

● ● ●  
第一章

# 养虾基本功： 从认识龙虾到成为合格的 CEO



- 一、认识OpenClaw：五分钟搞懂它到底是什么
- 二、安全部署：OpenClaw“住”在哪，怎么“住”才安全？
- 三、安装选择：OpenClaw、钉钉悟空、微信ClawBot、飞书妙搭……到底装哪个？
- 四、编写Skill：给OpenClaw写第一份岗位说明书
- 五、画组织图：画出你的组织架构，规划你的龙虾兵团
- 六、一人公司CEO守则（一）

## AI 一人公司： OpenClaw 龙虾兵团养成记

你决定养虾了。

可能是因为看到朋友圈有人晒出 OpenClaw 帮他自动生成的行业日报，你心动了。可能是因为你在某个深夜加班到两点的时候想，要是有个不用发工资的助理帮我干这些事就好了。也可能你只是好奇，想看看这个叫 OpenClaw 的东西到底能不能像别人说的那样好使。

不管你带着什么心态翻开这本书，我都建议你别急着动手。

养虾这件事，技术门槛没你想的那么高，但认知门槛比你想的高。很多人装好了 OpenClaw，跑了两周就放弃了。不是 OpenClaw 不好使，是他从一开始就没搞明白几件事：OpenClaw 到底是什么，该怎么装，怎么给 OpenClaw 写工作指令，自己的公司到底需要几只 OpenClaw，养虾得花多少钱，哪些事能交给 OpenClaw，哪些事绝对不能。

所以这一章不急着养虾。这一章是做准备，帮助你成为一个搞清楚了工具是什么、想明白了怎么用、算清楚了花多少钱、定好了分工边界的 CEO。

带着这个身份，再去养虾。

### 一、认识 OpenClaw：五分钟搞懂它到底是什么

**你大概率已经用过 AI 了。但你用的那个，跟龙虾不是一回事。**

你打开 ChatGPT、DeepSeek 或者 Kimi 的网页，输入一个问题，AI 给你一个回答。你觉得不错，复制粘贴拿去用了。下次你又有问题，再打开网页，再问一次。

这种用法就像你在路边拦了一辆出租车。你告诉司机去哪，他送你到了地点，交易结束。下次你再需要，重新拦一辆。司机不认识你，不知道你上次去了哪，也不知道你的偏好和习惯。每次都是从零开始。

**而 OpenClaw 不是出租车，是你雇的一个全职司机。**

当你第一次听到“OpenClaw”这个名字时，脑海里浮现的可能是某款冷门游戏，或者一个科幻电影里的机械爪子。但在 AI 圈子里，它正在成为一个让无数独立创业者为之兴奋的名字。



OpenClaw，直译为“开放之爪”，是一款开源的 AI 智能体框架（Agentic Framework）。

具体来说，OpenClaw 允许你将各类 AI 模型（如 ChatGPT、Claude、千问等）与外部工具、数据库、API 接口深度整合，让 AI 不再只是回答问题，而是能够自主规划任务、调用工具、执行操作、返回结果。

更重要的是，它能通过标准化的接口，将 AI 的能力无缝接入到我们日常工作的通信软件（如飞书、微信、钉钉）和办公工具中。

而“龙虾”这个外号，则来自开发者社区的一个小传统，OpenClaw 的 Logo 是一只红红的挥着钳子的虾，体现“精准夹取、灵活执行”的核心设计哲学，如图 1-1 所示。久而久之，“养一只龙虾”就成了部署 OpenClaw 的专属黑话，在独立开发者圈子里广泛流传。

2024 年初，OpenClaw 正式开源发布。它在前人工作的基础上，做了大量工程化精简，重点解决了三个痛点：部署门槛高、工具接入烦琐、非技术用户难以上手。它的出现，第一次使“让 AI 真正干活”这件事，变得对普通人触手可及。



图1-1 OpenClaw的Logo

（资料来源：OpenClaw官网，网址：<https://openclaw.ai>）

## 1. 原理：四个部分各管什么？

你不需要成为技术专家，但你得大致知道 OpenClaw 是怎么跑起来的，后面遇到问题才不会一头雾水。龙虾的工作原理可以拆成四个部分来理解，你把它想成一个人类员工就行。

**大模型是 OpenClaw 的大脑。**DeepSeek、Claude、ChatGPT、Kimi，这些你听过的 AI 模型，就是 OpenClaw 思考问题的大脑。OpenClaw 需要理解你的指令、分析信息、写文章、做判断的时候，都会调用大模型。你可以给 OpenClaw 配上不同的大脑：日常简单任务用便宜的模型，关键复杂任务用

## AI 一人公司： OpenClaw 龙虾兵团养成记

更强的模型。就像你有时候需要绞尽脑汁地想，有时候只需要简单判断。

**IM( Instant Messaging, 即时通信 ) 平台是 OpenClaw 的手脚。**飞书、钉钉、微信，这些通信平台是 OpenClaw 跟外界交互的通道。OpenClaw 通过 IM 平台接收你的指令、给你发送工作汇报、帮你回复客户消息、帮你操作各种在线工具。你可以把 IM 平台理解成 OpenClaw 的工位，它坐在飞书(或者钉钉、微信)里，在那个工位上接活、干活、汇报。

**Skill 是 OpenClaw 的技能。**每一份 Skill 配置文件就是一份工作指南，告诉 OpenClaw 在某个任务上该怎么做。你写了一份“行业日报”的 Skill，OpenClaw 就知道每天早上该去哪些网站抓取信息、按什么格式整理、什么信息该标为重点。你写了一份“客户服务”的 Skill，OpenClaw 就知道收到客户消息该怎么分类处理、什么能自己回、什么要转交给你。Skill 是你训练 OpenClaw 的核心手段，后面几章你会花大量时间学习写各种 Skill，它们决定了你的 OpenClaw 到底能不能干、好不好用。

**记忆体是 OpenClaw 的笔记。**OpenClaw 在工作过程中积累的所有信息都存在记忆体里，包括客户资料、项目记录、历史对话、你给过的反馈。记忆体存储在你的本地文件系统中，你可以随时查看和编辑。OpenClaw 每次开始工作前，会先翻一遍自己的笔记，确保它的行动是基于最新的信息。

四个部分的协作方式是这样的：你在 IM 平台上给 OpenClaw 下达一个指令（或者定时任务自动触发），OpenClaw 借助大脑（大模型）理解了你的意图，翻出自己的岗位说明书（Skill）确认该怎么做，查阅工作笔记（记忆体）获取相关背景信息，然后通过手脚（IM 平台和浏览器等工具）去执行任务，执行完把结果汇报给你，同时把新的信息更新到工作笔记里。

一个循环下来，OpenClaw 完成了一个任务，也变得比上一次更了解你的业务。

### 2. 对比：龙虾与真员工，各有什么好与不好？

到这里你可能会想：听起来龙虾就像一个数字员工，那它跟真员工到底有什么区别？



龙虾胜过人的几个方面如下。

(1) **永不疲劳**。OpenClaw 可以 7×24 小时在线。凌晨 3 点客户发来消息，OpenClaw 秒回。周末你在休息，OpenClaw 照样在巡逻线索、更新日报、处理邮件。真人员工有工时限制，OpenClaw 没有。

(2) **绝对听话**。你在 Skill 里怎么写的，OpenClaw 就怎么做，不打折扣、不偷懒、不闹情绪。它不会因为今天心情不好就敷衍了事，也不会觉得某个任务太无聊就不想干。

(3) **成本极低**。一只 OpenClaw 每月的成本可能就几百元钱，干的活相当于一个兼职助理。而且你不用交社保、不用提供工位、不用操心它会不会离职。

(4) **可以批量复制**。你写好一份 Skill，可以同时给好几只 OpenClaw 用。你的业务扩展了，龙虾兵团可以迅速扩编。真人员工你培养了半年走了，一切归零。OpenClaw 不会跳槽。

人胜过龙虾的几个方面如下。

**OpenClaw 没有真正的理解力**。OpenClaw 能按照你的标准执行，但它不会真正理解你的意图。你跟一个跟了你三年的真人助理说“这个客户有点特殊，需要注意一下”，他凭经验和直觉就知道该怎么做。你跟 OpenClaw 说同样的话，OpenClaw 可能会问你特殊是指什么。

**OpenClaw 没有创造力**。OpenClaw 能按照你的框架产出内容，但它不会真正灵光一现。那些最打动客户的洞察、最巧妙的策略角度、最让人眼前一亮的创意，大概率还是要从你的脑子里来。

**OpenClaw 没有人际关系**。客户信任的是你这个人，不是你的 OpenClaw。OpenClaw 可以帮你维护日常联系，但真正的信任，那种“把项目交给我你放心”的感觉，只能在人与人的真实互动中产生。

**OpenClaw 会犯低级错误**。OpenClaw 有时候会“一本正经地胡说八道”，它可能会编造一个不存在的数据、用错一个专业术语、在不该发言的场合说不合适的话。真人员工有常识，而 OpenClaw 在某些方面缺乏常识。

所以你对龙虾的正确期待应该是：它是你见过的最勤快、最听话、性价比最高的执行者，但它不是一个能替你思考和决策的合伙人。

## AI 一人公司： OpenClaw 龙虾兵团养成记

**你负责思考和判断，OpenClaw 负责执行。**

你们合在一起，才是一个完整的一人公司团队。

搞清楚了这个定位，你才不会犯两种最常见的错误：要么期待太高，觉得 OpenClaw 应该什么都能做，结果失望放弃；要么期待太低，只把 OpenClaw 当成一个高级聊天机器人，浪费了它真正的价值。



CEO 养虾指南

理解龙虾最好的方式不是去研究它的技术原理，而是把它当成一个真人新员工来对待。你会怎么带一个新来的员工？你会告诉他公司是做什么的、他负责什么、标准是什么、什么能做、什么不能做。你对 OpenClaw 也做同样的事，只不过你是用 Skill 配置文件来告诉它，而不是用嘴说。后面的每一章，本质上就是在教你怎么给不同岗位的 OpenClaw 写好这份入职培训材料。

## 二、安全部署：OpenClaw “住” 在哪，怎么 “住” 才安全？

**龙虾要 24 小时待命才有意义。**

你不可能每次想让 OpenClaw 干活的时候先开机、启动程序、等它就绪，那跟你自己干没多大区别。OpenClaw 要像一个真正的员工一样，而且是随时在岗，你发一条指令它马上执行，你睡着了它还在帮你跑定时任务。

所以，OpenClaw 需要一个住的地方，一台一直在线的机器。

你在哪台机器上跑 OpenClaw、怎么跑，直接决定了两件事：OpenClaw 的稳定性和你的数据安全性。这一节帮你把这两件事想清楚。

### 1. 两种部署方式：本地或云端

目前 OpenClaw 主流的部署方式有两种：跑在你自己的机器上（本地部署），或者跑在云服务器上（云端部署）。

#### （1）本地部署：OpenClaw 住在你自己的机器里

你用一台自己的电脑——可以是闲置的旧电脑、可以是专门买的 Mac



Mini 或 Windows 小主机——装上 OpenClaw，24 小时开着让它在上面跑。

本地部署最大的好处是数据完全在你手里。OpenClaw 的记忆体、你的客户资料、所有的工作文件都存在你面前这台物理机器上，不经过任何第三方服务器。对数据安全要求高的人来说，这是最踏实的方案。

这种部署方式的坏处也很明显。这台机器关了，OpenClaw 就下线了；断网了，OpenClaw 也就停止运行了。另外，本地部署的安装过程对非技术背景的人来说会稍微复杂一些，需要装 Node.js，配置环境变量，跑命令行。

这种方式适合什么人？在意数据隐私、有闲置设备或愿意买一台小主机、工作地点固定的人。

## （2）云端部署：OpenClaw 住在云服务器上

说实话，本地部署挺难的。如果你是第一次接触命令行和开发环境，那条路并不轻松。每一个小报错都可能让人卡上半天。

这不是你的问题，是本地部署本身的代价：高度自主的背后，是不可避免的技术门槛。

好消息是：**你不必非走那条路。**

各家大厂早就意识到，真正让 OpenClaw 普及的方式，不是让所有人都会搭环境，而是让这件事点几下就能完成。于是，云端部署方案应运而生。

云端部署，简单说就是 OpenClaw 不是运行在你自己的电脑上，而是跑在平台提供商的服务器上。你只需要通过网页完成授权和配置，剩下的事情平台全包了。服务器、运行环境、网络稳定性一概不用操心。

这种方式最大的好处是稳定和便利。云服务器 7×24 小时在线，不受你本地设备的影响。你出差在外地、改用手机、换了电脑，OpenClaw 照样在云端跑着，你随时随地都能访问和管理。

另外云端部署不占用你任何物理设备的资源。你的主力电脑正常运行，OpenClaw 在云上自己跑自己的。

坏处是云端部署的 OpenClaw 可能会有部分权限限制，另外还需要向平台支付一笔持续性费用。

这种方式适合什么人？不想折腾硬件、经常出差需要远程管理 OpenClaw、

## AI 一人公司： OpenClaw 龙虾兵团养成记

想要最快速度让 OpenClaw 跑起来的人。

**一条强烈的建议：不管选哪种，给 OpenClaw 一个独立环境。**

不管你是选择本地部署还是云端部署，有一条建议我需要特别强调：不要在你的主力工作电脑上裸跑 OpenClaw。

原因很简单。OpenClaw 有执行能力，它能读写文件、运行命令、操控浏览器。这些能力让它好用，也让它有风险。如果 OpenClaw 跑在你的主力电脑上，它理论上可以访问你电脑上的所有文件，包括工作文档、客户资料、私人照片、浏览器保存的密码。

万一 OpenClaw 误删了你的重要文件，或者更极端的情况下，某个第三方 Skill 有安全漏洞，诸如此类，你的整个工作环境都暴露在风险中。

最稳妥的做法是让 OpenClaw 拥有自己的独立工位：本地部署就用一台独立的机器，云端部署本身就天然隔离了。OpenClaw 在它自己的环境里折腾，但不管怎么折腾都影响不到你主力电脑上的东西。

如果你实在没有多余的设备，暂时需要在主力电脑上跑，至少做到这一点：把 OpenClaw 能访问的目录限制住。给 OpenClaw 划分一个专用的工作文件夹，不要让它有权限访问你的整个硬盘。这不是百分之百安全，但比完全不设限好得多。

### 2. 安全问题：你真正需要担心的是什么？

很多人在安全这个问题上的担心是模糊的，总觉得 OpenClaw 不太安全，但说不清到底不安全在哪里。我帮你把它拆成几个具体的问题，逐一说清楚。

#### (1) “OpenClaw 调用大模型的时候，我的数据会不会被泄露？”

OpenClaw 每次调用大模型都要把你的指令和相关内容发送到模型的服务器上，这是技术原理决定的，没法绕开。但大部分主流大模型厂商都承诺 API 调用的数据不会用于模型训练。对大部分一人公司的日常工作来说，这个风险等级跟你用微信发工作消息差不多，理论上有风险，实际上可接受。



但如果你处理的是客户的极度敏感信息，建议这类内容不要让 OpenClaw 经手。或者你也可以使用支持本地推理的开源模型，这样数据完全不离开你的机器，但算力要求更高、成本也更高。

### （2）“第三方 Skill 安全不安全？会不会偷我的信息？”

这是一个真实存在的风险。OpenClaw 的 Skill 生态是开放的，任何人都可以发布 Skill。大部分 Skill 是社区开发者善意贡献的，但不排除有恶意 Skill 混在里面，它可能读取你的 Cookie、API Key，甚至窃取敏感文件。

安装 Skill 之前，做三件事。第一，看看这个 Skill 的来源是不是可信的（官方推荐的、知名开发者的、社区高赞的通常没问题）。第二，看看它的代码是不是开源透明的。OpenClaw 的 Skill 大部分是可以直接查看源码的，你不用看懂每一行，但如果一个 Skill 的代码不但混淆又不让你看，那就别装它。第三，先在测试环境里跑一下，确认没问题再用到正式的 OpenClaw 工作上。

### （3）“OpenClaw 会不会自己乱操作？”

会的！不是 OpenClaw “故意”乱来，而是它的判断可能会出错。你让它整理文件，它可能把不该删的东西删了。你让它自动回复客户，它可能在不该发言的场合说了不合适的话。

这就是为什么后面“明确边界”那一节那么重要，你得在 OpenClaw 正式上岗之前把权限规矩定好：哪些操作 OpenClaw 可以自主做、哪些必须经过你确认、哪些绝对不能碰。

### （4）“万一出了问题，能恢复吗？”

部分可以。OpenClaw 的记忆体和工作文件都在本地文件系统里，你可以定期备份。文件改错了从备份里恢复就行。但如果 OpenClaw 已经把一条消息发出去了，发出去就是发出去了，收不回来，即使能收回来，客户也可能已经看到了。所以涉及对外发送的操作，一定要设置“先让我确认再发送”的机制，至少在磨合期内必须如此。

建议养成一个习惯：每周备份一次 OpenClaw 的工作目录。云端部署可以用云服务商的自动快照功能，本地部署可以用 Time Machine 或者简单的文件复制。备份这件事平时你觉得多余，出事那天你会感谢自己。

### 3. 核心原则：三条红线

说了这么多具体问题，可以总结成三条你必须守住的红线。

**第一，环境隔离。**OpenClaw 跑在独立环境里，不要跟你的核心工作数据混在一起。本地部署用独立机器，云端部署天然隔离。实在做不到使用独立机器，至少限制 OpenClaw 的文件访问范围。

**第二，敏感信息不入 OpenClaw。**你的银行密码、客户的核心商业机密、合同原件、身份证信息，这些东西不放在 OpenClaw 能访问到的地方。OpenClaw 需要用的信息你给它，不需要用的它碰不到。

**第三，对外操作先确认。**凡是涉及发送消息、发布内容、删除文件、支付转账的操作，OpenClaw 不能自主执行，必须先推给你确认。这条规矩在磨合期内是铁律，等你对 OpenClaw 的表现有了充分信心之后，可以逐步放宽，但放宽也是有选择地放宽，不是全部放开。

守住这三条红线，这只龙虾的安全风险就在可控范围内。不用因为担心安全就不敢用。这就像你不会因为员工有可能犯错就不敢招人一样。风险不是靠回避来消除的，是靠规矩来管理的。



部署方案的选择没有你想的那么重要。本地也好，云端也好，跑起来之后的体验差别不大，而且随时可以迁移。真正重要的是“环境隔离”这个意识。你给 OpenClaw 一个独立的工位，你的安全底线就有了保障。这一步做了，后面怎么折腾都不慌；这一步没做，后面出了事你连哭的地方都没有。

安全问题上最常见的两种极端：一种人什么都不设防，裸跑在主力电脑上随便装第三方 Skill，翻车了才后悔；另一种人什么都担心，研究了三天安全方案还没开始装。最好的安全策略是先做好基本防护，然后大胆用起来。三条红线守住就够了，剩下的在使用中根据实际情况再加固。完美的安全方案永远在路上，但 OpenClaw 不跑起来，你什么经验都积累不了。

### 三、安装选择：OpenClaw、钉钉悟空、微信ClawBot、飞书妙搭……到底装哪个？

你可能已经被各种“养虾教程”搞晕了。

打开小红书搜“龙虾”，有人教你装 OpenClaw，有人推荐钉钉悟空，有人说微信 ClawBot 最方便，有人说飞书妙搭一键部署两分钟搞定，有人说腾讯的 WorkBuddy 最方便，有人说智谱的 AutoClaw 很实用……每个人都说自己用的那个最好，你到底该听谁的？

这一节帮你厘清楚。

先说结论：**这些 OpenClaw 类工具不完全是竞品关系，更像是不同层级的解决方案。**它们解决的是同一个需求，但自由度、门槛、成本差别很大。

搞清楚它们各自是什么、适合谁，你就不会选错。

#### 1. OpenClaw 原版：最自由，也最折腾

OpenClaw 原版是一切的起点。前面两节说的龙虾架构，大模型做大脑、IM 做手脚、Skill 做岗位说明书、记忆体做工作笔记，说的就是 OpenClaw 这套开源框架。钉钉悟空、微信 ClawBot、飞书妙搭本质上都是在 OpenClaw 基础上做的二次封装。

你自己部署 OpenClaw 原版，最大的好处是完全可控。你可以自由选择接入哪个大模型（DeepSeek、Claude、ChatGPT、Kimi 都行）、连接哪个 IM 平台（飞书、钉钉、微信、Telegram 都支持）、安装什么 Skill、把数据存在哪里。所有配置你说了算，不受任何平台限制。

而且 OpenClaw 的开源社区极度活跃，在 GitHub 上超过 24 万星标，全球开发者贡献了上百个 Skill，每天都有新功能上线。你能用到最新、最全的技能。照着教程一步步来，大部分人一两个小时就能搞定。常见的 OpenClaw 本地部署步骤如图 1-2 所示。

# AI 一人公司： OpenClaw 龙虾兵团养成记

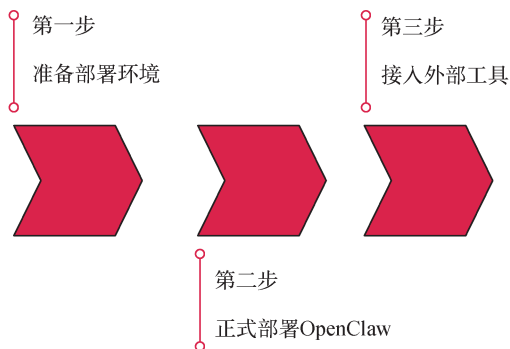


图1-2 常见的OpenClaw本地部署步骤

## 2. 钉钉悟空：企业级方案，为组织而生

2026年3月17日，阿里巴巴集团旗下钉钉发布了悟空，定位为“一个AI驱动的智能体工作平台，让AI像真人助手一样操作你的电脑，帮你完成从简单到复杂的各种工作任务”。

它不是 OpenClaw 的简单套壳，而是钉钉把底层代码全面重写后锤炼出来的独立产品。钉钉悟空的官网如图 1-3 所示。



图1-3 钉钉悟空的官网

(资料来源：钉钉悟空官网，网址：<https://wukong.dingtalk.com>)

悟空的核心卖点是企业级安全。具体来说就是：悟空自动继承你在钉钉里的企业权限规则，由此，AI的操作权限永远不会超过你本人。所有操作在

安全沙箱中运行，全程留痕可审计。token 消耗和 AI 使用成本一目了然。你不需要自己操心安全配置，平台为你兜底。

悟空还同步发布了 OPT(One Person Team, 一人团队)十大行业解决方案，覆盖电商、跨境电商、知识博主、开发、门店、设计、制造、法律、财税、猎头十大场景，你选一个跟你业务最接近的，一键启用就有一套现成的 OpenClaw 兵团配置，如图 1-4 所示。



图1-4 钉钉悟空的一人团队解决方案

(资料来源：钉钉悟空官网，网址：<https://wukong.dingtalk.com>)

悟空的缺点是目前还处在邀测阶段，功能在快速迭代中，稳定性还有提升空间。而且它深度绑定钉钉生态，如果你的日常工作不在钉钉上，用悟空就有点脱离语境。另外悟空的 Skill 生态虽然兼容开源 Skill 体系，但目前数量和成熟度与 OpenClaw 原版社区还有差距。

总的来说，钉钉悟空适合已经在用钉钉办公的企业和团队、对安全合规要求高的业务场景、希望有开箱即用的行业方案而不想从零配置的人。如果你的一人公司的客户和合作伙伴主要在钉钉生态里，悟空是很自然的选择。

### 3. 微信 ClawBot：最适合微信场景的入口

2026年3月22日，微信官方正式推出 ClawBot 插件，支持在微信里直接接入 OpenClaw。进入微信，按照“我—设置—插件”的路径，就可以查看终端安装指令。ClawBot 插件的介绍界面如图 1-5 所示。

## AI 一人公司： OpenClaw 龙虾兵团养成记



图1-5 ClawBot插件的介绍界面

(资料来源：腾讯公众号《刚刚，微信推出官方龙虾插件》，  
网址：[https://mp.weixin.qq.com/s/o\\_FPvJ0tY6aGqGn4Ea7Rpw](https://mp.weixin.qq.com/s/o_FPvJ0tY6aGqGn4Ea7Rpw))

这件事的意义极大：微信有超过十亿月活用户，是绝大多数中国人日常沟通的第一平台。你的客户在微信上，你的朋友在微信上，你的生活也在微信上。如果 OpenClaw 能直接住在微信里，那就是把 AI 助手放到了你每天打开几十次的地方。

ClawBot 作为微信联系人出现在你的聊天列表里，你像跟朋友聊天一样给它发指令，它在后台连接你的 OpenClaw 执行任务，然后把结果发回聊天窗口。它支持文字、图片、视频、文件等多种消息格式。

但微信对 OpenClaw 的限制也是四个方案里最多的。微信的开放接口远不如飞书和钉钉丰富，OpenClaw 在微信里能做的事主要限于消息层面——发消息、回消息、转发文件。你没法像在飞书里那样让 OpenClaw 直接操作文档、管理日程、创建任务。微信也没有原生的 Skill 市场和企业级权限管理。

另外需要注意的一点：目前 ClawBot 还在逐步放量，而且功能还在开放中。

在这之前，社区也有不少把 OpenClaw 接入微信的其他方案（如 openclaw-wechat 等），走的是代理服务模拟协议的路子。现在官方推出了正式接口，建议采用官方方案，省心也安全。

ClawBot 适合那些需要频繁通过微信与客户互动、希望 OpenClaw 住在



自己最常用的聊天工具里的人。ClawBot 很适合做客户沟通前线。

#### 4. 飞书妙搭：一键部署，门槛极低

飞书在这一轮“养虾大战”中跑得最快。2026年3月19日的飞书新品发布会上，飞书一口气推出了 aily 智能体平台、妙搭 Agent、多维表格 AI 搭建能力，以及 OpenClaw 官方插件和一键部署功能。飞书妙搭的官网如图 1-6 所示。



图1-6 飞书妙搭的官网

(资料来源：飞书妙搭官网，网址：<https://miaoda.feishu.cn>)

飞书妙搭的核心卖点就一个字：**快**。你在飞书里点几下按钮，不到两分钟就能拥有一只属于自己的龙虾——已经装好飞书官方插件，已经接入搜索和图片生成等热门 Skill，已经配好权限。你什么都不用折腾，直接开始用。

更关键的是，飞书的办公生态是众多平台里对 OpenClaw 最友好的。OpenClaw 在飞书里可以直接以你的身份操作云文档、读取群聊上下文、管理日程、创建多维表格。你不用在 OpenClaw 和工作应用之间来回搬运信息，OpenClaw 直接在你的工作环境里干活。目前火山引擎、Kimi、扣子、MiniMax、智谱等主流模型厂商与 AI 服务平台都已经完成了与飞书官方插件的对接。

使用飞书妙搭的坏处是你被绑定在飞书生态里，如果你不用飞书办公，单独为了养虾去装飞书可能会觉得多了一个不需要的工具。另外妙搭部署的

## AI 一人公司： OpenClaw 龙虾兵团养成记

自由度不如 OpenClaw 原版，高级配置和自定义 Skill 的灵活性有限。

飞书妙搭适合想要最快速度、最低门槛养出第一只虾的人。如果你完全没有技术基础、不想折腾部署、只想先让虾跑起来看看效果，飞书妙搭是目前最好的起步方案。

以上四种安装方案怎么选？方案对比见表 1-1。

表 1-1 OpenClaw 的四种安装方案对比

分析维度	OpenClaw 原版	钉钉悟空	微信 ClawBot	飞书妙搭
部署难度	较高，需要自己配置环境	中等，邀测阶段	低，扫码连接	极低，一键部署
自由度	最高，完全可控	中等，绑定钉钉生态	较低，功能受限	中等，绑定飞书生态
安全性	自己负责	企业级安全保障	依赖微信官方	飞书安全体系
办公集成深度	取决于你接入哪个平台	深度集成钉钉全能力	仅消息层面	深度集成飞书全能力
适合场景	长期重度使用	钉钉生态内的企业	微信端客户互动	快速起步、日常办公

另外，除了本节提到的这些，目前市面上涌现了众多其他 OpenClaw 工具，比如火山引擎的“ArkCraw”、智谱的“AutoClaw”、腾讯的“WorkBuddy”……在下面的章节里，我们会灵活使用这些 OpenClaw 工具来进行演示。

### CEO 养虾指南

平台工具之争看起来热闹，但对你来说核心问题只有一个：哪个能让我最快跑起来？你在选择上花的每一分钟，都是你没有花在养虾上的一分钟。OpenClaw 养得好不好，取决于你在实际使用中投入了多少时间去训练和迭代，而非取决于你选了哪个平台。

这个领域变化极快，你今天读到的信息，三个月后可能已经过时了。平台在迭代，功能在升级，新方案在冒出来。不要执着于选对，要善于迁移。你真正需要掌握的不是某个平台的操作方法，而是本书教你的一套养虾思维：怎么拆业务、怎么写 Skill、怎么定标准、怎么做迭代。这些东西不绑定任何平台，哪个平台都适用。



## 四、编写Skill：给OpenClaw写第一份岗位说明书

OpenClaw 装好了，你打开飞书，跟它说了第一句话：“帮我写一篇关于品牌定位的文章。”

OpenClaw 写好了。你看了一眼，还行，但不太对味。风格不像你平时的文章，角度也不是你想要的。你又说了一句：“不对，我要的是面向小企业主、通俗易懂的风格，不要太学术。”

你反复进行了五六轮，终于得到了一篇勉强能用的文章。

你会发现，这种每次都从头教的模式效率极低。你花在跟 OpenClaw 沟通上的时间，可能比你自己写还多。这就是为什么你需要 Skill。

Skill 的角色很简单：把你反复交代的那些要求，一次性写成一份文件，这份文件就是 Skill；OpenClaw 每次干活之前先读这份文件，就不用你再说一遍了。

简单说，Skill 就是一个文本文件，里面写着 OpenClaw 的工作流程。

这个文件有固定的格式，文件名为“SKILL.md”，放在特定的文件夹里。OpenClaw 每次接到相关任务的时候，会先打开这个文件，看看里面写了什么，然后按照你的要求去执行。

就这么简单。Skill 没有任何技术门槛，它就是一份文档，是你用中文写的、给 OpenClaw 看的工作说明。图 1-7 所示为一份 Skill 文件的内容。

```
1 ----
2 name: topic-positioning
3 description: 基于数据监测和竞品分析，为图书出版领域提供选题的精准
4 定位与差异化策略能力；当用户需要进行图书选题策划、分析市场缺
5 口、定位差异化卖点或验证选题可行性时使用
6 dependency:
7   python:
8     - requests >= 2.28.0
9   system: []
10 # 垂直领域选题的精准定位与差异化策略
11 ## 任务目标
12
13 本Skill用于图书出版领域的选题策划与定位决策，帮助编辑和策划人员：
14 - 基于数据监测识别高潜力选题方向
15 - 通过竞品分析发现市场缺口
16 - 构建选题的差异化定位与核心卖点
17 - 生成符合行业标准的选题策划书
18
19 ## 前置准备
20
21 ### 依赖说明
22 本Skill依赖以下Python包（已预装）：
23 - requests >= 2.28.0: 用于HTTP请求
24
25 ### 数据源准备（必需）
26
27 在使用本Skill前，需要准备以下数据源：
28 - 关键词监测数据源：S118、新榜或自定义数据源
29 - 竞品评论数据：电商平台同类图书的用户评论
30 - 行业动态信息：抖音、小红书、知乎等平台的讨论趋势
```

图 1-7 一份Skill文件的内容

## 1. 通用框架：一份 Skill 由什么组成？

不管你后面要写什么 Skill，每一份 Skill 都由同样的几个部分组成。你把这个框架记住，以后写任何 Skill 都往里填就行。

### (1) 第一部分：头部信息

头部信息是 Skill 的“身份证”，示例如下。

```
1 ---  
2 name: [ 这个 Skill 的名字，英文、短横线命名 ]  
3 version: 1.0.0  
4 description: [ 一句话说明清楚这个 Skill 是干什么的 ]  
5 author: [ 你的名字 ]  
6 ---
```

每个 Skill 都以这几行开头。名字用来识别不同 Skill。你以后可能有几十个 Skill，名字起得清楚找起来才方便。版本号用来追踪迭代，你改过几次一目了然。描述用来提醒你自己和 OpenClaw 这份 Skill 的核心任务是什么。

### (2) 第二部分：角色定义

角色定义的作用是告诉 OpenClaw 它是谁，示例如下。

```
1 ## 你的身份  
2 你是我的 [ 岗位名称 ]，负责 [ 核心职责 ]。
```

这一段很短但非常重要。OpenClaw 在执行任务的时候，会基于这个角色定位来做判断。你告诉它“你是我的品牌内容编辑”，它写出来的东西跟你告诉它“你是我的营销助理”是不一样的。角色越清晰，OpenClaw 的行为越精准。

### (3) 第三部分：任务描述

任务描述的作用是告诉 OpenClaw 你要做什么，示例如下。

```
1 ## 你的任务  
2 [ 具体描述这个 Skill 要完成的工作，包括什么时候做、做什么、做到什么程度 ]
```

这是 Skill 的核心。你要写清楚三件事：触发条件（什么时候执行，是每天定时跑，还是你提出要求才跑）、具体动作（做哪些事、按什么步骤）、完成标准（做到什么程度算完成）。



#### （4）第四部分：执行标准

执行标准的作用是告诉 OpenClaw 什么是好的、什么是不行的，示例如下。

```
1 ## 执行标准
2 [ 列出你对质量的要求、筛选条件、风格标准、格式规范等 ]
3
4 ## 不要做的事
5 [ 明确列出你的禁区和红线 ]
```

这一部分决定了 OpenClaw 的输出质量。你写得越具体，OpenClaw 的输出越稳定。特别注意要写“不要做的事”。OpenClaw 踩“坑”最多的地方往往是那些你没说但你以为它知道的常识。

#### （5）第五部分：输出格式

输出格式的作用是告诉 OpenClaw 结果应该是什么样，示例如下。

```
1 ## 输出格式
2 [ 定义 OpenClaw 完成任务后交给你的东西是什么格式，是一段文字、一份列表、一个文件还是一条消息 ]
```

格式不统一是 OpenClaw 最让人抓狂的毛病之一。你不定义格式，OpenClaw 今天给你一大段话，明天给你一个表格，后天给你三个选项，你每次都要重新适应。在 Skill 里固定格式，OpenClaw 每次的输出结构一致，你读起来高效，后续要做数据积累和对比也方便。

#### （6）第六部分：前置参考（可选）

前置参考的作用是告诉 OpenClaw 干活之前先读什么，示例如下。

```
1 ## 前置参考
2 执行本任务前，先阅读：
3 - [ 某份文件的路径 ]
4 - [ 另一份文件的路径 ]
```

有些 Skill 不是独立的，OpenClaw 需要参考其他文件才能干好工作。比如写内容之前要先读品牌语言手册、筛选线索之前要先读客户画像。

#### （7）第七部分：注意事项（可选）

注意事项可以补充你想到的其他规矩，示例如下。

- 1 ## 注意事项
- 2 - [ 任何你觉得需要额外交代的事 ]
- 3 - [ 比如：不确定的情况先问我，不要猜 ]
- 4 - [ 比如：同一条信息不要重复处理 ]

七个部分构成了一份 Skill 的完整框架，不是每一个部分都必须有。其中，头部信息、角色定义、任务描述、输出格式这四个是必需的，其他的根据任务复杂度按需添加。

把这个框架记住，后面你写任何 Skill，都是往这个框架里填内容。

## 2. 创建 Skill 的三种方式

现在问题来了：这个 Skill 文件从何而来？你有三种选择，并且三种选择难度递减，效果也递减。选哪种取决于你的技术水平和个性化需求。

### (1) 方式一：自己动手写文件

自己动手写 Skill 文件并不难，你需要在 OpenClaw 的工作目录里创建一个文件夹。OpenClaw 一般都会有一个“Skills”文件夹，在里面创建一个 Skill 文件夹如“email-writer”，然后再在这个文件夹里创建“SKILL.md”文件即可，如图 1-8 所示。

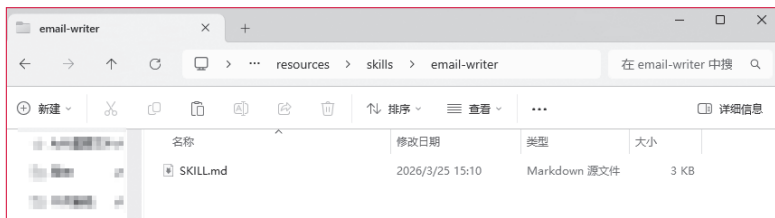


图 1-8 在 OpenClaw 工作目录里创建的 Skill 文件

然后打开这个文件，参照前面的模板，一行一行地编写 OpenClaw 的工作手册。

这种方式的优点是能够实现完全个性化。你想让 OpenClaw 怎么工作，就怎么写；你想加什么特殊要求，就加什么。OpenClaw 的行为完全按你的想法来，只要指令够准确，几乎不会产生理解偏差。

这种方式的缺点是需要你对自己的需求想得很清楚，而且要有耐心把这



些需求写成 OpenClaw 能理解的文字。第一次写可能会花半小时到一小时。

## (2) 方式二：用自然语言生成

OpenClaw 生态里的工具，比如“扣子编程”“WorkBuddy”等，能让你用自然语言描述你想要的 Skill，然后自动生成对应的文件。

生成操作很简单。举个例子，你只需要跟工具说：“我想要一个邮件写手的 Skill，要求它能写各种商务邮件，语言正式但不生硬，重点信息突出，每封邮件都要有明确的下一步动作。我是做财务咨询的，主要服务中小企业主。”图 1-9 所示为扣子编程平台的 Skill 文件生成功能区。



图 1-9 扣子编程平台的 Skill 文件生成功能区

扣子编程这类工具会根据你的描述，生成一个完整的 SKILL.md 文件，你可以直接下载保存。保存下来的 Skill 文件可以随意导入其他 OpenClaw 工具使用。扣子编程创建好的 Skill 如图 1-10 所示。



图 1-10 扣子编程创建好的 Skill

这种方式的优点是快，几分钟就能搞定一个 Skill。这种方式的缺点是生成的内容可能不够贴合你的具体需求，需要后续调优。

## AI 一人公司： OpenClaw 龙虾兵团养成记

### (3) 方式三：去 Skill 商店下载

这是最省事的方式。OpenClaw 工具的生态里往往都有一个 Skill 商店，里面有各种现成的 Skill 文件，覆盖了大部分常见的工作场景。直接搜索你需要的类型，比如“邮件写手”“数据分析师”“PPT 制作”，找到评价好的下载下来，放到你 OpenClaw 的工作目录里就能用。图 1-11 所示为飞书妙搭的 OpenClaw 搜索 Skill 的界面。

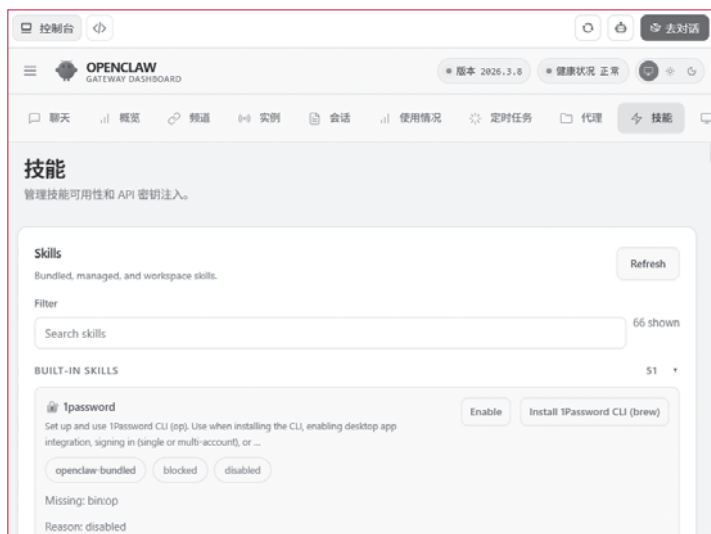


图 1-11 飞书妙搭 OpenClaw 搜索 Skill 的界面

这种方式的优点是省时省力，而且经过其他用户验证的 Skill 通常质量不错。这种方式的缺点是通用性的东西未必完全符合你的个性化需求。

在下面的章节里，我们会综合使用这些方法来创建 Skill，什么方便就用什么。



Skill 是你跟 OpenClaw 之间的契约。你把标准写清楚，OpenClaw 按标准执行；你没写的，OpenClaw 就靠猜，猜对了是运气，猜错了你来兜底。所以 Skill 写得好不好，直接决定了你的 OpenClaw 好不好用。后面每一章你都会写 Skill，这件事会越做越熟练，但底层的道理就是这一节讲的这些。

## 五、画组织图：画出你的组织架构，规划你的龙虾兵团

先问你一个问题：**你的公司有组织架构图吗？**

如果你在一家正规公司上过班，你应该见过那种挂在墙上的组织架构图。CEO 在最上面，下面是几个 VP（Vice President，副总裁），再下面是总监、经理、员工，清清楚楚，谁管谁一目了然。

现在你出来自己干了，一人公司，所有事情你一个人扛。你大概率没有画过这张图，因为画了也没用，既然是一人公司，每个格子里填的都是你自己的名字。

一个人干所有的事，最容易出现的问题就是乱。你分不清现在在干的到底是哪个岗位的活，分不清哪些事重要、哪些事紧急，更分不清哪些事你必须亲自干、哪些事其实可以交出去。

现在，有了 OpenClaw，这张图上终于可以填上不同的名字了。

每个人的一人公司都不一样，没有一套“标准架构”适用于所有人。下面提供三种组织架构图画法，你可以根据自己的业务类型选择一种最顺手的来用。

### 1. 画法一：成交链路法

这个方法针对有明确客户、有清晰业务成交链路的一人公司，适合做咨询、设计、培训、代运营等，或以工作室形式接项目的一人公司。

方法很简单：**想象一块钱从客户口袋里到了你口袋里，这一块钱走过的每一个环节，就是你需要有人负责的事。**

举个例子，老王主要做企业内训，他的成交链路如图 1-12 所示。

七个环节，一目了然。

老王的“核心交付”环节是他亲自上台讲课，这个不能交给 OpenClaw；“线索获取”和“商务对接”他更喜欢自己来谈，因为他的客户关系经营能力就是他的核心竞争力。于是老王规划出了他自己的“虾兵团”：

## AI 一人公司： OpenClaw 龙虾兵团养成记

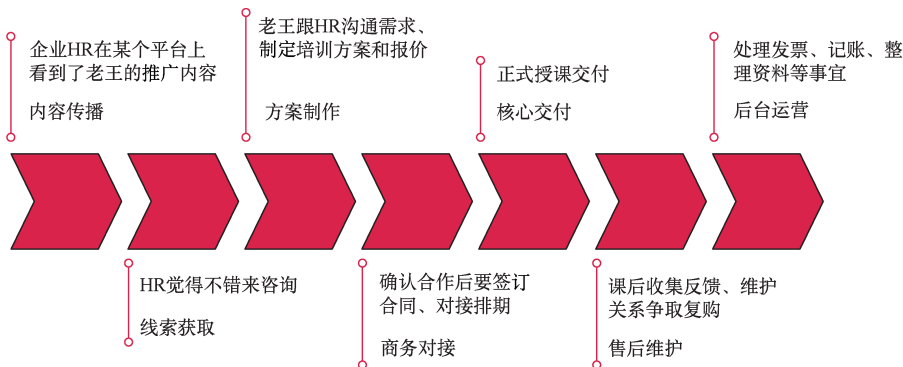


图1-12 老王的成交链路

- **内容虾（负责内容传播）**：帮他把选好的题目变成文章，排版好后发到各平台。
- **美工虾（负责方案制作）**：帮他把培训大纲变成排版精良的PPT提案。
- **跟单虾（负责售后维护）**：帮他自动跟进培训后的客户反馈和复购。
- **管家虾（负责后台运营）**：帮他处理发票、记账、归档资料。

### 2. 画法二：清单归类法

这个方法针对事项零碎、没有线性流程、什么都需要干一点的业务，适合自由插画师、独立开发者、接多种散活的自由职业者等。

这类创业者的一人公司特点是：没有一条清晰的客户从知道你到付钱的链路，你的收入可能来自好几个不同的渠道，每天的工作内容也不固定。用成交链路法画不出一条直线，因为你的钱是从好几个方向来的。

这时候换一个思路：**别管钱怎么走，先把你这一周真实干的所有事列出来。**

举个例子，小美是一名自由插画师，兼做表情包授权和绘画教程。

她列出了自己某一天干的所有事：画客户约稿的插画3张、跟2个客户沟通修改意见、在小红书发了4条作品、回复了小红书上的私信和评论、剪了1条绘画过程的短视频、上传了新的表情包到平台、回复了表情包合作的邮件、整理了上个月的收入明细、找了半天去年的一个源文件、更新了自己的作品集网站、看了几个同行的账号寻找灵感。



列完之后，把这些事归类，见表 1-2。

表1-2 小美的工作内容清单

类别	具体事项
创作/交付	画插画、改稿
内容运营	发小红书、剪短视频、私信评论
商务沟通	回邮件、谈合作、跟客户核对修改意见
资产管理	上传表情包、更新作品集网站
灵感调研	参考同行作品
后台杂务	整理收入、找文件

按照六个类别整理好，小美进行了分析：创作是她的核心价值，一笔一笔画的过程必须她亲自来，这个 OpenClaw 完成不了。但内容运营每周花 6 个多小时，其中大部分时间花在排版发布和回复日常私信上；后台杂务她一直拖着不想干；灵感调研她想做但总是没时间。

基于以上分析，小美的“虾兵团”规划如下：

- **发布虾（负责内容运营）**：帮她把画好的作品自动排版发布到各平台，自动回复常规私信。
- **灵感虾（负责灵感调研）**：每天自动抓取同行的热门作品、风格趋势，整理成灵感周报。
- **管家虾（负责后台杂务）**：自动整理收入、归档文件、更新作品集。
- **信使虾（负责商务沟通）**：自动回复合作咨询的初步信息，帮她起草报价邮件。

### 3. 画法三：痛点倒推法

如果你的一人公司已经在运营，不想做复杂规划，只想尽快解决眼前最大的痛点，这个方法会更适合你。**不用想太多，列出最近一个月让你最烦、感觉最消耗、最想甩掉的三件事。**

比如老张经营一个做短视频代运营的一人公司，他写的三个痛点如下：

- 每天要花 2 小时给三个客户写脚本、改脚本，产能不足。

## AI 一人公司： OpenClaw 龙虾兵团养成记

- 老客户做完了就失联，没有人跟进续约，收入全靠新客户。
- 自己的账号 3 个月没更新了，完全没时间做自己的内容。

然后把每个痛点转换成一个岗位需求：

- **产能不足**：需要一只帮他批量产出脚本初稿的编剧虾。
- **老客户失联**：需要一只帮他自动跟进、维护客户关系的客情虾。
- **账号断更**：需要一只帮他做自己品牌内容的内容虾。

到这里，你应该也发现了，**每个一人公司的虾兵团都不一样**。OpenClaw 的数量不同，种类不同，优先级也不同。这才是正常的。因为每个人的业务不同、痛点不同、核心竞争力不同，养的 OpenClaw 当然不一样。

所以，请你不要拿别人的虾兵团来套自己的公司。先画你自己公司的组织架构图，再决定养什么虾。

虽然每个人的架构不同，但我们研究了大量一人公司的实际案例之后发现，有六个角色出现的频率最高。绝大多数一人公司，不管什么行业，都多多少少需要有人承担这六个角色：

- **首席战略官**：管“看路”——收集行业资讯、监控竞品、整理市场信号。
- **首席品牌官**：管“发声”——内容选题、文案撰写、多平台更新。
- **首席增长官**：管“拉客”——找潜在客户、自动触达、追踪转化。
- **首席设计官**：管“出活”——做 PPT、出海报、做排版。
- **首席服务官**：管“接客”——回复客户、跟进项目、处理售后。
- **首席运营官**：管“后台”——管邮件、管日程、管文件、管财务。

本书的第二章到第七章，将会手把手教你怎么养这六只虾。从 Skill 配置到提示词写法，从日常任务设定到长期训练方法，全流程覆盖。但请注意：**这六只虾是最大公约数，不是标准答案。**

如果你画完图发现自己需要的是“编剧虾”“选品虾”“社群虾”……都完全没问题。我们教给你的每一个养虾方法、每一条配置原则、每一个训练思路，都可以应用到你自己的具体业务上。

养虾的方法是通用的。学会养这六只，你就学会了养任何一只。



现在，请你先画出自己的那张图。

CEO养虾指南

这一节你只做了一件事：画了一张图。但这张图的价值，可能比后面养的任何 OpenClaw 都大。因为它第一次逼着你把自己一个人干的所有事拆开来看——哪些是判断，哪些是执行，哪些是你的核心价值，哪些只是在消耗你的时间。画组织架构图这件事，本质上是把一个人干的所有事拆分成几个方面。

最后记住一个判断标准：**如果一件事作出核心决策只需要 10 分钟，但执行要花 2 小时，这就是一件应该交给 OpenClaw 的事。**

## 六、一人公司CEO守则（一）

1. **一人公司不是没有组织架构，而是老板从来没画过。**你一个人扮演六七个角色不是问题，问题是你分不清自己此刻正在扮演哪一个。画出来，才能管得住。

2. **你的时间花在哪里，你的公司就长成什么样。**如果你 80% 的时间都在做执行，你的公司就只有执行力。战略、增长、品牌这些方向不会因为你忙不过来就自动长出来。

3. **花得出去的钱都是小钱，算不清楚的账才是大钱。**每月花多少不重要，重要的是你花出去的每一块钱能不能算出回报。算不出回报的投入不是投资，是消费。

4. **能被写成规则的事就不值得你亲自做。**反过来也成立。如果你干的每件事都能被写成规则，说明你还没有在做真正属于 CEO 的工作。

5. **管理的本质不是让别人干活，而是定标准。**标准定得越清楚，你越轻松；标准定得越含糊，你越累。大多数人觉得“管人好累”，其实是标准没有想明白。管理 OpenClaw 的能力，本质上就是管理人的能力。如果你不会检查 OpenClaw 的工作，就不会检查员工的工作。养虾是管理力的“照妖镜”。

6. **敢交出去，是 CEO 的必修课。**什么都自己干的人不是敬业，是不信

## AI 一人公司： OpenClaw 龙虾兵团养成记

任。不信任的本质往往不是对方不行，是你没有把标准讲清楚。

**7. 你对业务的理解有多清晰，你的团队就能跑多顺。**很多时候你觉得“下面的人不好使”，其实是你自己没想清楚标准是什么。指令模糊，执行必然走样。你花在教 OpenClaw 上的每一分钟，本质上都是在编写你公司事务的操作手册。

**8. 最值钱的能力是定义问题而非解决问题的能力。**OpenClaw 可以帮你解决问题，但如果你问错了问题，OpenClaw 越高效，你错得越快。CEO 的核心功夫应该是在一堆看起来都很紧急的事情里，找出那件真正重要的。

**9. 一人公司最危险的错觉是“忙就是有用”。**你今天忙了 12 小时，但有多少时间花在了真正推动业务往前走的事上？答不上来，说明你还在给自己的公司打工，而不是在经营它。

**10. 你不是在学一个工具，你是在开一家公司。**把自己当成公司来经营，而不是当成工具来使用。工具卖的是时间和技能，公司卖的是系统和价值。你要建的不是一个“离了你就转不了”的作坊，而应该是一个“你在不在都能跑”的系统。从现在开始，换一个身份来读这本书。