# LangChain + OpenRouter Chatbot (CLI)

A minimal, fast CLI chatbot built with LangChain and OpenRouter (OpenAI-compatible API).

## ✨ Features

- OpenRouter-backed LLM with OpenAI-compatible endpoint

- Clean CLI loop (`exit` or `quit` to stop)

- Secrets managed via `.env` (never hardcode keys)

- Minimal dependencies

## 🏗️ Project Structure

langchain-openrouter-chatbot/  
├─ main.py # CLI entry point  
├─ requirements.txt # Python deps  
├─ .gitignore # Keep junk and secrets out of Git  
├─ LICENSE # MIT (change if you prefer)  
└─ .github/ # (optional) issue/PR templates later

## 🚀 Quickstart

1. Clone & enter the folder

git clone https://github.com/YOUR-USERNAME/langchain-openrouter-chatbot.git

cd langchain-openrouter-chatbot

2. Create & activate a virtual environment (recommended)

python -m venv .venv

.venv\Scripts\activate # Windows

# source .venv/bin/activate # macOS/Linux

3. Install dependencies

python -m pip install -r requirements.txt

4. Create `.env` (DO NOT COMMIT)

OPENROUTER\_API\_KEY=your\_api\_key\_here

5. Run

python main.py

## 🔒 Security Notes

- Never commit API keys. Keep them in `.env` which is already gitignored.

- If you ever leak a key, revoke & rotate it immediately.

## 📝 Suggested GitHub Name & Description

- Repository name: langchain-openrouter-chatbot

- Description: "CLI chatbot using LangChain with OpenRouter (OpenAI-compatible). Env-based secrets, ready to run."

- Topics/Tags: langchain, openrouter, openai-api, chatbot, python, cli

## 📦 How to Publish to GitHub (one-time setup)

git init  
git add .  
git commit -m "feat: initial commit - LangChain + OpenRouter CLI bot"  
git branch -M main  
git remote add origin https://github.com/YOUR-USERNAME/langchain-openrouter-chatbot.git  
git push -u origin main

## 🧭 Next Steps

- Add ISSUE\_TEMPLATE.md / PULL\_REQUEST\_TEMPLATE.md under `.github/`

- Add tests (e.g., pytest) and CI (GitHub Actions)

- Extend with memory, tools, or a Streamlit UI