



FREE SETUP GUIDE

Free Claude Code via OpenRouter

A structured, copy-paste walkthrough for connecting Claude Code to OpenRouter's free AI models — configuration prompt, API key setup, and terminal install, all in one place.

STEP 01

Configure Claude Code

STEP 02

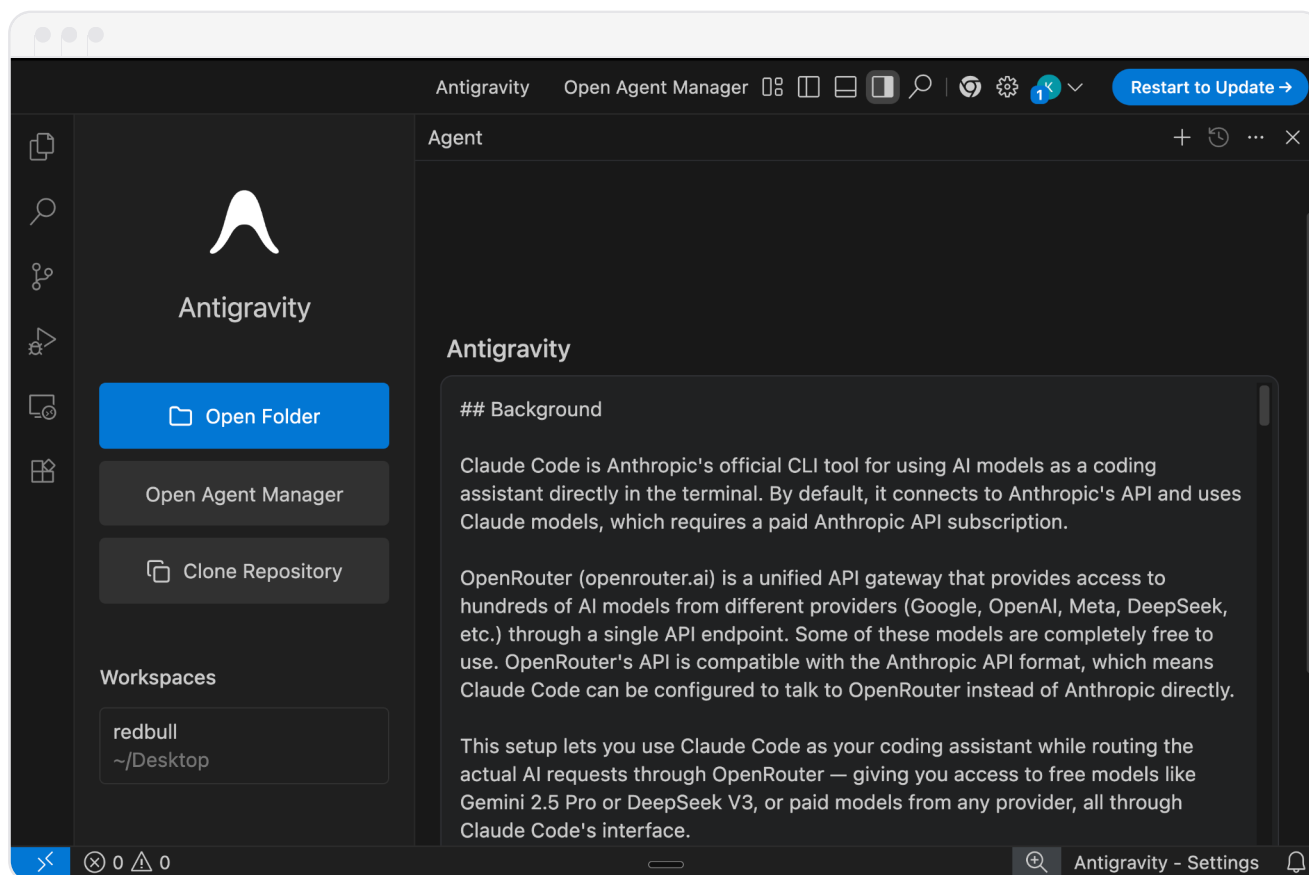
Get OpenRouter API Key

STEP 03

Install in Terminal

01 Step 1

Copt this paste inside antigravity chat -



PROMPT (COPY IT TILL THE RED LINE)

Background

Claude Code is Anthropic's official CLI tool for using AI models as a coding assistant directly in the terminal. By default, it connects to Anthropic's API and uses Claude models, which requires a paid Anthropic API subscription.

OpenRouter (openrouter.ai) is a unified API gateway that provides access to hundreds of AI models from different providers (Google, OpenAI, Meta, DeepSeek, etc.) through a single API endpoint. Some of these models are completely free to use. OpenRouter's API is compatible with the Anthropic API format, which means Claude Code can be configured to talk to OpenRouter instead of Anthropic directly.

This setup lets you use Claude Code as your coding assistant while routing the actual AI requests through OpenRouter — giving you access to free models like Gemini 2.5 Pro or DeepSeek V3, or paid models from any provider, all through Claude Code's interface.

What You Need To Do

Configure Claude Code's settings file (`~/.claude/settings.json`) so that it sends API requests to OpenRouter instead of Anthropic. This is done by setting environment variables inside the settings file that override Claude Code's default API connection.

My Details

- OpenRouter API Key: [PASTE YOUR KEY HERE — get one free at <https://openrouter.ai/keys>]
- Model: `openrouter/free`

Step 1: Create the config directory

Run this command:

```
mkdir -p ~/.claude
```

This creates Claude Code's configuration directory if it doesn't already exist. It's safe to run even if the directory already exists.

Step 2: Write the settings file

The file `~/.claude/settings.json` controls Claude Code's behavior. It supports an "env" block that sets environment variables for the Claude Code process. We use this to redirect API calls to OpenRouter.

Here is what each variable does:

ANTHROPIC_BASE_URL: The API endpoint Claude Code sends requests to. We point this to OpenRouter instead of Anthropic.

ANTHROPIC_AUTH_TOKEN: The authentication token sent with API requests. We use our OpenRouter API key here.

ANTHROPIC_API_KEY: Claude Code's default Anthropic API key. We set this to an empty string to prevent Claude Code from trying to authenticate with Anthropic or prompting the user to log in.

ANTHROPIC_MODEL: The model ID to use for AI requests. This is the OpenRouter model ID.

Create or update `~/.claude/settings.json` with this content. Replace the two placeholders with my details from above:

```
{
  "env": {
    "ANTHROPIC_BASE_URL": "https://openrouter.ai/api",
    "ANTHROPIC_AUTH_TOKEN": "[MY OPENROUTER API KEY]",
    "ANTHROPIC_API_KEY": "",
    "ANTHROPIC_MODEL": "openrouter/free"
  },
  "model": "sonnet[1m]"
}
```

The "model" field at the top level is required for Claude Code's internal validation. It tells Claude Code which internal model profile to use for things like token limits and context windows.

"sonnet[1m]" means "use the Sonnet profile with a 1 million token context window." This does NOT control which AI model answers your questions — ANTHROPIC_MODEL does that. Do not remove or change the "model" field.

If `~/.claude/settings.json` already exists and contains other settings (like "permissions", "projects", or "allowedTools"), merge the "env" block and "model" field into the existing file. Preserve all existing settings — only add or update the keys shown above.

Rules You MUST Follow

- 1 ANTHROPIC_BASE_URL must be exactly `"https://openrouter.ai/api"` — do NOT append `/v1` to it. Claude Code automatically appends `/v1` when constructing API request URLs. If you write `"https://openrouter.ai/api/v1"` here, the actual request will go to `/api/v1/v1/messages`, which is a broken URL that returns 404 errors on every single API call.
- 2 The OpenRouter API key goes in ANTHROPIC_AUTH_TOKEN — NOT in ANTHROPIC_API_KEY. These are two different environment variables that serve different purposes. ANTHROPIC_AUTH_TOKEN is the bearer token sent in the Authorization header of API requests. ANTHROPIC_API_KEY is Claude Code's internal Anthropic key, which must be set to an empty string `""` to prevent Claude Code from trying to use Anthropic's API directly.
- 3 Do NOT modify `~/.zshrc`, `~/.bashrc`, `~/.bash_profile`, or any other shell configuration file. The settings.json "env" block is the only correct place for these variables. Setting them in shell config creates conflicts and unpredictable behavior because both sources try to control the same settings.
- 4 Do NOT remove or change `"model": "sonnet[1m]"`. It is required for Claude Code to start. Without it, Claude Code's model validation fails. It does not affect which AI model processes your requests.

- 5 Do NOT add extra fields like "effortLevel", "preferences", "skipDangerousModePermissionPrompt", or any other keys not shown above (unless merging with an existing file that already has them).
- 6 Do NOT try to test the API connection by running a curl command or making any API request. The settings only take effect when Claude Code starts a new session. Testing from the current session will fail because the environment variables aren't loaded yet.
- 7 The JSON file must be syntactically valid — no trailing commas, no comments, proper double-quoting on all keys and string values.

Step 3: Verify the file

After writing the file, read back the contents of `~/.claude/settings.json` and confirm ALL of the following:

- ✓ ANTHROPIC_BASE_URL is exactly "https://openrouter.ai/api" (ends with /api, NOT /api/v1)
- ✓ ANTHROPIC_AUTH_TOKEN contains the full OpenRouter API key (should start with sk-or-)
- ✓ ANTHROPIC_API_KEY is exactly "" (an empty string — it must be present as a key with an empty string value, not missing, not null)
- ✓ ANTHROPIC_MODEL contains the exact model ID: openrouter/free
- ✓ "model": "sonnet[1m]" is present at the top level of the JSON object
- ✓ The file is valid JSON with no syntax errors

If any of these checks fail, fix the file before proceeding.

Common Errors the User May See After Setup

"Model not found"

→ The ANTHROPIC_MODEL value doesn't match any model on OpenRouter. Model IDs are case-sensitive and must include the provider prefix. Check the exact ID at openrouter.ai/models.

429 rate limit errors

→ Free models have per-minute and per-day usage limits set by OpenRouter. Wait 60 seconds and try again. If it keeps happening, the daily limit may be reached.

"Invalid API key" or authentication errors

→ The API key is in the wrong variable. It must be in ANTHROPIC_AUTH_TOKEN. Confirm the key starts with sk-or- and was copied completely.

Claude Code still asks to log in to Anthropic

→ ANTHROPIC_API_KEY is either missing from the settings file or has a non-empty value. It must be present and set to exactly "" (empty string).

Settings don't seem to take effect

→ The user must quit their current terminal entirely and open a brand new terminal window. Then run ``claude``. Settings from settings.json are only loaded when Claude Code starts — they cannot be hot-reloaded into an existing session.

Step 4: Tell the user it's done

Tell me:

1. Setup is complete — Claude Code is now configured to use OpenRouter
 2. I need to QUIT my current terminal completely and open a brand new terminal window (not just a new tab)
 3. Run `claude` in the new terminal
 4. I should see my chosen model name in the Claude Code startup output, confirming it's connected through OpenRouter
 5. If I want to switch models later, I just change the ANTHROPIC_MODEL value in `~/claude/settings.json` and open a new terminal session
-

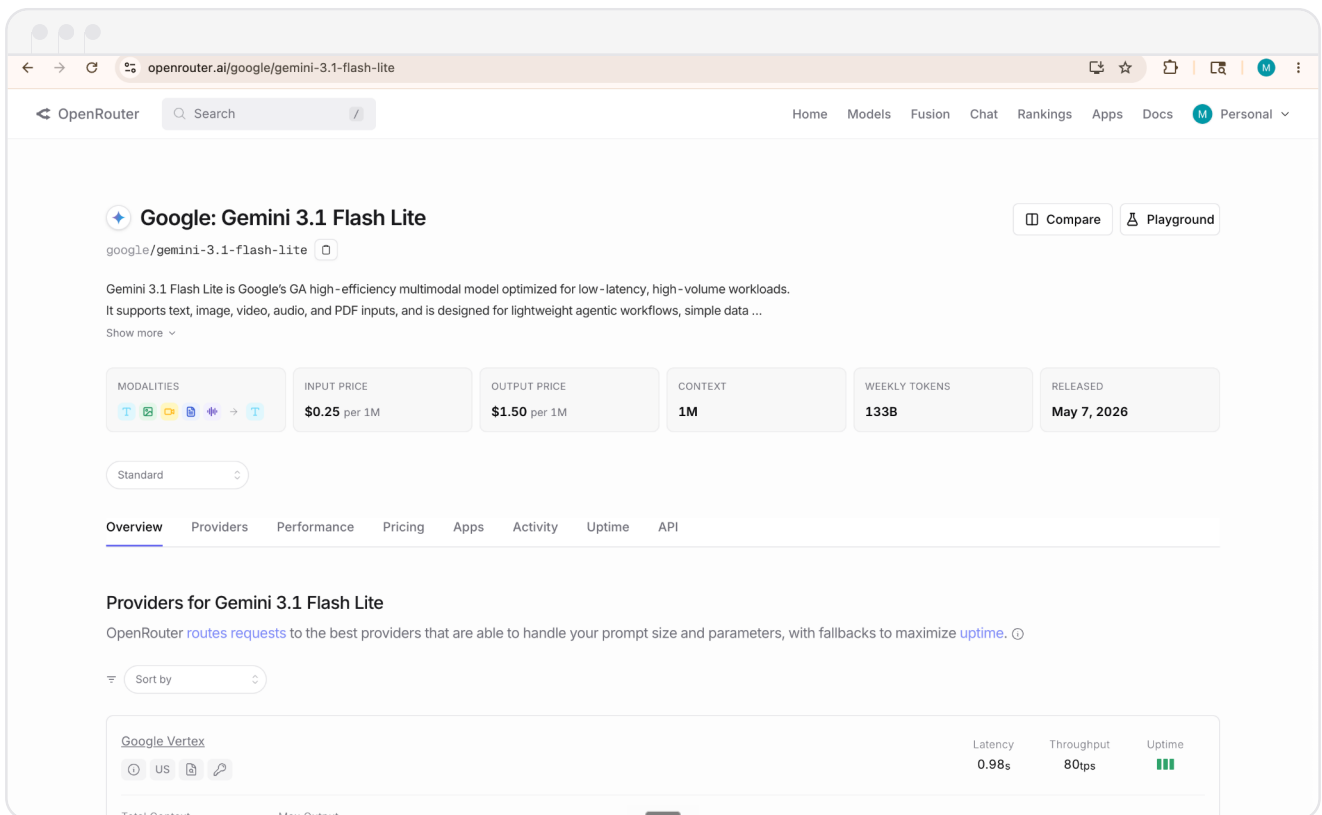
02 Step 2

After that go to open Router website and copy the API key

Open router: <https://openrouter.ai/>

Go to this website > Get an API Key > Create a new key and Copy the key > Paste it inside Antigravity

After that go to Models > Select API from any of the models > and paste it to antigravity



If you want to use multiple models

Tell Antigravity that you need to do multiple models, paste the prompt below and add more API from the models

Code to run in antigravity for making using multiple models, take API key's from Open router and put it

```
{  
  "env": {  
    "ANTHROPIC_AUTH_TOKEN": "your_api_key",  
    "ANTHROPIC_DEFAULT_SONNET_MODEL": "claude-sonnet-4",  
    "ANTHROPIC_DEFAULT_OPUS_MODEL": "claude-opus-4",  
    "ANTHROPIC_DEFAULT_HAIKU_MODEL": "claude-haiku-4"  
  }  
}
```

03 Step 3

Install Claude Code in the terminal

For terminal - run this command -

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

And then use this command to change the path

```
echo 'export PATH="$HOME/.local/bin:$PATH"' >> ~/.zshrc && source ~/.zshrc
```



ADCODEZ.COM

BUILD · AUTOMATE · SCALE