

# AI Assistants in Terminal

Shuwei Ye

HSF Seminar

June 4, 2025

# Contents

- ❑ **Aider**
- ❑ **Ollama**
- ❑ **Warp** Terminal
- ❑ **GitHub Copilot** CLI



# What is Aider?

**Aider** is **AI** pair programming in your **terminal**:

- **Powerful AI Coding**
- **Chat questions**
- **File editing**
- **Git Integration**
- **Linting & Testing**
- **More ...**

**Aider** is also available in **VSCode**, **JetBrains**, **NVim**, etc

Visit <https://aider.chat/> for more details

# Comparison: **aider** vs **ChatGPT**

Why **aider** in terminal?

- **Workflow Efficiency:** No need to copy/paste code b/w terminal and browser; edits are applied directly to files
- **Command Execution:** Can capture stdout/stderr directly and run suggested commands in the terminal env
- **Integration with Codebase:** Seamlessly works with local Git repos and files; maps entire codebase for context
- **Other useful features in **aider**, option **--watch-files****  
<https://aider.chat/docs/usage/watch.html>

# User Feedback

## Frank shared via email:

Hi Shuwei,

thanks for the Aider talk today. Super useful. I tried it for the first time this morning and already made good use of it for some Gaudi migrations, e.g.:

[https://gitlab.cern.ch/atlas/athena/-/merge\\_requests/80492](https://gitlab.cern.ch/atlas/athena/-/merge_requests/80492)

Definitely saved me lots of typing.

-Frank



## IOVRegistrationSvc: migrate to modern property declaration

 Open Frank Winklmeier requested to merge [fwinkl/athena:fwinkl\\_202...](#) into [main](#) 6 hours ago

**Overview** 2 Commits 1 Pipelines 1 Changes 2

Fun fact: Done entirely (apart from one missing include) via `gpt-4o` using `aider` with the following prompt:

```
Migrate the declareProperty calls in the .cxx file to Gaudi::Property calls in the .h file and remove the no longer needed variable initializations.
```

# Aider Installation

If **uv** is available, just run:

```
uv tool install --force --python python3.12 aider-chat
```

Otherwise, use **pip** to install:

```
pip install aider-install && aider-install
```

# Aider Setup

**API keys** are defined via **env variables**,  
or through **configuration** file: `~/.aider.conf.yml`

```
$ cat ~/.aider.conf.yml
# Specify the default model to use
model: gemini/gemini-2.5-flash-preview-05-20

# Chat API key, without enclosing quotation mark
api-key:
  - gemini=...
  - openrouter=...
```

## Free API Keys

- **Gemini models:** <https://aistudio.google.com/>  
Check rate limits: <https://ai.google.dev/gemini-api/docs/rate-limits>
- **OpenRouter:** <https://openrouter.ai/>
- **Local LLM Models**

# Aider Demo

How to use **aider** on command line (focus on shell commands, not on AI coding)

# Examples of Aider Runs

Examples on command line

- **aider --help**
- **aider --no-git --list-models=gemini/**
- **aider --no-git --list-models=openrouter/ | grep -i free**
- **aider --model=gemini-2.5**
- **aider --commit**  
# Commit all pending changes with a suitable commit message
- **aider**
- **aider --no-git**
- **aider file**

# Examples of **Aider** Commands

Commands inside **aider**

- **/ask** #Ask question with editing files
- **/clear** #Clear the chat history
- **/chat-mode ask** #Switch to ask chat mode
- **/copy** #Copy the last assistant message to the clipboard
- **/run** #Run a shell cmd & add the output to the chat
- **/test** #Run a shell cmd & add the output to the chat on non-zero exit code
- **/exit** #Exit the application

Run **/help** for more commands.

# Aider Chat History

The **Aider chat history** can be found in CWD

```
$ ls -1 .aider.*history*  
.aider.chat.history.md  
.aider.input.history
```

# Using Local ollama Models

**Aider** can also use **local LLM models**

To use **local models** through **ollama**:

```
$ ollama list
```

```
$ aider --openai-api-base http://localhost:11434/v1 \  
    --openai-api-key any --model ollama/gemma3:4b
```

# Ollama: Local AI Models

---

Local AI models can run via **ollama**

Download from

<https://ollama.com/>

Search/full the models from:

<https://ollama.com/library>



# Warp: Intelligent Terminal

Warp: AI-Powered Terminal

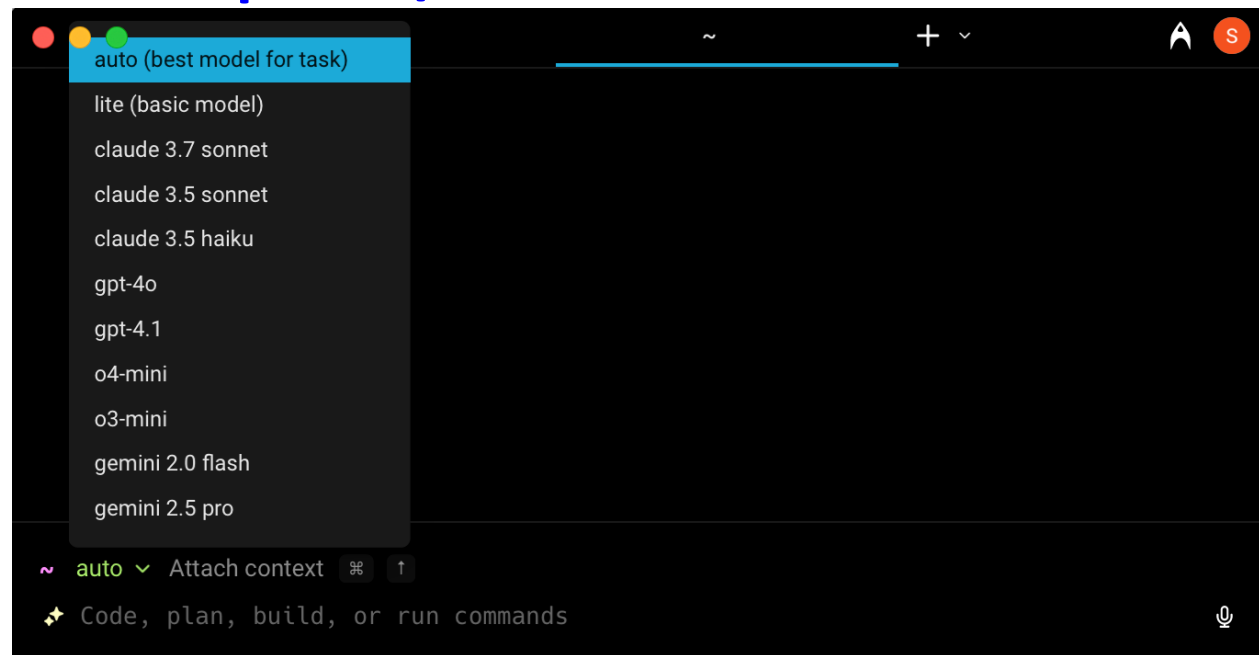
<https://www.warp.dev/>



Available for **Windows**, **Mac**, and **Linux**

The **built-in AI** is also available on **remote ssh machines**

**Free** plan: **Up to 150 requests / month**



# GitHub Copilot in Terminal

GitHub Copilot CLI can be installed



➤ **Mac:** `brew install gh`

➤ **Linux:**

```
$ sudo dnf config-manager \
  --add-repo https://cli.github.com/packages/rpm/gh-cli.repo
$ sudo dnf install -y gh
```

Then run:

```
$ gh auth login
$ gh extension install github/gh-copilot
```

Usage examples:

```
$ gh copilot suggest "Install git"
$ gh copilot explain "traceroute github.com"
```

For usage help, run **“gh copilot --help”**