

OpenAI Codex CLI 快速参考

CLI 用法、模式、配置、常用命令

快速上手

安装

```
npm install -g @openai/codex
```

身份验证

```
export OPENAI_API_KEY="sk-..."
```

在 shell profile 或 .env 文件中设置

首次运行

```
codex "explain this project"
codex "add input validation to app.py"
```

命令

交互模式 (默认)

```
codex # start interactive
session
codex "fix the login bug" # with initial prompt
```

codex exec

```
codex exec "write unit tests for utils.py"
codex exec "refactor db.py to use async"
```

非交互模式——执行任务直至完成

codex review

```
codex review # review staged
changes
codex review --diff HEAD~3 # review last 3
commits
```

模式

审批模式

suggest 展示建议更改，每次文件编辑和命令都需审批

auto-edit 自动应用文件编辑，命令需审批

full-auto 无需审批自动应用编辑并执行命令

设置模式

```
codex --approval-mode suggest "add tests"
codex --approval-mode auto-edit "refactor"
codex --approval-mode full-auto "fix lint"
```

配置

配置文件

```
# ~/.codex/config.yaml
model: o4-mini
approval_mode: suggest
providers:
  - name: openai
    api_key_env: OPENAI_API_KEY
```

项目级覆盖：在项目根目录的 .codex/config.yaml

项目指令

```
# AGENTS.md (in project root)
- Run tests with: uv run pytest
- Use ruff for linting
- Never modify migration files
```

常用配置项

model 使用的模型 (如 o4-mini、o3)

approval_mode 默认审批模式

providers API provider 配置 (密钥、base URL)

history 保存对话历史: true / false

沙箱

工作原理

Codex 在沙箱环境中运行命令以防止意外副作用。默认禁用网络访问。文件写入限制在项目目录内。

沙箱选项

macOS Apple Seatbelt (默认, 内置)

Linux 基于 Docker 的沙箱

--full-auto 需要启用沙箱

--dangerously-auto-approve 跳过沙箱 (不推荐)

技巧

高效提示

具体描述 "在 fetch_data() 中添加重试逻辑" > "改进代码"

引用文件 "修复 src/auth.py 中的 bug" 缩小范围

说明约束 "不要修改公共 API" 设定边界

迭代优化 在同一会话中继续追问细化

工作流模式

```
# Explore -> plan -> execute
codex "explain the auth module"
codex "plan how to add OAuth support"
codex --approval-mode auto-edit "add OAuth"

# Review before commit
git add -A
codex review
```

常用参数

--model, -m 本次会话覆盖模型

--approval-mode 设置审批模式

--quiet, -q 最小化输出

--no-project-doc 忽略 AGENTS.md