashboard`.

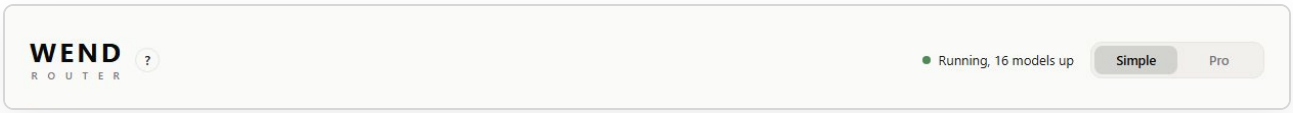


FIGURE 2 *The top of the dashboard. On the right it says the router is running and how many models are up.*

ON YOUR OWN SERVER

The same command works on a server. By default the router listens only on the machine itself. To let other machines use it, start it with `ROUTER_ADDR=0.0.0.0:8888` and a password in `ROUTER_TOKEN`; every client then sends that password as its key.

02 Add providers

Providers are the companies whose models do the thinking. A few need no key at all, so this step is optional: it simply gives you more models and more room before daily limits.

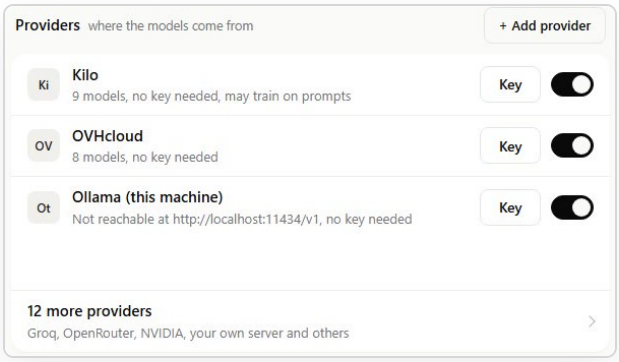


FIGURE 3 *Each row says how many models a provider gives and whether it needs a key.*

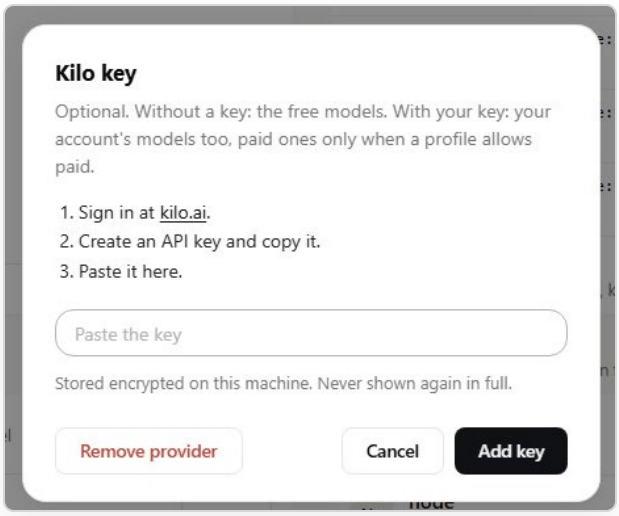


FIGURE 4 *The key window walks you through it.*

1. Press **Key** next to a provider.
2. Open its sign-up link, create a free API key and copy it.
3. Paste it and press **Add key**. Keys are stored encrypted on your own computer.

Which providers give free keys, and their limits: **klymentiev.com/blog/free-llm-api**.

PRIVACY

Some free providers may train on what you send. The Providers panel marks them. For anything you would not post publicly, use the `private` profile, which skips them.

03 One profile for each kind of job

A profile is a name your agent sends instead of a model name. The router then chooses only among the models that fit it, and switches between them.

THE JOB	PROFILE	WHAT IT TAKES
Simple, quick answers	`no-think`	models without a reasoning phase, quickest first
Harder questions	`think`	models that reason; you create this one below
Pictures	`vision`	models that read images
Code	`tools`	models that call tools well, for coding agents

Also ready to use: `auto` (every model, strongest first), `fast` (quickest to start first) and `private` (only providers that do not train on prompts).

Create the `think` profile

1. Switch the dashboard to **Pro**, top right.
2. Under **Profiles**, press **+ New profile**.
3. Name it `think`, set **Thinking (reasoning)** to **Required**, press **Save**.

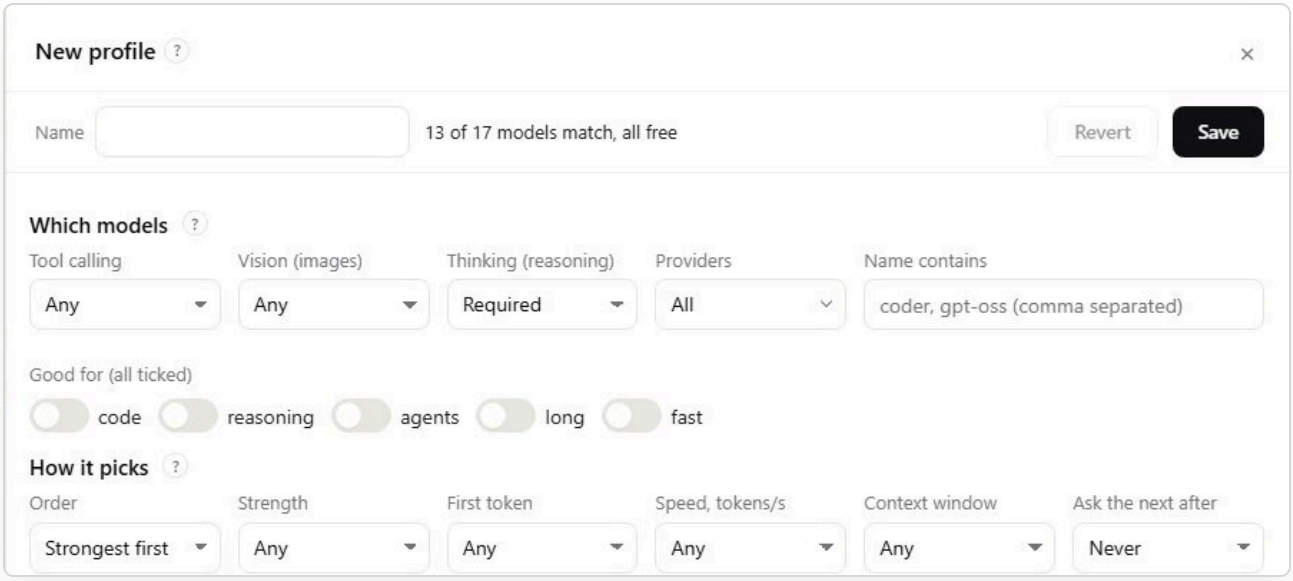


FIGURE 5 *The editor shows at once how many models match, and lists them in the order it will try them.*

04 Point your agents at the router

Every agent gets the same three settings. The dashboard shows them under Connect your agent, with a Test button.

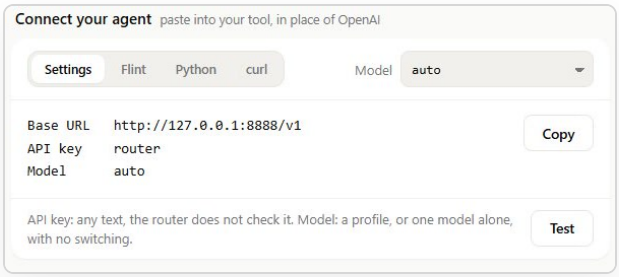


FIGURE 6 *Copy the settings, or press Test to send one request through.*

SETTING	VALUE
Base URL	`http://localhost:8888/v1`
API key	any text, unless you set `ROUTER_TOKEN`
Model	a profile

Tools that speak the Anthropic API, such as Claude Code, take the address without `/v1`.

Four agents, four profiles

AGENT	PROFILE	IN OUR TEST
Claude Code	`tools`	created a file in 19 seconds, $0
Flint Agent	`think`	solved a trick question and saved the answer, about 5 seconds, $0
Open WebUI	`vision`	described a picture correctly
A helper script	`no-think`	turned a sentence into a to-do item

The next two pages give the exact settings for each one. Any other tool that asks for an OpenAI address works the same way.

AGENT 1 OF 4

tools

Claude Code

Claude Code is free to install, but with Anthropic's own models it needs a paid plan or API billing. Pointed at the router, it runs on free models instead. Set three variables, then start it.

Linux or macOS

```
export ANTHROPIC_BASE_URL=http://localhost:8888
export ANTHROPIC_AUTH_TOKEN=wend
export ANTHROPIC_MODEL=tools
claude
```

Windows PowerShell

```
$env:ANTHROPIC_BASE_URL = "http://localhost:8888"
$env:ANTHROPIC_AUTH_TOKEN = "wend"
$env:ANTHROPIC_MODEL = "tools"
claude
```

The answers then come from free third-party models, not from Anthropic's Claude. For hard, long coding work they are weaker than Claude's best models.

AGENT 2 OF 4

think

Flint Agent

Flint Agent is an open-source AI agent for the terminal (flintagent.dev, source at github.com/dklymentiev/flint-agent, MIT). It works with the router out of the box; here it runs on the reasoning profile.

```
npm install -g flint-agent
```

```
export FLINT_PROVIDER=openai
export FLINT_BASE_URL=http://localhost:8888/v1
export OPENAI_API_KEY=wend
flint --model think
```

On Windows write each line as `$env:FLINT_PROVIDER = "openai"`. The dashboard's Connect section also has a ready Flint tab.

AGENT 3 OF 4

vision

Open WebUI, a chat that reads images

Open WebUI is a free chat interface you run yourself (openwebui.com). Start it with two settings that point it at the router:

```
OPENAI_API_BASE_URL=http://localhost:8888/v1
OPENAI_API_KEY=wend
```

It then lists the router's profiles as models. Choose `vision`, attach a picture and ask about it.

AGENT 4 OF 4

no-think

A helper script for quick jobs

For small everyday jobs a reasoning phase only adds waiting. A few lines of Python on the quick profile, after `pip install openai`:

```
from openai import OpenAI
client = OpenAI(base_url="http://localhost:8888/v1", api_key="wend")
reply = client.chat.completions.create(
    model="no-think",
    messages=[{"role": "user",
               "content": "Turn this into a to-do item: "
                          "call the supplier about the late order."}])
print(reply.choices[0].message.content)
```

Our run printed: *Call the supplier regarding the overdue order.*

ANY OTHER TOOL

Wherever an app asks for an OpenAI address, give it `http://localhost:8888/v1`, any text as the key, and a profile as the model.

GOOD TO KNOW

What free costs you, and what to do about it

Free is real here, but it comes with limits. None of them is a surprise if you know them up front.

THE LIMIT	WHAT TO DO
Free models are weaker than the best paid ones	Use them for everyday jobs. For hard, long coding work a paid plan is still the stronger tool.
Each provider has a daily limit	The router moves to another model on its own when one runs out. Add more providers in Step 02 for more room.
Some free tiers may train on prompts	They are marked on the Providers panel. Use the `private` profile for anything you would not post publicly.
Paid models	Stay off unless you allow them, and then only within a budget you set.

GOING TO PRODUCTION

Free tiers are for learning, side projects and prototypes. When real users, deadlines or customer data depend on it, use paid models: they are stronger, and their limits are ones you pay to raise rather than ones that run out mid-task. Wend router can stay in front of them. Add a paid provider's key, allow paid models in a profile (in Simple mode: **Allow paid**, only after the free ones, within the daily budget), and set the budget.

HOW THIS GUIDE WAS TESTED

On November 10, 2026, on a fresh install of Wend router 0.2.0 with no provider keys. Every command in this guide was run as printed, and every result quoted came from that run.

WEND

ROUTER

Where to go next

Wend router, free and open source (MIT)	github.com/dklymentiev/wend-router
The full article	klymentiev.com/blog/claude-code-free
Where to get free keys	klymentiev.com/blog/free-llm-api
Flint Agent	flintagent.dev github.com/dklymentiev/flint-agent

SHARE THIS GUIDE

X | LinkedIn | Reddit | Telegram | Email

Free tiers change fast. Subscribers of klymentiev.com hear when a provider cuts its free plan or a new one appears.