

EP-02 · 上海夜场 · NIGHT EDITION

手工川 Agent Skill 简明教程

手把手从 0 到 1 开发一个属于自己的 Skill

AI 大夜校 · 道法术器 + Agent Skill 实战工坊



讲师 · 手工川

LovStudio 创始人

2026-04-30 · 上海陆家嘴

19:00 — 23:00 · 4 HOURS HANDS-ON

v0.2

4 小时 / 50 页 / 6 模块

每个核心概念 = 1 页提问 + 1 页揭答 — 留出讨论时间



M0 开场

· 15 min ·
4 页



**M1 先看
(生态)**

50 min · 10 页



**M2 先用
(装)**

35 min · 6 页



**M3 再写
(手搓)**

50 min · 8 页



**M4 再升级
(creator)**

40 min · 6 页



**M5 再评估
(5DQM)**

45 min · 8 页



**M6 再发布
(变现)**

30 min · 6 页



**收尾 (Final
Share)**

25 min · 2 页

⚠ QUESTION · 现场举手 · 1 MIN

现在请举手 ↑

- ① 用过 Claude Code 的请举手
- ② 装过任意 skill 的请举手
- ③ 自己写过 skill 的请举手

看一下你周围 — 这 4 小时我们的目标，是让所有人都能写出第三个

开场 · 承诺

今天结束时，你能做到 →

01



会装会用

能在自己电脑上装一个现成 skill 并跑出结果

02



能改能写

能从 0 写一个属于自己的 skill，并用 lovstudio-skill-creator 升级

03



能发能赚

能把 skill 发布到 marketplace，并知道怎么变现

⚠ QUESTION · 讨论 1 MIN

Skill 是什么？

用你自己的话说出来 — 不要看手机

和身边的人聊 1 分钟，等会对答案

ANSWER ←

Skill = 给 Agent 的「工作手册」

一份让 AI 知道"什么时候做什么、按什么顺序、用哪些工具"的标准化文件包



本质上：你日常工作里反复做的事，AI 也能照着做

⚠ QUESTION · 现场举例 · 2 MIN

你听过 / 见过哪些超牛的 skill ?

随便说 — 任何让你觉得"卧槽这也能做"的都算

朋友用 X 做了 Y

我自己想做但还没做的 X

完全不知道是不是 skill
但很神奇的 X

等会我会展示我自己最 PROUD 的两个

ANSWER ←

我自己最 PROUD 的两个 →

100+ 页

精美 PDF 排版 Skill

用一个 skill 生成 100+ 页 Claude Code 系统报告，自动排版、目录、页眉页脚

手工川 · 公众号 · 2026

47 min

跑通 O'Reilly 出版级 Skill 写书工作流

47 分钟从空白到出版级章节，含模板、风格、章节联动

手工川 · 公众号 · 2026

"这两个 skill 都不是手搓的 — 是 creator 生成的"

QUESTION · 讨论 1 MIN

Plugin 和 Skill — 一个东西吗？ 是同一个东西吗？

先猜一猜 — 不用看资料

我装的是 plugin
还是 skill？

一个 plugin 里能
有几个 skill？

它们是 1:1 的吗？

ANSWER ←

Plugin = 容器 · Skill = 单元

Plugin · 容器

分发 + 安装单位

commands

hooks

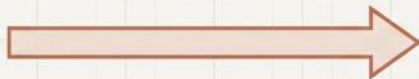
agents

skills (多个)

settings

...

1 个 Plugin → 可以装上 1 ~ N 个 Skills



Skill · 单一能力

any2pdf

image-gen

deep-research

fill-web-form

维度	Plugin	Skill
角色	容器 / 包	能力 / 单元
安装	<code>`/plugin install foo`</code>	通常随 plugin 自动装
包含	多个 skill + 配套	单一 SKILL.md
类比	npm package	一个 function

你装一个 plugin, 可能就装上了 5 个 skill

claude.com/plugins · 跨场景明星

在 Claude Code 里输入 `/plugins` 也能看到这个有排序的列表



DEVELOPER

frontend-design

前端 UI 设计与组件生成



DEVELOPER

context7

为代码自动注入文档上下文



META

skill-creator

描述需求 → 自动生成 skill 骨架



AUTOMATION

ralph-loop

长流程 / loop 类工作流自动化



PRODUCT

product-management

PRD / 路线图 / OKR 全套 PM



LEGAL

legal

合同审阅 / 条款分析 / 法务助理

每出一个新 plugin，对应行业都要抖一抖 — 监控 Anthropic Twitter / Blog 是必修课

TAG MODULE 1 · 中文生态

除了 claude.com/plugins, 还有这些 →

①

OPEN SOURCE

[anthropics/skills](https://github.com/anthropics/skills)

Anthropic 官方开源 skill 仓库 (GitHub)



②

DISCOVERY

skillsmp.com

社区 skill 排行榜与发现



③

CN MARKET

[Ich-skill-store.lovstudio.ai](https://lovstudio.ai/skill-store)

lovstudio 中文 skill 交易市场, 含付费 skill



④

MONITOR

写一个 monitor

跟踪 Anthropic Twitter / Blog / GitHub,
每出新 plugin 第一时间知晓



带电脑的同学现在就可以扫码访问

⚠ QUESTION · 讨论 1 MIN

prompt 也能用， 为啥还要写 skill?

从你自己的工作场景想一想

我每次写同一个 prompt 烦不烦?

我换电脑 / 换团队成员，prompt 怎么传?

prompt 写得好，下次还记得吗?

Prompt 是耗材 · Skill 是资产

维度	Prompt (每次粘贴)	Skill (一次写好)
触发	主动想起、复制粘贴 	AI 自动识别需要时调用 
复用	一次性 	永久复用 
协作	个人剪贴板 	Git 版本控制 + PR review 
Token	每次 ~500-2000 	命中才加载 ~500 
演进	散落各处难维护 	集中管理 + CHANGELOG 

prompt 烧掉就没了, skill 越用越值钱

MODULE 2 · 先用

你需要装这些工具



01

Claude Code

版本: $\geq 0.X$

官网: claude.com/claude-code



02

Terminal

macOS / Windows / Linux

基础命令会用即可



03

编辑器

看下一页对比

拿不准就装齐 — 这 4 小时我们都会用上

⚠ QUESTION · 现场分享 · 1 MIN

你平时用什么编辑器？

写代码用什么、写文档用什么 — 同样吗？

→ 你的笔记用什么写？

→ 你的代码用什么写？

→ 你写过 markdown 吗？

两种心智 → 两类编辑器

VS Code · 写代码心智

📄 适合: **scripts/** 子目录、**json**、**ts** 等

✓ 优势: **扩展生态强**、类型提示

🧠 心智: 把每个文件当**代码**

Obsidian · 写知识心智

📄 适合: **SKILL.md**、**reference/** 长文档

🔗 优势: **双向链接**、图谱、Markdown 渲染好看

🧠 心智: 把每个文件当**笔记节点**

我自己综合使用 VS Code · Obsidian · Typora · Vim —— 不同心智用不同工具
—— 手工川

“Skill 既是代码也是知识 —— 两个都装上没毛病”

▲ HANDS-ON 1 · 5 MIN

装 + 跑 lovstudio-any2pdf

01 · 装

```
/plugin marketplace add lovstudio/skills  
/plugin install lovstudio-any2pdf
```

02 · 跑

打开 Claude Code, 说:
> 把 README.md 转成精美 PDF

03 · 看

- 自动排版 / 自动目录 / 自动页脚 / 输出到 ./output/

装不上的同学举手 · 现场我们一起 debug

MODULE 2 · 拆解

刚才那个 skill 长什么样？

```

1  ---
2  name: lovstudio-any2pdf
3  description: 把 markdown / 网页 / 图片转成精美 PDF。
4    当用户说"转 PDF""导出 PDF""做成报告"时调用。
5  ---
6
7  # Any2PDF
8
9  ## 何时使用此 skill
10  当用户提供 markdown / URL / 图片, 并希望输出 PDF 时...
11
12  ## 工作流
13  1. 解析输入格式
14  2. 调用 scripts/render.py
15  3. 输出到 ./output/
    
```

① **frontmatter**: YAML 元数据 (name + description)

② **instructions**: 人话写的工作流

③ **files**: 附属脚本与参考 (这里 scripts/render.py)

SKILL.md = 给 AI 看的工作手册

CHECKPOINT

你装上了吗？

✓ 成功状态卡

- ✓ 输入 ``/plugin list`` 能看到 `lovstudio-any2pdf`
- ✓ 在 Claude Code 里说"转 PDF"会触发
- ✓ 输出文件出现在 `./output/`

✗ 5 个常见失败原因

- ✗ 1. 网络问题 — 代理 / 镜像设置
- ✗ 2. Claude Code 版本太旧 — 升到 $\geq$ 最新
- ✗ 3. marketplace 没加 — 先 add `lovstudio/skills`
- ✗ 4. 拼写错误 — skill 名 `lovstudio-any2pdf` 不是 `any-pdf`
- ✗ 5. 没重启 — 装完重启 Claude Code

装不上 → 举手 → 现场看屏幕

QUESTION · 讨论 1 MIN

一个 skill 至少要写哪几样东西？

刚才你看了 any2pdf 一印象里有几块？

没有它就跑不起来的是？



让 AI 知道何时调用的是？



让 AI 知道怎么做的是？



SKILL.md 三段式

① frontmatter

YAML 元数据 — AI 看到的“广告语”

- name (必填)
- description (必填) — 触发器
- 其他可选



frontmatter 决定
能不能被触发

② instructions

正文 markdown — 写给 AI 看的工作流

- 何时使用 (What & When)
- 怎么做 (How)
- 边界 (Don'ts)



instructions 决定 做得对不对

③ files

附属文件 — 按需加载

- scripts/ 确定性强的代码
- references/ 长文档、图表、字典
- assets/ 图片、模板



files 决定 做得快不快

QUESTION · 现场写 · 3 MIN

你的 skill 是「自动填表」 ←

现在用 30 个字以内，写一条 description

description: _____

等会我们对答案 — 看看你的 AI 会不会调用

description 是 skill 的「触发器」

写不好就永远不会被调用

✕ 我的 PDF 工具

✓ 把 markdown / 网页 / 图片转成精美 PDF。当用户说"转 PDF"导出 PDF"做成报告"时调用

✕ 处理表单数据

✓ 自动填写网页表单。当用户提供表单 URL + 字段值时调用，支持 Google Form、问卷星

✕ 写文档的

✓ 生成技术文档与 README。当用户说"写 README"补充文档"时使用，自动从代码推断

✕ deep research

✓ 深度研究并产出带引用的报告。当用户问"调研 X"分析 Y 对比"时启动多轮搜索

✕ agent skill 创建

✓ 从用户描述生成完整 skill 骨架。当用户说"帮我做一个 skill"新建一个 skill"时使用

description 写给 AI 看，但要用人话

MODULE 3 · 规范

命名 + 目录 + scope

01 · 用户级

`~/.claude/skills/` ←

适用：你个人的工作 SOP

特点：所有项目共享

02 · 项目级

`./.claude/skills/` ←

适用：跟某个 repo 强绑定

特点：跟代码一起 commit

03 · 团队级 / marketplace

通过 plugin install ←

适用：团队共用 / 商业销售

特点：版本管理 + 升级通知

命名规范

- ✓ 小写 + 连字符 (kebab-case)
- ✓ 不超过 30 字符
- ✓ 体现领域 + 动作 (lovstudio-any2pdf 而非 my-pdf-tool)
- ✓ 避免泛化名 (helper、utils、tool)

⚠ HANDS-ON 2 · 8 MIN

手写你的第一个 skill

Step 1 · 创建文件

```
mkdir -p ~/.claude/skills/hello-yourname
cd ~/.claude/skills/hello-yourname
touch SKILL.md
```

Step 3 · 重启并测试

重启 Claude Code
> 打个招呼吧

Step 2 · 填三段式

```
---
name: hello-yourname
description: 自我介绍并打印当前时间。当用户说"自我介绍"
              "你是谁""hello"时调用。
---
```

Hello YourName

当被调用时:

1. 输出 "你好, 我是 [你的名字]"
2. 用 date 命令打印当前时间
3. 加一句你今天的心情

替换 `yourname` 为你自己的名字 (避免冲突)

MODULE 3 · 排错

调试三件套

01 reload



- 什么：让 Claude Code 重新读 skill
- 何时：改完 SKILL.md 没生效
- 怎么：重启 Claude Code（最稳）

02 list



- 什么：看所有装上的 skill
- 何时：怀疑没装上 / 名字错
- 怎么：在 Claude Code 输入 /plugin list

03 直接读



- 什么：用编辑器打开 SKILL.md
- 何时：description 似乎不触发
- 怎么：用 VS Code / Obsidian 检查格式

99% 的 skill 不工作 一是 **description** 写得太抽象

TAG : QUESTION · 现场排查 · 3 MIN

你的 hello skill 触发了吗？

没触发的同学 — 在你猜的那个原因下面打钩

1. ☐ 我没重启 Claude Code
2. ☐ description 太抽象 / 没触发关键词
3. ☐ name 写错了 / 路径放错了
4. ☐ frontmatter YAML 格式错 (少了 ---)
5. ☐ 我说的话和触发关键词不匹配

打完钩举手 — 我们看哪个最常见

▲ QUESTION · 写下来 · 2 MIN

你工作里最烦的 重复任务 是什么？

写下 3 个 — 不要思考‘它能不能写成 skill’，先写下来

1. _____

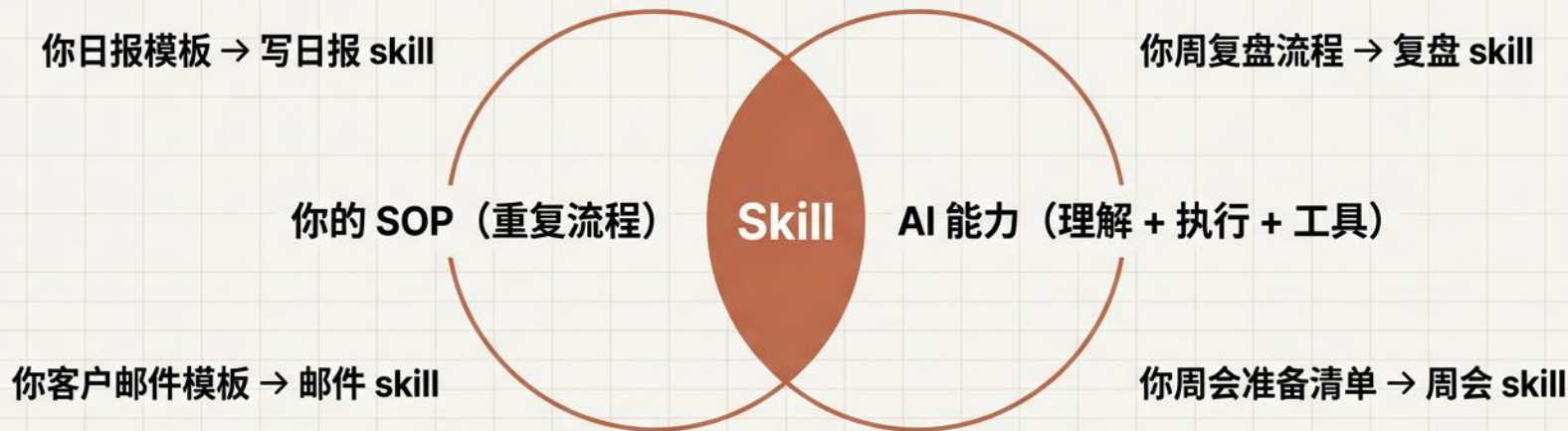
2. _____

3. _____

等会我会告诉你 — 这就是你下一个 skill 的原料

你刚写的 — 就是你下一个 skill 的雏形

Skill = 你工作里 反复做的步骤 + AI 的执行能力



你刚才写的那 3 件烦事 — 至少有 2 件能写成 skill

不用手搓 → 用 lovstudio-skill-creator

你描述需求，它生成完整 skill 骨架



自动生成 frontmatter

description 写得像广告语，触发
关键词自动提取



自动建立目录结构

SKILL.md / scripts/ /
references/ / assets/ 一键就位



自动注入最佳实践

单一职责、progressive
disclosure、错误兜底全部默认带上

> 帮我做一个 skill：每周一早上自动生成上周 git
commit 总结，发到飞书

→ Creator 生成完整 skill 骨架 (SKILL.md + scripts/)
→ 你只需要补 webhook URL

手搓是练肌肉，creator 是搬砖

⚠ HANDS-ON 3 · 15 MIN

用 creator 生成你的 skill

翻回 p29 — 你写的 3 件烦事里，挑 1 件

Step 1 · 装 creator

```
/plugin install lovstudio-skill-creator
```

Step 2 · 描述需求 (举手分享)

现场让 3 位学员分享他们的需求，老师挑 1 个现场演示：

```
> 帮我做一个 skill: [学员的需求]
```

这一步出问题别慌，下一节我们专门讲怎么调

Step 3 · 检查生成结果

- description 是否清晰?
- scripts/ 是否合理?
- 是否需要 reference?

MODULE 4 · 复查

creator 生成后，必须检查 3 处

01 · description

✓ 看：是不是用第三人称 + 触发关键词

→ 改：把"我的 X"改成"当用户说 X 时调用"

02 · scripts

✓ 看：参数 / 返回值 / 错误处理是否合理

→ 改：硬编码路径换成参数 / 加 try-except

03 · reference

✓ 看：长文档是不是真有用，还是凑数

→ 改：删掉冗余 / 保留真要参考的

creator 给你 80 分骨架 — 最后 20 分靠你自己改

▲ QUESTION · 讨论 1 MIN

scripts/ 和 reference/ — 该放哪？

调 GitHub API 拉
PR 列表

→ 放哪？

公司写作风格指南
5000 字

→ 放哪？

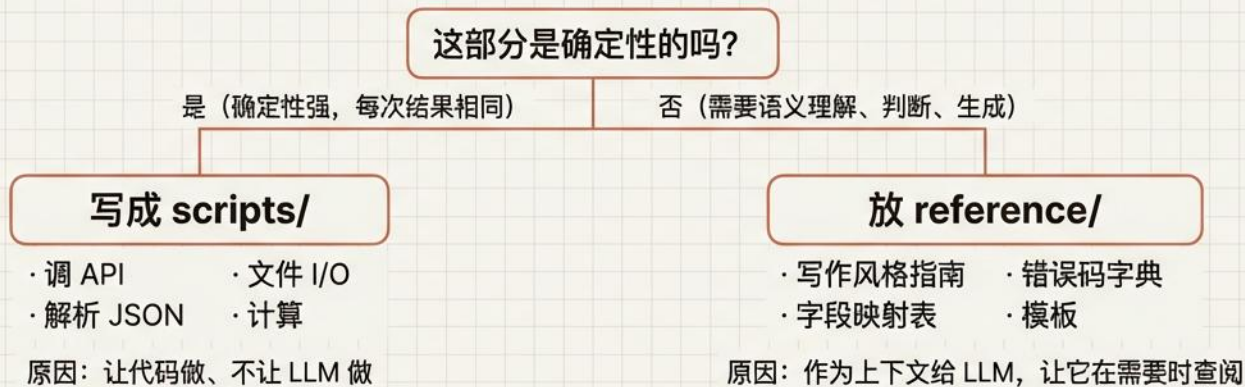
把 markdown 转
HTML 的代码

→ 放哪？

想清楚再翻页

ANSWER ←

scripts/ vs reference/ 决策树



回扣 p34 三个例子的答案：

- 调 GitHub API → **scripts/**
- 写作风格指南 → **reference/**
- markdown 转 HTML → **scripts/**

scripts 是肌肉，**reference** 是知识。LLM 是大脑 — 别让它做手脚的活

QUESTION · 讨论 1 MIN ▲

你的 skill 跑通了一 但是「好用」吗？

写出来 ≠ 好用 — 差在哪？



别人能不能
也用上？



失败了能不能
看懂为啥？



同样输入会不会
出不同结果？

ANSWER ←

5DQM — Skill 五维质量模型



D1 · Discoverable

能被找到吗？

description 是否覆盖
触发关键词



D2 · Decisive

能做决策吗？

是否给出"做"或"不
做"的明确边界



D3 · Deterministic

可回放吗？

同样输入 → 同样输
出，不靠运气



D4 · Documented

看得懂吗？

注释、reference、
example 是否完整



D5 · Defensive

失败兜底了吗？

错误信息是否给出
修复方向

每维 1-5 分。低于 3 分的维度，立刻迭代

⚠ QUESTION · 自检 · 2 MIN

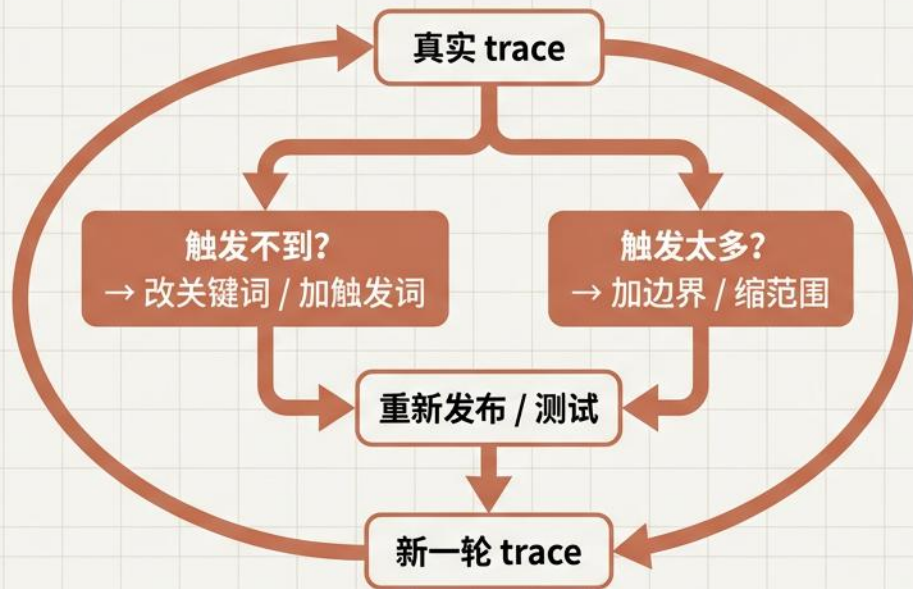
你的 description 过关了吗？

1. ☐ Q1 · 它是用第三人称写的吗？（“当用户...”而不是“我...”）
2. ☐ Q2 · 它列出了至少 3 个触发关键词吗？
3. ☐ Q3 · 它说清楚了 不适用 的边界吗？
4. ☐ Q4 · 完全不熟你工作的 AI 看了能调用吗？
5. ☐ Q5 · 它和现有其他 skill 的 description 有冲突吗？

5 个全勾 → 过关。少 1 个 → 翻下一页

ANSWER ←

description 不是写一次就完



v0.1: "深度研究并出报告"

→ 触发不到 (关键词太抽象)

```
v0.2: + "当用户问'调研 X'"分析 Y'时"
```

→ 触发太多 (任何提问都触发)

v0.3: + "仅当涉及多源对比 / 长报告"

→ ✓ 命中率合适

description 是不断喂数据微调的

MODULE 5 · 速查

Skill 设计 6 条黄金法则

01 单一职责

一个 skill 只解决一个问题

02 可回放

同样输入产出同样结果，幂等

03 失败兜底

错误要给方向，不要静默失败

04 触发明确

description 是合同，越具体越好

05 分层加载

metadata → body → reference → scripts

06 持续演进

用真实 trace 反哺迭代

法则不是教条 —— 但违反任何一条前，要给出充分理由

▲ QUESTION · 猜一猜 · 1 MIN

你猜 —— 新手最容易踩的是哪个坑？

A · 描述不清, AI 不调用

B · 描述太宽, AI 总调用

C · 写死路径, 换台机就报错

D · 把所有内容塞进 instructions

E · 让 LLM 算确定性的事

投个票 —— 翻页看答案

ANSWER ←

5 个最常见的坑 (按频率排)

✗ A · 触发不到 (最常见)

症状：明明装了，AI 不调用

原因：description 太抽象 / 关键词缺失

修：参考 p24 的 ✓ 正例

✗ B · 触发太频繁

症状：每次对话都触发，烦

原因：description 关键词太宽泛

修：加 "仅当...时调用" 边界

✗ D · 撑爆 token

症状：调用慢、context 满

原因：把所有内容塞 instructions

修：长内容挪到 reference/

✗ C · 路径硬编码

症状：换台电脑就报错

原因：写死了 /Users/xxx/...

修：用 ~ 或参数化

✗ D · 路径硬编码

症状：换台电脑就报错

原因：写死了 /Users/xxx/...

修：用 ~ 或参数化

✗ E · scripts 让 LLM 算

症状：精度问题、不稳定

原因：能用代码的事让 LLM 做

修：写成 Python script

新手最容易踩的是 A — 别紧张，所有人都踩过

QUESTION · 选一个 · 1 MIN

你刚写好的 skill — 你想给谁用？

A · 只给我自己用，私密 SOP

B · 给我团队 / 公司用，半开放

C · 全网公开，挣点钱

心里有数了再翻页

ANSWER ←

3 条路 · 不同 destinations

01 · 私人



路径: ~/.claude/skills/

受众: 只有你自己

适合: 私密 SOP、未成熟实验

工作量: 低

02 · 团队



路径: 私有 git repo

受众: 团队成员

适合: 公司内部规范、敏感 workflow

工作量: 中 (要写 README + 培训)

03 · 公开市场



路径: anthropics/skills · skillshare · lch-skill-store

受众: 全网

适合: 通用工具、品牌资产、变现

工作量: 高 (要 polish + 维护)

想清楚 — 选错路径会做白工

发布到 Ich-skill-store 5 步

01 fork 模板仓库

```
gh repo fork lovstudio/skill-template  
cd skill-template
```

02 把 skill 放进去

→ 文件夹 skills/your-skill-name/

✓ 带上 README.md + LICENSE + tests/

03 填写 metadata

package.json (name / version / description / author)

03 填写 metadata

package.json (name / version / description / author)

04 提 PR

```
gh pr create --title "Add: your-skill-name"
```

05 等 review

通常 1-3 天 / 通过后自动上架

第一次发布会被 review 提一堆建议 — 那是好事

⚠ QUESTION · 头脑风暴 · 2 MIN

你的 skill 能怎么帮你赚钱？

开个脑洞 — 越离谱越好



它能直接卖钱吗？



它能引流到我的别的产品吗？



它能让我接到工作吗？

写下 1 个想法 — 翻页对答案

5 种变现路径

① 付费 skill

模式：按月订阅 / 一次性买断

案例：高级排版、专业领域工具

价格：\$5-\$50 / 月

② 流量入口

模式：免费 skill 引流到付费课程 / 服务

案例：你装我的 skill → 关注公众号 → 报课

价格：转化率 5-15%

③ 接咨询

免费 skill 当作品集，吸引企业咨询

单价：\$500-\$5000

④ 团队培训

“我教你团队写 skill”

单价：\$1000-\$10000 / 次

⑤ 生态联运

和品牌方合作出联名 skill

比如：和某 SaaS 出官方 skill

“skill 是 2026 年最被低估的个人 IP 资产”

MODULE 6 · 协作

让 skill 长期演进

版本管理（陶土色细线）

semver 0.x 起步

- 0.1.0 → 第一个能用版本
- 0.2.0 → 加新功能 / 不破坏
- 1.0.0 → 公开正式发布

CHANGELOG.md 必备

每次 release 写：新增 / 修复 / 破坏性变更

PR review 6 条 checklist（陶土色编号）

1. ☐ description 是否有触发关键词？
2. ☐ 是否有单一职责？
3. ☐ scripts 是否有错误处理？
4. ☐ reference 是否真的需要？
5. ☐ 是否有 example？
6. ☐ CHANGELOG 是否更新？

review skill 跟 review code 一个标准

⚠ FINAL SHARE · 20 MIN

现在轮到你了

01 · 展示 (每人 2 min)

- ◎ 你的 skill 是什么?
- 解决什么重复任务?
- ↘ 现场跑一次

02 · 互评 (5DQM 维度)

左右两位同学打分: D1 / ☆☆☆☆☆ D2 / D3 ☆☆☆☆☆
D3 / ☆☆☆☆☆ D4 / D5 ☆☆☆☆☆

03 · 案例池 (自愿提交)

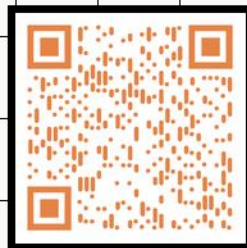
扫码加入“AI 创造营 · Skill 案例池”
所有学员的 skill 永久共享 + 月度精选发到公众号



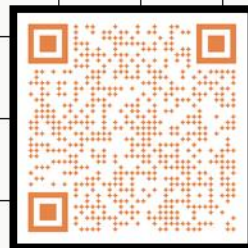
扫码加入案例池

▲ SCAN NOW · 现场扫码

让 AI 编程 从 SKILL 开始



公众号 · 手工川



个人微信 · 南川

谢谢

手工川 · LovStudio 创始人

END OF TALK · Q&A