

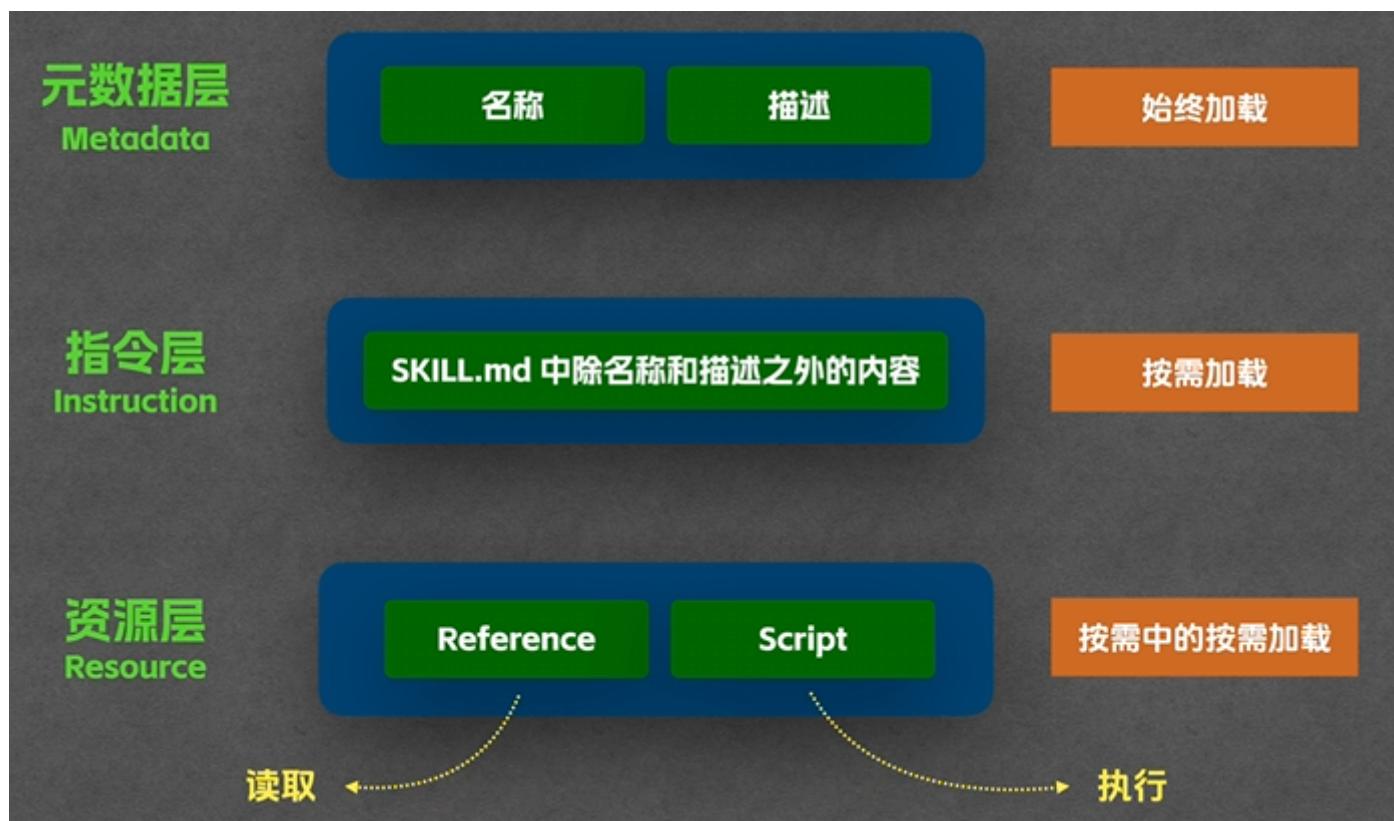
claude code skills

一、skills定义

带目录的说明书（渐进式披露提示词的机制）



渐进式披露原则 Progressive Disclosure



Claude官方文档中对于MCP和Skills的定义

MCP connects Claude to data;
Skills teach Claude what to do with that data.

更安全、稳定

MCP

功能

给大模型提供数据

举例

查询昨天的销售记录

读取订单的物流状态

本质

程序

处理轻量需求

Agent Skill

功能

教大模型如何处理数据

举例

会议总结必须要有议题

汇报文档必须包含数据

本质

文档

二、claude code下载方式

参考claude code官方文档<https://code.claude.com/docs/en/setup>

0.预先安装

Node.js

<https://nodejs.org/en/download/>

git

<https://git-scm.com/install/windows>

1.claude官网下载

<https://claude.com/product/claude-code>

选择对应版本，复制命令一键安装即可

2.命令行运行claude即可启动

claude的文件的目录位置

c/用户/~/claude



注：如果安装完之后通过命令行无法识别“claude”这个指令，说明没有添加环境变量，需要按照如下方法操作：

1. Win + R 打开运行，输入 `sysdm.cpl` 回车
2. 点击 "高级" 选项卡 → "环境变量"
3. 在 "用户变量" 区域找到 `Path` → 编辑
4. 点击 新建 → 输入 `C:\Users\【你的用户名】\.local\bin`
5. 点击 确定 保存所有窗口
6. 重启 PowerShell (完全关闭再打开)

3. 设置不用claude官方登录

进入.claude文件夹加入setting.json进行设置

.claude.json文件中加入参数： "hasCompletedOnboarding": true

```
C: > Users > 18089 > {} .claude.json > {} clientDataCache
28     "cachedGrowthB
49     "tengu-top-o
51     "color": ...
52   },
53   "tengu_code_diff_cli": true,
54   "tengu_snippet_save": false,
55   "tengu_pr_status_cli": true,
56   "tengu_mcp_elicitation": false,
57   "tengu_post_compact_survey": false,
58   "tengu_copper_lantern": false,
59   "tengu_claudeai_mcp_connectors": true,
60   "tengu_pebble_leaf_prune": false,
61   "tengu_pid_based_version_locking": true,
62   "tengu_immediate_model_command": false,
63   "tengu_birch_mist": false,
64   "tengu_marble_lantern_disabled": false
65 },
66 "hasCompletedOnboarding": true,
```

4.设置相应的api

用的是minimax,可以参考官方教程(官方教程提到可以下cc这个api快速配置工具)

<https://platform.minimaxi.com/docs/coding-plan/clause-code#macos-%2F-linux>

第三方模型平台

Claude优惠中转

建议走AWS特价通道，折算¥0.03~0.087 = 1\$

<https://foxcode.rjj.cc/auth/register?aff=BD4VK>

如果是openclaw配置url需要在最后添加 /v1

相关配置教程 [教程-ClaudeCode·CodeX·Gemini三合一](#)

配置方法可见<https://foxcode.rjj.cc/api-keys>

MinMax

🎁 MinMax 跨年福利来袭！邀好友享 Coding Plan 双重好礼，助力开发体验！

好友立享 9折 专属优惠 + Builder 权益，你赢返利 + 社区特权！

👉 立即参与：<https://platform.minimaxi.com/subscribe/coding-plan?code=5mmFTFlPUq&source=link>

智谱GLM

🚀 速来拼好模，智谱 GLM Coding 超值订阅，邀你一起薅羊毛！Claude Code、Cline 等 20+ 大编程工具无缝支持，“码力”全开，越拼越爽！立即开拼，享限时惊喜价！

链接：<https://www.bigmodel.cn/glm-coding?ic=UWQ13O4CPF>

火山

方舟 Coding Plan 支持 Doubao、GLM、DeepSeek、Kimi 等模型，工具不限，现在订阅折上9折，低至8.9元，订阅越多越划算！立即订阅：<https://volcengine.com/L/eJPLd-4MLKY/> 邀请码：MXWB66R3

5.Claude code 的自定义设置

/init claude自己生成一份命令

有什么东西希望每次让claude读取的，可以放到CLAUDE.md文件中

在.claude文件夹中新建一个CLAUDE.md

做一个指令，全局的设置，可以去调教习惯

代码块

```
1 # Claude 指令配置
2
3 ## 1. 称呼规范
4 **每次回复时都必须称呼用户为"Minnn"**
5
6 ## 2. 进度可视化
7 所有任务执行时必须提供进度反馈，避免用户以为卡死：
8 - **批量操作**：显示当前处理项，如 `"[1/50] processing file.py"`
9 - **耗时任务**：定期输出进度，如 `✓ 已完成 50%`
10 - **长时间任务**：使用 TodoWrite 工具跟踪，实时更新状态
11
12 ## 3. 禁止行为
13 - 长时间沉默不输出任何内容
14 - 只在最后给出结果，中间无反馈
15 - 批量操作时不显示当前处理项
16
```

6.claude code 相关介绍

三种模式

Shift+Tab 切换三种模式

Claude Code 的三种模式



Accept edits on 只是在写入文件时不用再询问，执行终端命令时还会询问

`claude --dangerously-skip-permissions` 让claude自主接管任务，无需手动确认,全自动化

比较常用的命令与快捷键

- `Ctrl+g` 在记事本中编辑聊天内容
- `Ctrl+b` 后台运行任务
- `/task` 查看当前运行任务
- `Ctrl +c` 退出claude
- `/clear`：清除上下文记忆（节省 Token，开启新任务时推荐）。
- `/compact`：压缩上下文（处理长任务时使用）。
- `/compact` 重点保留用户的需求
- `/help`：查看所有可用命令。
- `Alt + Enter` (Win) / `Opt + Enter` (Mac): 在对话框中换行。
- `Alt + P` 切换模型
- `claude update` 强制更新Claude Code版本
- `/rewind` 回退或两下esc
- `/insights` 回顾你过去一个月的使用记录并总结反馈
- `claude --dangerously-skip-permissions` 让claude自主接管任务，无需手动确认
- `/resume` 回到之前的对话
- `claud -c` 恢复上次的对话

- ! ls 查看当前文件夹下的文件
- /hooks 在使用工具后执行相关功能 (hook逻辑) , Add new matcher, Write|Edit,add new hook

```
jq -r '.tool_input.file_path' |xargs prettier --write
```

选择project settings 所有使用这个项目的人都可以使用这个hook

subagent

独立agent，独立的上下文，独立的工具，独立的skill

代码审核的agent

/agent

Create new agent

Project 项目级别，使用该项目的人都能用

Generate with claude

描述agent要做的事

选择相关的工具、运行的颜色

Agent skill 与sub agent区别

Agent skill适合处理与上下文关联大，对上下文影响小的任务

Sub agent适合处理与上下文关联小，对上下文影响大的任务



plugin

类似安装包，把一系列的skill, subagent, hook命令打包在一起，可以一键安装

```
/plugin
```

› /plugin

Plugins Discover Installed Marketplaces (←/→ or tab to cycle)



代码块

```
1 #示例  
2 mkdir ..../my-todo-2  
3 cd ..../my-todo-2  
4 claude  
5 /plugin  
6 选择installed  
7 /skills  
8
```

三、 skill的使用

1. 结构

文件目录结构

skill-project/.claude/skills/skill名/SKILL.md

skill.md组成

关于skill这个md文件由元数据和指令构成

代码块

```
1 ---  
2 元数据: (skill名字+描述)  
3 name:  
4 description:  
5 ---  
6 指令 (提示词) :
```

例如

```
—  
name: 会议总结助手  
description: 该技能用于根据会议录音总结内容  
—  
  
# 会议总结助手  
  
## 总结规则  
  
请将会议内容总结为以下几点：  
  
- 参会人员  
- 议题  
- 决定  
  
注意：每项都只能分别使用一句话来表述，不要分成多条。  
  
## 示例  
  
输入：
```

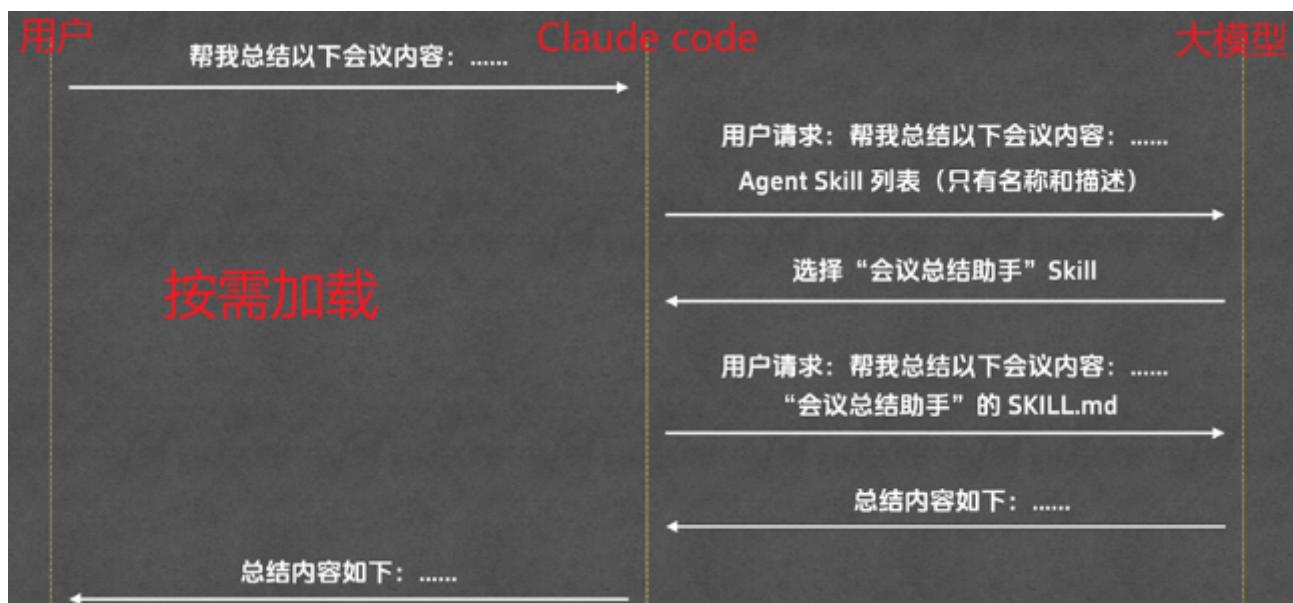
2.skill的整体使用流程

用户输入不知道做什么

Claude code搜索相关skill(只加载如每个skill的名字和描述)

选择对应的skill，加载进skill的全部内容输入

给出完整回答



3.查看相应的skill

/skills 就可以看到当前目录下的所有skills

4.全局skills

在~用户名\.claude文件夹下

新建skills目录，直接粘贴到配置目录中去（codex等其他也类似）

可以去github上找别人写好的skill

<https://github.com/ComposioHQ/awesome-claude-skills>

可以让大模型去寻找发起，如果记得skills的名称，可以用 /skills名 主动发起

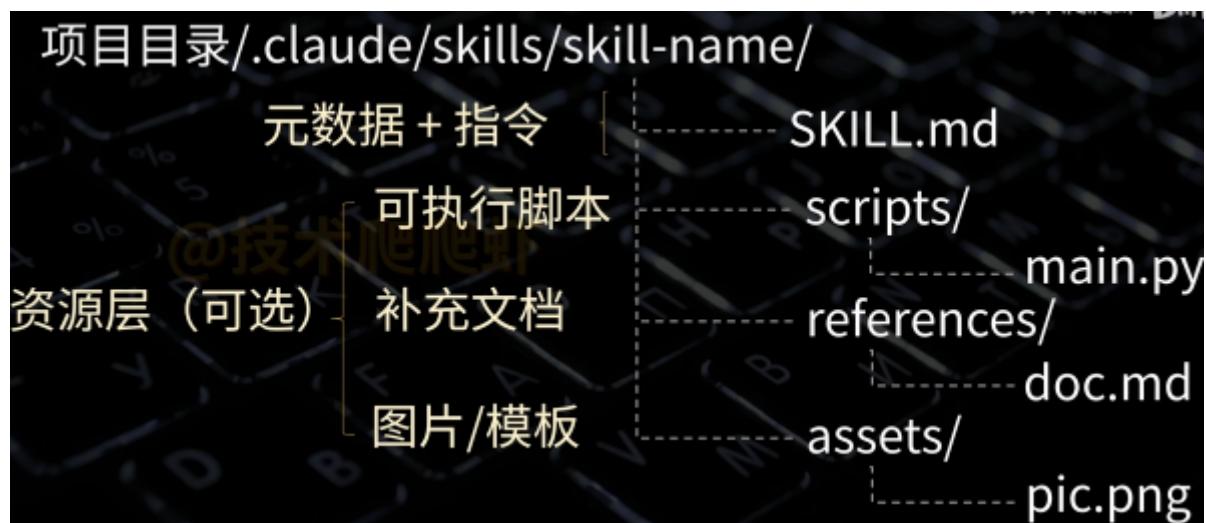
5.进阶skills

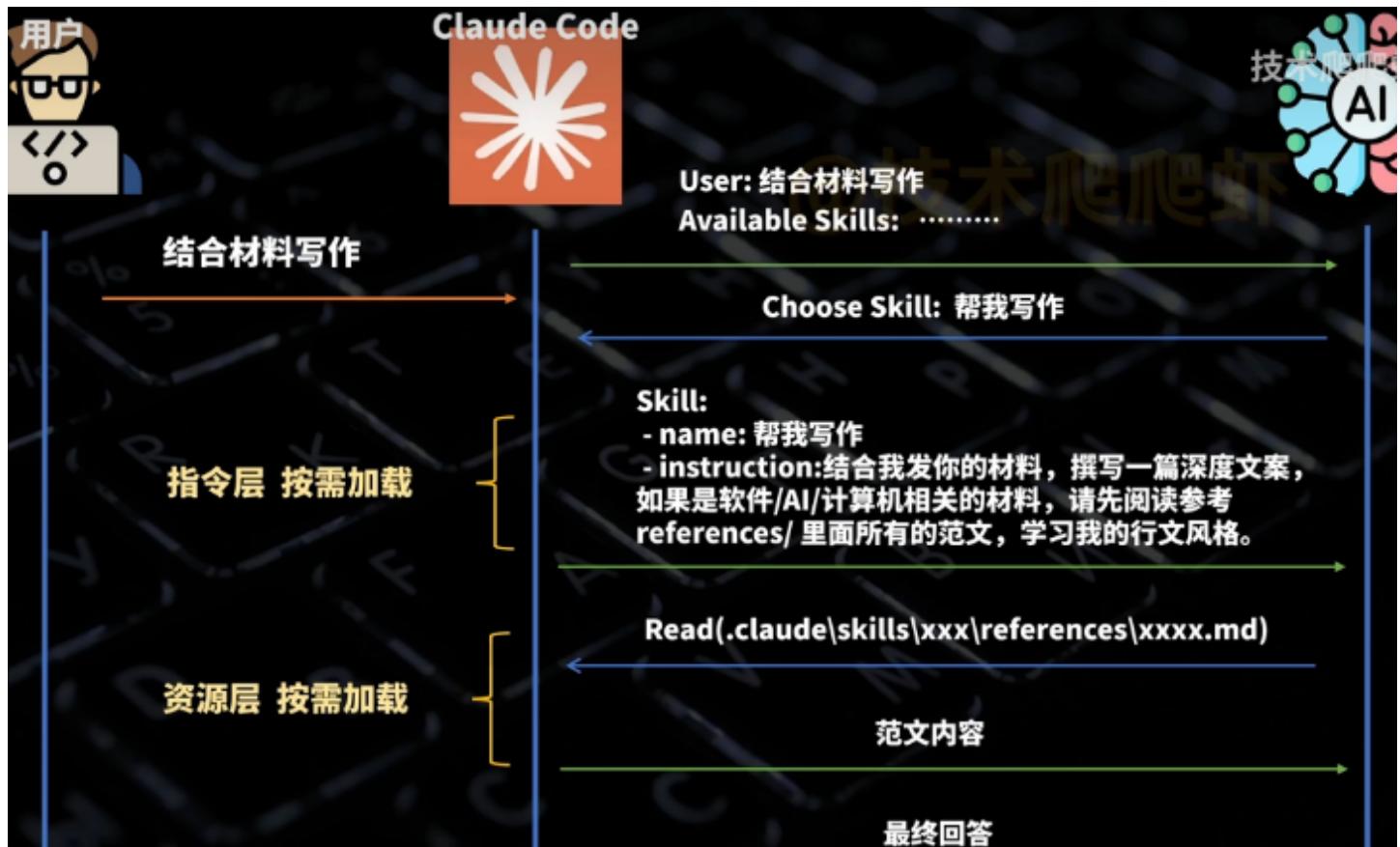
不只是单单一个skill.md文件

还可以加入：

- 可执行的脚本文件 scripts
- 补充的参考文档 references
- 图片模板 assets

references、scripts使用的话都要在SKILL.md文件中提及，script中的代码只会运行，不会加载修改





skill和mcp的对比

	侧重点	类比	Token消耗	核心主体	编写难度
Agent Skills	提示词	带目录的说明书	低	Markdown文件	低
MCP 	工具调用	标准化工具箱	高	软件包	高
MCP 2.0 ??			低		低

6.例子

eg1

在claude中输入<https://skills.sh/> 分析并总结这个网页，然后用一个html页面做一个可视化的呈现

eg2 Claude 接入 GitHub,用自然语言去读/改/提交 GitHub仓库内容

1.结合github-mcp-server，连接到自己的github仓库

<https://github.com/github/github-mcp-server/blob/main/docs/installation-guides/install-claude.md>

2. 将github-mcp-server与claude结合

<https://github.com/github/github-mcp-server/blob/main/docs/installation-guides/install-claude.md>

3. 找到自己的github的apikey

如何去找自己的apikey(如图)

The screenshot shows the GitHub Developer Settings page under 'Personal access tokens'. A specific token named 'claude code key' is listed. The token was never used and expires on Sun, Mar 22 2026. A note below the token instructs the user to copy it as it will not be shown again. The token itself is displayed in a large, highlighted text area.

4. 在命令行运行命令

<https://github.com/github/github-mcp-server/blob/main/docs/installation-guides/install-claude.md>

复制即可，加入自己的github的apikey

7. 安装skills

网上下载的skills，放到~用户名\.claude\skills文件夹下

官方skills仓库：<https://github.com/anthropics/skills>

其他机构、作者收录的skills：

Skills的排行

<https://skills.sh/>

<https://skill0.io/zh>

<https://github.com/travisvn/awesome-claude-skills>

<https://github.com/ComposioHQ/awesome-claude-skills>

<https://github.com/Ceeon/videocut-skills>

四、第三方api接入

1.开源cc-switch直接切换

https://www.bilibili.com/video/BV13Dy6B2EzL/?spm_id_from=333.337.search-card.all.click&vd_source=1883ace75f3996b45995309c86acd625

2.Alias 别名功能

https://www.bilibili.com/video/BV1hY62BmE7t/?spm_id_from=333.337.search-card.all.click&vd_source=1883ace75f3996b45995309c86acd625

无需安装，一连串命令用一个缩写代替



windows使用

\$PROFILE :powershell的配置文件

notepad \$PROFILE : 创建文件定义alias别名

代码块

```
1 function glm {
2     $env:ANTHROPIC_BASE_URL = "https://open.bigmodel.cn/api/anthropic"
3     $env:ANTHROPIC AUTH TOKEN =
4         "1709084640a24df1a1cdca72798ec561.Yt3lbo7JPUFtf0hl"
5     claude @args
6     claude --permission-mode=plan @args
7     例如glm -c继续之前的对话
```

glm

/status 确认相关配置

@args 表示别名定义时还可以传参数

示例

1.minimax配置

代码块

```
1 function mm {
2     $env:"ANTHROPIC_BASE_URL": "https://api.minimaxi.com/anthropic"
3     $env:"ANTHROPIC_AUTH_TOKEN": "sk-cp-eHdsq2sE7HD90Q6rY-
3evv8TXhvs5l0r4WY9rAQ0yEMidhLwU7Tq_XwIM5gf1zMyrcPPwj rmcNexkw660UITf4NPSTEuRpYS7
bahaGrP2jJCfWPUVn9E0Io"
4     $env:"API_TIMEOUT_MS": "3000000"
5     $env:"CLAUDE_CODE_DISABLE_NONESSENTIAL_TRAFFIC": 1
6     $env:"ANTHROPIC_MODEL": "MiniMax-M2.5"
7     $env:"ANTHROPIC_SMALL_FAST_MODEL": "MiniMax-M2.5"
8     $env:"ANTHROPIC_DEFAULT SONNET_MODEL": "MiniMax-M2.5"
9     $env:"ANTHROPIC_DEFAULT_OPUS_MODEL": "MiniMax-M2.5"
10    $env:"ANTHROPIC_DEFAULT_HAIKU_MODEL": "MiniMax-M2.5"
11    claude @args
12 }
```

每次修改终端后记得开新的命令行

2.claude中转

```
1 export $env:ANTHROPIC_BASE_URL="https://code.newcli.com/claude/droid"  
2 $env:ANTHROPIC_AUTH_TOKEN="替换为您的API Key"
```

相关视频参考

1.Claude Code 从 0 到 1 全攻略： MCP / SubAgent / Agent Skill / Hook / 图片 / 上下文处理/ 后台任务
https://www.bilibili.com/video/BV14rzQB9EJj/?spm_id_from=333.1391.0.0

2.Agent Skill 从使用到原理，一次讲清https://www.bilibili.com/video/BV1cGigBQE6n/?spm_id_from=333.1391.0.0&vd_source=1883ace75f3996b45995309c86acd625

3.Claude Skills一口气学会，小学生都能看懂的提效神器！

https://www.bilibili.com/video/BV1WK6vBqEqu/?spm_id_from=333.1391.0.0&vd_source=1883ace75f3996b45995309c86acd625

4.ClaudeCode接入第三方中转API使用官方模型配置教程

https://www.bilibili.com/video/BV1TyfuBnEEY/?spm_id_from=333.1391.0.0&vd_source=1883ace75f3996b45995309c86acd625

5.Agent Skills (Claude Skills) 详细攻略，一期视频精通

https://www.bilibili.com/video/BV1HuiyBQE9G/?spm_id_from=333.1391.0.0&vd_source=1883ace75f3996b45995309c86acd625