AI CLI

面向 Windows PowerShell 与 Git Bash 的安装与使用指南

2025 年 11 月 9 日

1 准备工作

- **Node.js：** 前往 nodejs.org/en/download 下载并安装（LTS 版本更稳）。
- **PowerShell 7:** 前往 PowerShell Releases，建议安装包：`PowerShell-7.5.4-win-x64.msi`。
- **Git Bash（Git for Windows）：** 前往 git-scm.com/install/windows 安装（含 Git Bash 终端）。

安装完成后，打开终端（PowerShell 或 Git Bash）检查版本：

```
node -v
npm -v
```

2 安装与更新：Codex、Claude Code、Gemini CLI

2.1 全局安装（PowerShell 或 Git Bash 通用）

Codex	`npm install -g @openai/codex`
Claude Code	`npm install -g @anthropic-ai/claude-code`
Gemini CLI	`npm install -g @google/gemini-cli`

2.2 Codex 更新

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

3 在 PowerShell 7 中运行

常用命令（命令名以安装输出为准，可用 `--help` 检查）：

```
# 启动交互式 TUI
codex
claude
gemini

# 查看帮助（示例）
codex --help
claude --help
gemini --help
```

4 在 Git Bash 中运行

4.1 基本用法

安装完成后，直接在 Git Bash 中执行：

```
codex
claude
gemini
```

4.2 PATH 提示（如遇命令找不到）

npm 全局可执行通常位于 Windows 的 `%USERPROFILE%\AppData\Roaming\npm`。在 Git Bash 可将其加入 `~/.bashrc`：

```
echo 'export PATH="$PATH:/c/Users/<你的用户名>/AppData/Roaming/npm"' >> ~/.bashrc
source ~/.bashrc
```

4.3 交互/TTY 兼容性

若在 Git Bash 出现交互异常（如无法输入、界面错乱），可尝试：

```
winpty codex
```

或改用 WSL2（附录有链接）。

5 终端代理（Clash）

本文以 **Clash 的 HTTP 代理端口**为 127.0.0.1:7890 为例（如你在 Clash 中自定义了端口，请按实际配置替换）。

5.1 PowerShell：临时为当前会话启用

仅影响**当前 PowerShell 会话**与其子进程：

```
# 设置 HTTP/HTTPS 代理（当前会话）
$env:HTTP_PROXY = "http://127.0.0.1:7890"
$env:HTTPS_PROXY = "http://127.0.0.1:7890"

# （可选）不走代理的地址；用逗号或分号分隔
$env:NO_PROXY = "localhost,127.0.0.1"

# 验证（可查看变量是否生效）
[Environment]::GetEnvironmentVariable("HTTP_PROXY","Process")
[Environment]::GetEnvironmentVariable("HTTPS_PROXY","Process")
```

关闭（仅当前会话）：

```
Remove-Item Env:HTTP_PROXY -ErrorAction SilentlyContinue
Remove-Item Env:HTTPS_PROXY -ErrorAction SilentlyContinue
Remove-Item Env:NO_PROXY -ErrorAction SilentlyContinue
```

5.2 PowerShell：长期使用（用户级环境变量）

对当前用户持久生效；**需新开终端**才会加载到新进程：

```
# 写入用户级环境变量（持久化）
setx HTTP_PROXY "http://127.0.0.1:7890"
setx HTTPS_PROXY "http://127.0.0.1:7890"
setx NO_PROXY "localhost;127.0.0.1"

# 说明：setx 不影响“当前已打开”的会话；请重新打开 PowerShell
```

还原（清空持久变量）：

```
setx HTTP_PROXY ""
setx HTTPS_PROXY ""
setx NO_PROXY ""
```

5.3 Git Bash：临时与持久配置

一次性（当前 shell）

```
export HTTP_PROXY="http://127.0.0.1:7890"
export HTTPS_PROXY="http://127.0.0.1:7890"
export NO_PROXY="localhost,127.0.0.1"
```

持久化（写入 /.bashrc）

```
echo 'export HTTP_PROXY="http://127.0.0.1:7890"' >> ~/.bashrc
echo 'export HTTPS_PROXY="http://127.0.0.1:7890"' >> ~/.bashrc
echo 'export NO_PROXY="localhost,127.0.0.1"' >> ~/.bashrc
source ~/.bashrc
```

兼容性提示 部分工具仅识别小写变量，可**同时**设置：

```
export http_proxy="$HTTP_PROXY"
export https_proxy="$HTTPS_PROXY"
export no_proxy="$NO_PROXY"
```

5.4 与 npm / Git 的常见搭配

npm 级代理（覆盖环境变量）

```
npm config set proxy "http://127.0.0.1:7890"
npm config set https-proxy "http://127.0.0.1:7890"

# 取消
npm config delete proxy
npm config delete https-proxy
```

Git 的全局代理（可选）

```
git config --global http.proxy http://127.0.0.1:7890
git config --global https.proxy http://127.0.0.1:7890

# 取消
git config --global --unset http.proxy
git config --global --unset https.proxy
```

WinHTTP（系统级，少数服务使用，可选）

```
# 设定
netsh winhttp set proxy 127.0.0.1:7890
# 还原
netsh winhttp reset proxy
```

6 快速验证

```
# 版本/诊断（不同 CLI 略有差异）
codex --version
claude --help
gemini --version

# npm 走代理的小测试（能否获取包信息）
npm view @openai/codex version
```

7 Codex 常用斜杠命令

进入 Codex 交互界面后（执行 `codex`），可直接输入斜杠命令。

7.1 /model：选择模型与推理强度

- **切换方式:**在会话中输入 `/model` 按提示选择;或启动时使用 `codex --model gpt-5-codex`。
- **常见选项:**
  - `gpt-5-codex`：为代码智能体优化，仓库理解与工具使用更强。
  - `gpt-5-codex-mini`：更经济，能力略低但配额更友好。
- **推理强度:** 可在 `/model` 菜单中选择 *low/medium/high* 等级；任务复杂度高时再提升。

7.2 /approvals：审批模式（安全权限）

- **Auto:** 默认。允许在当前目录读写与运行必要命令；涉及更敏感操作时仍会请求确认。
- **Read Only:** 只读。查看文件与提出修改建议，不落地改动或执行命令，直至你批准。
- **Full Access:** 更高权限（含更广本机/网络访问）。仅在信任仓库与任务时短时启用。

8 常见问题与建议

- **Windows 体验:** 若遇兼容性问题，优先考虑在 **WSL2** 中运行 Codex（Linux 环境更稳定）。
- **Git Bash:** TTY 异常可用 `winpty` 前缀；或直接改用 PowerShell/WSL。
- **代理优先级:** 通常为「**CLI 专用配置（如 npm config）> 进程环境变量 > 系统级代理**」。排查时从上到下逐级检查。

附录：下载与参考链接（点击可跳转）

- Node.js：https://nodejs.org/en/download
- PowerShell 7 Releases：https://github.com/PowerShell/PowerShell/releases
- Git for Windows（含 Git Bash）：https://git-scm.com/install/windows
- Codex CLI（概览/安装/升级）：https://developers.openai.com/codex/cli