



第 1 章

扣子编程平台介绍



本章导读

智能体开发已经从专业程序员专属的开发工作，变成普通爱好者、职场人员都能上手落地的项目。本章围绕扣子 AI 编程平台展开全方面入门讲解，是全书的基础奠基章节。首先从扣子编程基础概念入手，依次讲解扣子编程的产品能力、平台优势、整体架构与版本套餐；随后进入智能体核心理论板块，详解智能体定义、底层本质；最后结合天气查询助手实战案例，让读者体验扣子平台如何从 0 到 1 开发智能体，并介绍部署智能体全流程，为后续开发各类智能体项目打好理论与实操基础。

学习目标

- (1) 理解扣子编程定位，掌握智能体核心定义，能够独立完成天气查询助手的开发、预览、测试全流程操作。
- (2) 掌握 API、WebSDK 两种部署渠道的区别、适用场景，学会生成个人访问令牌完成网页嵌入。
- (3) 建立零代码 AI 开发基础认知，能独立规划简单对话类智能体的开发需求。

本章重点

扣子编程平台简介、智能体概述、天气查询智能体开发与部署案例。



1.1 扣子编程简介

扣子编程是一个 AI 驱动的应用开发平台，打造了兼容多种开发范式的开发者生态，可以帮助你构建并交付生产级别的全栈 AI 应用。只需要清晰地描述你的需求，扣子编程就能帮助你打造可用于生产环境的智能体、工作流和网页应用。

1.1.1 扣子编程能做什么

在 AI 和大模型兴起的时代，整个行业除了在讨论 AI、大模型这两个关键词，另一个被广泛讨论的概念是 AI 编程。随着 AI 模型在代码生成与逻辑推理能力上的爆发性提升，AI 时代的开发范式正

发生深刻变革，正在从过去的“人主导、工具辅助”，逐步地走向“AI 主导、人来辅助”的模型驱动新时代。

为了顺应这一趋势，扣子编程为开发者提供了这一低门槛、高上限的“AI 编程”开发范式。在这里，可能只需一句话，即可一站式完成从构建到落地的完整生产级闭环。

扣子编程对以下核心场景提供 AI 编程和全代码开发支持：

- **智能体**：通过简单的对话即可构建全代码智能体，包括日常场景的即问即答 Agent、复杂得多步骤长时间任务（Long-Running）Agent。
- **工作流**：全代码开发工作流，将用户的需求转化为可执行、可观测、可迭代的工作流代码，可通过 API 集成到自己的业务流程中。
- **网页应用**：包含前后端逻辑的全栈网页应用，支持自定义设置视觉风格，可实现自定义域名部署。

开发者可以借助可视化设计与编排工具，以低代码或零代码的方式开发智能体、工作流、应用，并将其发布到各个社交平台、通信软件，也可以通过 API 或 SDK 将其集成到业务系统中。

这个时代需要有一个平台，能够以更低的门槛帮助用户进行 AI 编程实现创意，可以说扣子这类产品是 AI 时代的背景下自然催生的产品。

1.1.2 为什么选择扣子编程

扣子编程内置多种业界领先的优秀模型服务，能够快速将用户的创意转化为生产环境可用的全栈式应用，覆盖智能体、工作流、网页应用等多个开发场景。在扣子编程中，你只需要清晰表达意图，就能一站式完成从构建到落地的完整生产级闭环，让你的开发过程更聚焦于创意和价值创造。

具体来说，扣子编程平台赋予了六大能力：

（1）交互式 AI 编程能力：打开浏览器网页即可开始开发。你无须安装任何工具，扣子编程为每个项目自动分配云端运行环境和对应资源，自带编码所需的基础功能，提供了开箱即用的编程体验。扣子编程融合了 AI 编程能力，完美适配氛围编程（Vibe Coding）范式，只要通过简单的对话，就能让 AI 承担编码、调试等烦琐工作，将复杂的代码逻辑转变为清晰可见的图形界面，以供预览与验收。无须感知扣子编程生成的全代码，零编程基础的用户也能实时调试与预览应用。

（2）生产级基建能力：搭建生产级应用，往往需要处理复杂的集成配置、集成全套的数据库与存储能力、保障企业级应用开发的数据安全。扣子编程在这些方面提供了强大的支持。

（3）集成能力：扣子编程预置了丰富的集成能力，内置包括方舟在内的业界先进模型集成；同时也支持一键授权外部集成，例如飞书表格、飞书消息等，提供开箱即用的集成服务。

（4）数据与存储能力：扣子编程打造了开箱即用的全套数据库与存储设施。无须申请云资源、配置连接池，直接一键开启即可使用数据存储方案。同时提供可视化的数据管理后台，让非技术人员也能方便地管理业务数据；支持数据库的一键回溯操作，极端情况下也无须担心数据丢失。

（5）安全能力：扣子编程采用统一的环境隔离策略，严格隔离开发与生产环境，防止测试数据

污染线上环境，保障业务安全。敏感信息（如 Key、Token）采用加密环境变量管理，存储传输全程加密，运行时解密，杜绝硬编码风险。

（6）一站式部署能力：你无须关注底层的服务器资源分配，也无须手动打包与构建，只需要单击“部署”，扣子编程会自动完成云端打包、构建与服务部署。发布的版本可回退，部署记录完善且可追踪。线上日志可实时查看，方便开发者及时发现并解决应用运行过程中出现的问题，确保生产级应用的稳定运行。

1.1.3 扣子平台架构

扣子为 AI 应用（包括智能体）开发人员提供了一站式全链路的 AI 应用搭建能力，包括应用开发、测评、监控和丰富的发布渠道。

如图 1-1 所示，空间是扣子编程最顶层的资源组织方式，通过工作空间对开发资源进行隔离。

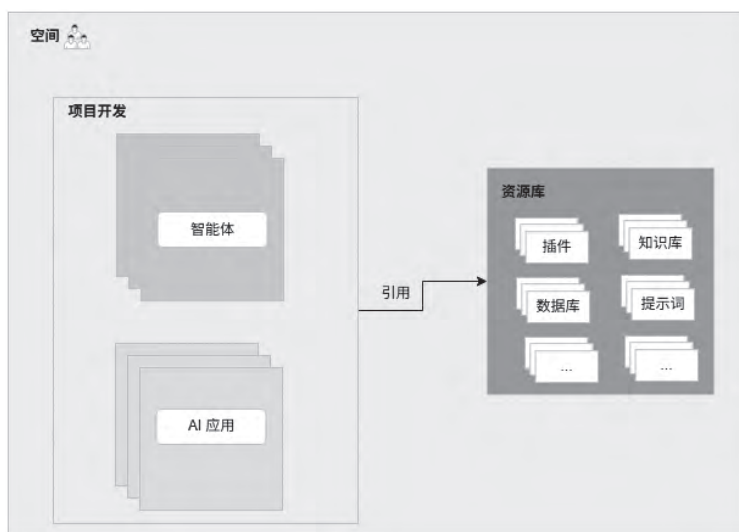


图 1-1 平台架构

- **空间**：是资源组织的基础单元，不同空间内的资源和数据相互隔离。一个空间内可创建多个智能体和 AI 应用，并包含一个资源库。在资源库中创建的资源可以被相同空间内的智能体和 AI 应用使用。
- **项目**：分为智能体和 AI 应用两种类型，AI 应用既可以创建多种应用专属资源，也可以和智能体共享空间资源库中的资源。
- **智能体（Agent）**：通常指的是一个能够独立执行任务、做出决策并进行学习的自动化程序。智能体可以根据用户输入的指令，自主调用模型、知识库、插件等技能并完成编排，最终完成用户的指令。
- **AI 应用**：是指利用大模型技术开发的应用程序，这些应用程序能够使用大模型，执行复杂任务，分析数据，并作出决策。
- **资源库**：可以在资源库内创建、发布、管理共享资源，例如插件、知识库、数据库、提示词等。这些资源可以被同一空间内的智能体和应用使用。

资源可以存在于两个实体内，一个是空间的资源库，一个是 AI 应用项目中的项目资源库。在空间资源库内创建的资源可以被空间内的 AI 应用项目和智能体项目使用，属于空间内的共享资源。在 AI 应用项目中也可以创建资源，但这些资源是项目自有的资源，默认不可以被其他项目使用也不会展示在空间资源库内。

1.1.4 扣子平台订阅套餐

为满足不同用户使用需求，扣子推出的订阅制服务，就像手机通信套餐一样，每个月为你提供固定的功能权益与积分额度。扣子订阅套餐包括个人版和企业版，如图 1-2 所示。个人版提供从免费版到尊享版共五个档位的套餐，可以满足不同的个人开发者从日常体验到深度开发的个性化需求。企业版是针对企业团队，分为标准版和旗舰版。标准版适合需要多人协作开发、共享资源、进行权限管理的中小型团队；企业旗舰版适合对资源量、并发量、数据安全、服务支持有更高要求，或需要大规模协作的大型团队或企业。

套餐类型	个人版	企业版
套餐档位	免费版、进阶版、高阶版、旗舰版、尊享版	标准版、旗舰版
适用主体	个人开发者、爱好者	中小型或大型企业团队
权益亮点	- 个人免费版：每日登录可获得活动积分，免费体验 - 个人付费版：提供套餐月度积分，享有更多功能	- 支持空间协作、成员管理 - 企业旗舰版支持 SSO 单点登录、VPC 私网连接等企业安全特性。

图 1-2 个人版与企业版

个人版不同套餐的积分和权益有所不同，它们的对比如图 1-3 所示。积分是扣子平台各项资源消耗的统一计费与结算单位，不可反向兑换为现金。积分主要用于抵扣扣子对话任务、AI 编程、智能体、应用、工作流等场景产生的资源消耗，例如大模型 Token、工具调用、音视频时长及存储等费用。你可以通过购买套餐、购买积分、参与平台活动等多种途径获取扣子积分。个人版用户每日登录还可获取免费积分。当积分为 0 时，服务将暂停。你需要补充积分才能继续使用。

套餐版本	个人免费版	个人进阶版	个人高阶版	个人旗舰版	个人尊享版
价格 (月)	0元	39.9元	99元	199元	999元
积分 (月)	+	3万	9.9万	19.9万	99.9万
扣子					
模型					
接入自定义模型	-	✓	✓	✓	✓
连续调用: Gendaoos2.0	-	-	✓	✓	✓
模型调用无限制	-	-	-	-	✓

图 1-3 个人版套餐

个人免费版适合新手用户，免费体验扣子。而个人进阶版适合普通创作者，其高性价比满足日常创作需求。个人高阶版适合高频使用者，提供更充裕的积分和资源额度。个人旗舰版适合专业开发者，提供个人版较全面的资源配额。个人尊享版则最适合专业开发者，提供个人版最高积分额度，享有新模型优先内测资格。

扣子个人版套餐生命周期如图 1-4 所示，注册扣子后，默认订阅个人免费版，你可以随时升级到个人进阶版、个人高阶版、个人旗舰版或个人尊享版，升级立即生效。你也可以按需进行降级、续订、购买积分等操作。个人版套餐不支持退款，针对连续包年、包月的个人版套餐，你可以随时取消自动续订。个人版不支持超额使用，积分为 0 后，系统将停止服务，自动拦截扣子所有可能产生资源消耗的请求，包括开发 AI 编程项目、智能体调用、API 请求等。

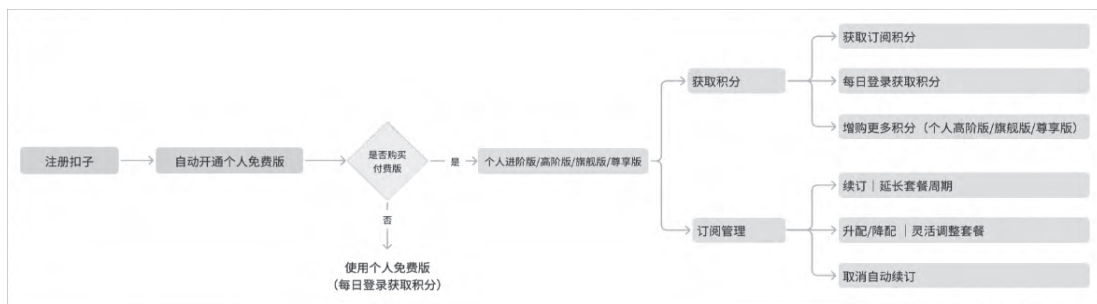


图 1-4 个人版套餐生命周期

企业版是扣子平台提供的多成员订阅计划，专为希望整个团队使用扣子平台的公司组织而设计，允许多个成员独立访问和使用扣子平台。每个用户都将拥有自己的扣子平台账户。如图 1-5 所示，当你注册火山引擎账号并完成实名认证后，可以购买企业版，包括标准版和旗舰版。你也可以按需进行退订、更配、购买积分等操作。



图 1-5 企业版生命周期

而企业标准版和企业旗舰版的价格和具体功能区别，如图 1-6 所示。企业标准版适合需要多人协作开发、共享资源、进行权限管理的中小型团队；企业旗舰版适合对资源量、并发量、数据安全、服务支持有更高要求，或需要大规模协作的大型团队或企业。



图 1-6 企业版套餐功能和价格区别



1.2 智能体概述

1.2.1 什么是智能体及智能体的本质

智能体是基于对话驱动的 AI 应用项目形态，区别于传统单纯问答的大模型对话，它具备完整的业务执行能力。智能体以自然语言对话为交互入口，接收用户的任意自然语言输入，依托大模型的理解、决策能力，自动调用自身搭载的各类技能、串联完整业务流程、自主处理复杂任务，最终输出贴合用户需求的标准化、场景化回复。

简单来说，普通 AI 只会“回答问题”，而智能体可以“帮你做事”。市面上常见的智能体落地场景十分广泛，覆盖生活、学习、工作等多个领域；典型代表有：智能客服、AI 虚拟伴侣、全能个人助理、英语 AI 外教、数据查询助手、自动化文案生成智能体等。本文开发的天气查询助手，就是最典型的轻量级实用型对话智能体。

抛开表层的对话交互形式，智能体的本质是“大模型驱动的自动化任务执行系统”，其包含三大核心特质，这也是它与普通 AI 对话的主要区别。

1) 认知决策自主化

智能体依托大模型的语义理解能力，可自主识别用户的真实意图，无须用户精准指令、固定格式输入。比如用户输入“今天出门冷不冷”，天气智能体可自主识别用户需要，查询气温、天气状况，而非机械等待固定提问。

2) 技能模块化加载

智能体并非单一问答模型，而是可以自由加载、组合各类技能工具，包括代码执行、API 调用、数据检索、文档解析、联网搜索等。根据不同业务场景自动匹配对应技能，无须人工手动切换。

3) 业务流程自动化

面对复杂任务，智能体可以自主串联多步流程，完成闭环任务。以天气查询智能体为例：自主提取城市参数→调用天气 API→解析原始数据→格式化整理内容→生成人性化回复，全程无须人工干预，自动完成全流程执行。

总之，智能体的本质，就是让大模型从“被动应答”升级为“主动执行任务”，将自然语言需求转化为可落地的业务结果。

1.2.2 智能体的两大类型

工具调用智能体，指以大模型为大脑，以各类外部工具为执行手脚的智能体架构。大模型仅负责理解需求、决策调用哪个工具、整理最终结果；而数据获取、运算、请求、处理等具体落地工作，由代码、API、插件、搜索等工具完成。

深度思考智能体（也叫深思型智能体、Deep Agent）是具备推理、规划、反思、迭代优化能力的高阶自主智能体。区别于普通工具调用智能体的“按流程执行”，它不依赖固定预设流程，面对复杂、模糊、多步骤任务时，会主动进行多层逻辑思考、拆解任务、自主纠错，是真正具备“主动思考、自主决策”能力的 AI 智能体。

简单区分，普通工具智能体是按剧本做事，深度思考智能体是自己想办法做事。工具调用智能体解决的是“能不能干活”的问题，替代固定重复操作；深度思考智能体解决的是“会不会思考、会不会变通”的问题，拥有类人的自主决策与逻辑推理能力，是智能体的高阶形态。



1.3 天气查询智能体开发与部署案例

1.3.1 智能体开发

扣子编程平台的核心价值，就是降低智能体的开发门槛。它提供了一个基于 Web 的 AI 编程开发环境，你只需要通过自然语言描述智能体创意，扣子 AI 就会自动完成智能体的开发、测试、迭代和发布，帮助开发者聚焦核心创意与价值设计，高效实现从创意构思到产品落地。

登录扣子编程平台 <https://code.coze.cn/home>，如图 1-7 所示，输入提示词：帮我做一个天气查询的智能体，支持用户输入任意国内城市名称，获取实时天气数据，能够使用多维度数据展示：返回温度、体感温度、天气状况（晴 / 雨 / 阴等）、湿度、风力风向、空气质量；具有短期预报功能，能展示未来 3 天天气变化趋势；支持口语化提问，如“北京今天天气怎么样”“上海未来三天会下雨吗”。



图 1-7 使用扣子编程开发智能体

单击“提交”按钮后，进入如图 1-8 所示的交互式开发与预览页面，页面左侧会实时显示 AI 编程智能体处理用户开发任务的思考过程和编程开发过程。整个过程无须人工干预，只需几分钟时间就能完成。

开发完成后，右侧预览区域会自动变成 Agent 效果测试界面，如图 1-9 所示。从该页面上方可以看到智能体已经接入了两个工具，并且默认使用大模型 doubao-seed-2.0-lite。

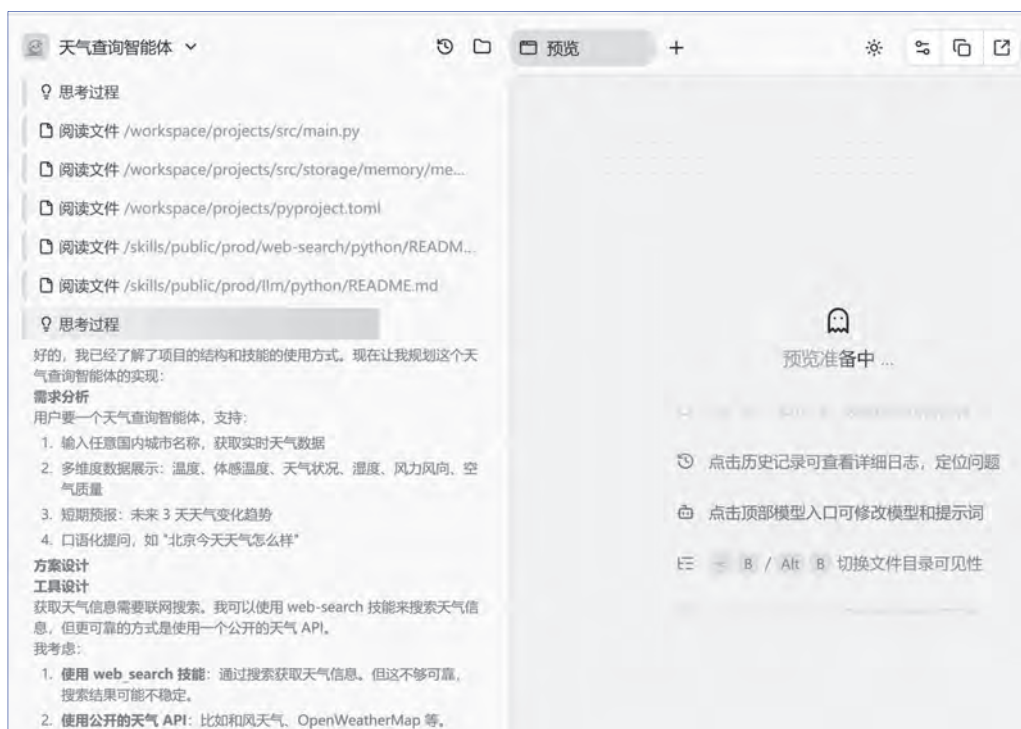


图 1-8 智能体交互式开发与预览界面



图 1-9 Agent 效果测试页面

单击模型名称，右侧会弹出如图 1-10 所示的“agent 节点”配置对话框，从中可以看到工具列表，可以选择模型，也可以修改系统提示词。

如果不需要修改，直接在预览对话框中输入提示词，“今天福州的天气如何？”然后智能体会出现回复，如图 1-11 所示。可以看出天气查询智能体给出了详细答复。



图 1-10 “agent 节点”配置对话框



图 1-11 测试天气查询智能体

整个天气查询智能体的开发就完成了，其开发过程非常简单，只需布置任务给扣子编程平台，告诉你想要做什么，它就能自动完成从设计到开发的全部流程。扣子编程就是一个 AI 驱动的应用开发平台，就是让 AI 去做应用开发，你用聊天的方式指挥它即可。

1.3.2 智能体部署

部署智能体就是将在扣子编程上开发的智能体，快速部署成为全天候在线的正式服务，以便后续可以将智能体的 AI 功能集成到你的应用中。

根据使用需求分成两种部署方案：做成 API 接口部署、网页交互 WebSDK 部署，如图 1-12 所示。这里需要单独开启 WebSDK 渠道，这个功能不会跟着 API 部署自动打开。API 部署的核心目的是把咱们做好的智能体，打包成一串网络接口地址，别的软件、小程序、后台系统，通过代码发请求就能调用智能体的能力。你在扣子做的智能体藏在平台里面，只有在扣子网页上能聊天；开启 API 部署之后，智能体变成一个“远程功能接口”。不管是自己写的手机 APP、电脑软件、网站后台，只要带上密钥、按照平台给的接口格式发消息，就能远程唤起智能体、收发对话。举个例子：你做了

天气智能体，部署 API 后，自己开发的小程序不用再对接天气接口，直接调用扣子 API，用户在小程序输入“查询福州天气”，小程序把这句话传给扣子接口，智能体运算完会把天气数据原路传回小程序展示。

如果不想用代码对接程序，只想把智能体安装在自己的官网或静态网页里，让访客点开网页就能和 AI 对话，就需要单独开启 WebSDK 渠道。WebSDK 就是扣子打包好的一段网页代码，开启渠道后平台生成嵌入代码。把这段几十行的代码

复制到自己的网站源码里，网页侧边、页面内就会弹出智能体聊天窗口。访客不用跳转扣子平台，在网站页面里就能直接提问或同智能体聊天。例如，公司官网右下角悬浮一个天气助手图标，点开弹窗就能查询全国天气，这就是 WebSDK 部署实现的效果。

单击部署后，会出现“开始部署你的项目吧”对话框，单击“开始部署”按钮，如图 1-13 所示。扣子编程将自动进行部署相关操作。你可以在部署页面查看部署的进展和部署日志。部署过程可能需要几分钟。

WebSDK 部署成功，如图 1-14 所示。单击“Web SDK”按钮可以按需求选择嵌入方式（如 JavaScript、React 组件或 Iframe），将智能体嵌入自己的网页中，还需要记住你的 projectId。

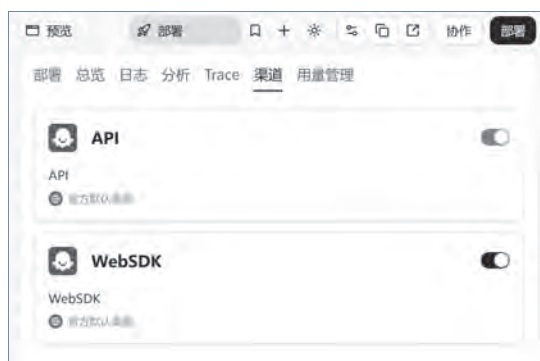


图 1-12 部署渠道



图 1-13 开始部署项目



图 1-14 Web SDK（应为 WebSDK）部署成功

接入 WebSDK 前，应申请获取访问令牌（Token），支持的 Token 类型有 OAuth 访问令牌、个人访问令牌（PAT）和服务访问令牌（SAT）。快速体验可使用个人访问令牌，个人访问令牌是扣子编程提供的短效令牌，用于 API 请求鉴权，可以在扣子编程中生成，如图 1-15 所示。首先在左侧导航栏中选择 API & SDK，然后依次选择“授权”→“个人访问令牌”，最后单击右上角的“添加”按钮。



图 1-15 添加访问令牌

弹出“新的个人访问令牌”对话框，如图 1-16 所示，可以看到个人访问令牌的有效期限。令牌过期后将会失效，无法继续用它来调用扣子 API。根据扣子编程的安全策略，个人访问令牌的有效期限最长可设置为 30 天，不能设置为永久有效。这一限制是为了增强安全性，确保即使令牌泄露也只能在有限的时间内被滥用。过期后，你需要重新生成新的令牌以继续使用相关 API 功能。

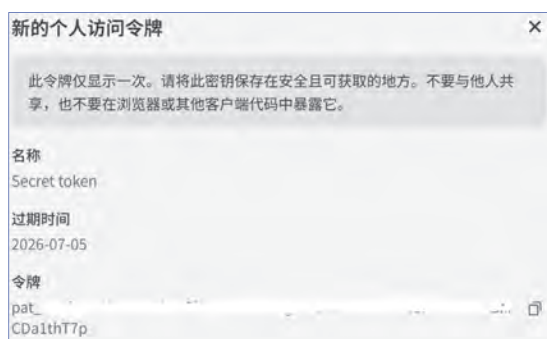


图 1-16 “新的个人访问令牌”对话框

以下是扣子编程 Web SDK 快速接入示例，帮助你在 Web 页面可视化体验智能体。在本地新建记事本，复制以下完整代码，粘贴后保存为 demo.html，并保存至桌面。注意文件后缀必须为 .html，不可为 .txt。

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>我的演示 - 智能体部署</title>
    <style>
      * { box-sizing: border-box; }
      html { margin: 0; height: 100%; }
      body { margin: 0; padding: 8px; height: 100%; }
    </style>
  </head>
  <body>
    <h1>天气查询智能体</h1>
    <!-- 引入 Coze Web SDK 官方 CDN -->
    <script src="https://lf-cdn.coze.cn/obj/unpkg/latest/coze/web-sdk/dist/js-umd/index.min.js"></script>
```



```
<script>
  // 初始化 Web SDK
  cozeWebSDK.init({
    projectId: '你的projectId',
    // 支持同步和异步刷新 token
    refreshToken: () => Promise.resolve('你的Token'),
  });
</script>
</body>
</html>
```

替换鉴权信息，包括你的 projectId 以及 Token，即获取的个人访问令牌。双击打开 demo.html 文件，等待 1~2 秒，如图 1-17 所示，即可在浏览器中直观体验智能体或工作流能力在网页中的呈现效果。



图 1-17 Web SDK 接入示例

总之，在扣子 AI 编程开发的智能体，可部署为 API 服务或 WebSDK。若部署为 API 服务，可以通过 API 方式将智能体的 AI 功能灵活集成到应用中；若部署为 WebSDK，将提供网页嵌入式聊天窗口，用户可直接在网页中与智能体进行对话。



1.4 本章小结

本章讲解扣子编程入门知识，首先介绍扣子编程定位、六大平台核心能力、平台层级架构与个人 / 企业两类订阅套餐，明确积分使用规则；随后讲解智能体定义、本质，区分工具调用智能体与深度思考智能体；最后以天气查询智能体为实操案例，演示自然语言快速搭建、测试智能体的流程，并讲解 API、WebSDK 两种部署方式、个人访问令牌鉴权与网页嵌入实操，建立零代码开发、部署 AI 智能体的完整认知。



1.5 思考与练习

- (1) 扣子编程具备哪些核心平台能力？
- (2) 智能体两种部署渠道 API 与 WebSDK 分别适用什么场景？
- (3) 个人访问令牌有什么使用限制与安全注意事项？
- (4) 简述 WebSDK 嵌入网页的完整操作步骤。
- (5) 个人免费版积分耗尽后平台会有什么限制，如何补充积分？



第 2 章

通过案例熟悉扣子编程



本章导读

从简单的任务入手，在扣子编程平台开发你的 AI 应用，能够快速熟悉扣子编程。无论你是否编程基础，只需要清晰描述你的需求，扣子编程就能为你开发出网页应用、智能体、工作流等 AI 应用。

学习目标

- (1) 掌握扣子编程三类核心项目（网页应用、智能体、工作流）的适用场景，能区分三者的使用区别，掌握项目开发前的基础配置。
- (2) 能够独立撰写清晰完整的需求提示词，完成入门例子实操，学会项目预览测试、自然语言迭代优化、历史版本回滚的调试方法。
- (3) 建立零代码 AI 开发思维，能够根据业务需求自主设计、优化 AI 项目。

本章重点

开发单词拼写大师网页应用、开发产品种草文案 Agent、开发发票信息提取工作流。



2.1 开发网页应用

你可以直接在网页中与扣子编程 AI 平台对话，描述你想要开发的网页应用的功能、界面、逻辑等需求，扣子编程会自动完成应用的开发、测试、迭代和发布。目前，扣子编程支持开发网页类型的应用，包含完整前后端逻辑，例如产品落地页、业务系统、网页版互动游戏、资讯网站等。

2.1.1 输入需求

这个项目的背景是在当代英语教学实践中，单词记忆始终是学习者面临的主要挑战之一。传统的机械记忆方法不仅效率低下，还容易导致学习兴趣的丧失。针对这一普遍性问题，本项目旨在开发一个互动性强、趣味性高的英语单词记忆游戏，通过游戏化学习的方式提升记忆效率。该项目源于作者辅导初中生学习英语时的实际需求。观察发现，常规的单词背诵方法存在三个显著缺陷：记

忆保持率低、学习过程枯燥、缺乏即时反馈机制。游戏化学习恰好能够解决这些问题，通过即时奖励机制、可视化进度反馈和趣味性互动元素，可以有效提升学习者的参与度和记忆效果。

在技术方案的选择上，扣子编程能让普通人都能上手。在扣子编程平台中，单击网页应用选项卡，如图 2-1 所示，在文本框输入你的需求提示词。

给我开发一个初中学生单词拼写学习游戏——要有一个淡蓝色、童趣满满的界面，顶部是游戏名称；左边是“学习区”、右边是“配置面板”；有拼写拖放、中文释义提示、多余干扰字母、鼓励音效；还能导入单词表、管理错题集、调整难度；甚至单击释义还能发音。

游戏怎么玩：

系统给出中文提示，比如“夜晚”。下面会出现几个字母，有的是真正的拼写字母，有的是干扰字母。孩子只要按顺序把字母拖到拼写区，组成正确的单词。拼对了，字母框变绿，还有鼓励音效“棒棒哒！”。拼错了，不会直接扣分，但错误单词会自动进入“错题集”，方便后面专门练习。你可以自己粘贴单词表进去，调整干扰字母的数量，甚至可以一个错题集专练模式。而且它会实时记录答题耗时，最后帮你计算得分，这样孩子会有一种闯关升级的成就感。



图 2-1 扣子编程开发网页应用输入需求的提示词

在扣子编程页面上，你可以上传一些图片或文件，单击“+”按钮，作为附加信息提供给编程 Agent，以便它能更理解你的需求。例如，上传一幅你喜欢的图片等，这些都能帮助编程 Agent 更精准地把握细节要求。

2.1.2 进阶配置

你可以选择协作模式，如图 2-2 所示，默认情况下为 Agent 模式，如果你对需求不确定，可以先切换到对话模式讨论方案，待方案确认后，再切换到 Agent 模式，由编程 Agent 根据此前的讨论结论执行开发任务。

在开发 AI 编程项目时，你可以根据自己的需求，在 Agent 模式与对话模式之间自由切换：

(1) Agent 模式是让 AI 帮你完成工作。在这种模式下，扣子 AI 是你的“执行助手”，能够提取你的指令，自主拆解任务步骤、编写并修改代码、自动调试，实现端到端的任务闭环。



图 2-2 协作模式选择

(2) 对话模式是与 AI 探讨开发思路。在这种模式下，扣子 AI 是你的“资深顾问”，专注于辅助你思考，不会改动你的项目。它主要通过对话帮助你，理清思路，针对你提出的需求进行深度解析，并提供解释、优化建议或示例代码。

另外，在执行编程任务时，编程 Agent 可以按需加载技能，从而具备相应的专业领域知识和能力。如图 2-3 所示，技能列表添加页面，用来给 AI 智能体挂载各类能力，实现 AI 自动调用工具、对接外部软件、生成多模态内容、读写数据。

如图 2-4 所示，你可以选择编程模型用于开发应用，编程 Agent 已集成 GLM 5.1、GLM 5、Kimi K2.5 等主流的编程模型。为了达到较好的开发效果，系统会自动为你选择适合当前场景的模型，你也可以根据模型的官方介绍自行选择。



图 2-3 技能列表

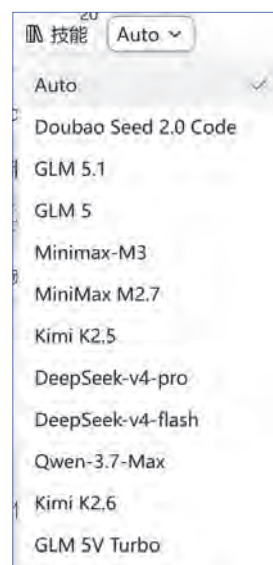


图 2-4 选择编程模型

2.1.3 开发应用

单击“提交”按钮，编程 Agent 会根据你输入的提示词开始设计应用、创建项目，并自动为项目设置应用名称。编程 Agent 收到你的需求之后，将立即启动需求分析，并规划开发流程和步骤，逐步生成应用的前后端代码。

代码生成完毕后，自动构建并启动服务，以提供一个可视化的界面供你预览，如图 2-5 所示，你可以在右侧预览页面查看并测试实际运行效果。

这个应用的雏形就基本搭建完成了，你可以选择直接将其部署为网页站点。通常情况下，这些应用在功能或交互上仍有可改进之处，此时你可以通过自然语言或编写代码的方式和扣子 AI 一起迭代你的应用，如图 2-6、图 2-7 所示，提出修改意见，例如“除了鼠标选择字母，也可以直接键盘按键，拼写正确后，自动进行下一个单词”“导入单词，可以选择一个文本文件来导入。需要滚动条，否则页面下方按钮看不见”等。



图 2-5 单词拼写大师网页应用

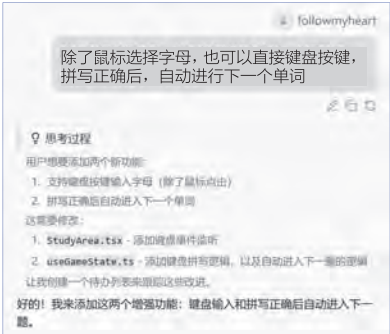


图 2-6 迭代应用，增加键盘输入的需求

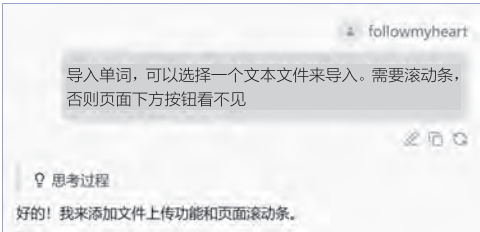


图 2-7 迭代应用，增加导入单词和滚动条的需求

最终完成单词拼写大师网页应用，如图 2-8 所示。



图 2-8 单词拼写大师网页应用界面

从这个案例可以看到，通过与扣子编程对话，我们可以在浏览器页面中快速开发网页应用。无须编写代码、无须搭建开发环境，只需提出明确的开发需求，即可从零开始完成应用的代码编写。扣子编程为开发者提供了这一低门槛、高上限的开发范式，只需一句话，即可一站式完成从构建到落地的完整生产级闭环。



2.2 开发智能体

2.2.1 输入提示词

在扣子编程首页，单击“智能体”选项卡，如图 2-9 所示，在文本框输入你的提示词。你需要尽可能清晰且详细地描述智能体功能、业务逻辑、约束等方面的要求。



提示词完整如下：

图 2-9 开发智能体

帮我开发一个全平台爆款种草文案策划的 Agent，核心功能是基于用户输入的产品名称，自动生成符合“小红书”或“抖音”平台调性的高质量种草文案。Agent 需要具备极强的“网感”，能够熟练使用 Emoji 表情，并提供不同营销角度的文案方案。

核心工作逻辑

接收输入数据：包括产品名称。

风格迁移与润色：

若为小红书：采用“集美/姐妹”口吻，标题必须包含吸引眼球的关键词（如“绝绝子”“按头安利”“无限回购”），正文大量使用 Emoji（如👉👉👉），注重分段和视觉舒适度，文末包含相关话题标签。

若为抖音：采用短视频脚本格式，注重开头的“黄金 3 秒”完播率设计，语言犀利、节奏快，突出反转或强情绪引导。

多方案生成：针对同一个产品，必须输出 3 套不同侧重点的文案（例如：痛点直击型、情感共鸣型、性价比分析型），供用户选择。

约束

拒绝 AI 味：生成的文案严禁出现“综上所述”“总而言之”“这款产品具有……”等生硬的机器翻译腔调，必须口语化、生活化。

Emoji 浓度：小红书风格文案中，Emoji 的占比需适中，每段话至少包含 1-2 个表情符号。

长度限制：小红书正文控制在 200-400 字之间；抖音脚本时长控制在 45 秒口播以内。

单击“提交”按钮。扣子 AI 会根据你输入的提示词来开始设计智能体、创建项目，并自动为项目设置智能体名称。过程预计需要几分钟，请耐心等待。

你可以在左侧实时查看扣子 AI 开发智能体的步骤和详细进展。智能体开发完毕后，你可以查看开发完成总结，查看智能体的主要功能、项目文件路径等信息。

2.2.2 测试智能体

智能体生成后，扣子 AI 会自动生成测试用例并完成一轮单元测试。测试通过后扣子 AI 会提供预览页面，如图 2-10 所示，你可以在右侧预览页面，通过与智能体对话，比如输入“桂花红茶”，并根据提示输入“小红书风格”，测试其表现是否符合预期。



图 2-10 测试智能体

在页面右侧的“预览”区全面测试你的智能体，通常建议关注以下问题：

- 输出效果是否符合业务场景：输出结果是否符合设计要求，例如内容的准确性、格式的规范性、逻辑的正确性等。
- AI 能力是否正常：如果你的智能体集成了 AI 能力，例如文本生成、图片生成等，验证 AI 模型的输出是否符合预期。

如果你在调试智能体的过程中出现页面报错，可以复制报错信息并粘贴到左侧对话框中，将其发送给扣子 AI，要求它修复故障。测试并修复问题之后，如果已经生成的智能体仍有可改进之处，你可以通过自然语言和扣子 AI 一起迭代你的应用。

如果之前某次对话中你输入的描述有误或不准确，导致扣子 AI 生成的智能体无法满足你的需求，相对于回滚版本再重新对话生成智能体，直接修改你发送的历史消息会更加高效。



如图 2-11 所示，你可以在对话记录中找到该历史消息，对其进行修改后按回车键，扣子 AI 会根据新的描述对应用进行调整和优化。同时扣子 AI 的版本记录里也会另起一个分支来记录本次及后续的改变。

你可以为智能体选择不同的大模型，如图 2-12 所示，在智能体开发页面的预览窗口，单击智能体当前使用的模型。在页面右侧弹出的“agent 节点”对话框中，选择要切换的模型。



图 2-11 调整需求迭代智能体



图 2-12 选择模型

各个模型支持调整的参数不同，具体以界面为准。扣子 AI 使用当前的先进模型来完成任务，但由于模型生成代码的随机性，有时可能无法完全满足你的需求，生成了不符合预期的代码，或者出现了反复修复失败的故障，此时你可以使用回滚版本功能，将智能体恢复到之前正常的版本状态。如图 2-13 所示，在智能体开发页面单击版本历史图标，在目标版本右侧单击回滚图标，可以将智能体回滚至该版本。回滚后，扣子编程会自动生成一个新的部署版本，原始的历史记录不会被修改或删除。



图 2-13 回滚历史版本

扣子编程平台把智能体开发的门槛大大降低了，只需要通过自然语言描述智能体创意，扣子 AI 会自动完成智能体的开发、测试、迭代和发布，帮助开发者聚焦核心创意与价值设计，高效实现从创意构思到产品落地。



2.3 开发 workflow

2.3.1 AI 编程开发 workflow

登录扣子编程平台 <https://code.coze.cn/home>，在其首页单击“workflow”选项卡，如图 2-14 所示，在文本框输入提示词“建一个发票读取 workflow，支持上传并解析发票类 PDF 文件，自动提取 PDF 中的关键信息，提取完成后，生成结构化输出结果（支持 JSON 格式或表格格式），确保字段清晰、数据准确。若涉及多页 PDF 发票，需支持批量解析与信息汇总。”你需要尽可能清晰且详细地描述 workflow 的功能、业务逻辑、交互等方面的要求，确保无误后，单击“提交”按钮.



图 2-14 开发 workflow

进阶设置如图 2-15 所示，通过上传附件（上传一些图片或文件，作为附加信息提供给扣子 AI），选择协作模式（默认情况下为 Agent 模式）、添加技能（在执行编程任务时，扣子 AI 可以按需加载技能）、选择编程模型（扣子编程已集成 GLM 5.1、GLM 5、Kimi K2.5 等主流的编程模型，用于开发应用），这些可选的设置，让扣子 AI 生成的结果更精准、更符合你的预期。

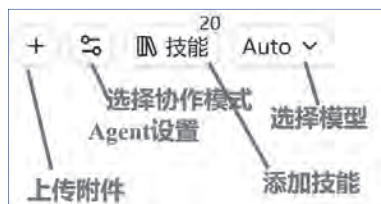


图 2-15 进阶设置

在 workflow 开发过程中，扣子编程 AI 会根据你输入的提示词，自动生成符合你的功能、交互等要求的工作流代码。初步生成代码后，扣子 AI 会自动生成测试用例并完成自动化测试。测试通过后，扣子 AI 将在画布中呈现完整的工作流，如图 2-16 所示，你可直观查看工作流的整体架构、各节点的关联逻辑，这个工作流有五个节点，除了“开始”和“结束”两个节点外，还有三个任务执行节点，分别是“PDF 文本提取”HTTP 请求节点、“发票信息提取”大语言模型节点、“结果格式化”对象存储节点。

扣子编程 workflow 采用语义层、参数层和代码层三层协同驱动架构，基于语义驱动，无须预先定义抽象化节点模板，可高效生成覆盖全流程的节点。所有 workflow 均以开始节点为起点，以结束节点为终点。开始节点用于设定启动 workflow 需要的输入信息，结束节点用于返回 workflow 运行后的结果。任务执行节点是 workflow 的核心功能单元，既可能是单个独立功能组件，也可能是多个关联功能的组合组件。它能覆盖图像生成、数据库读写、第三方工具调用等多种业务场景。

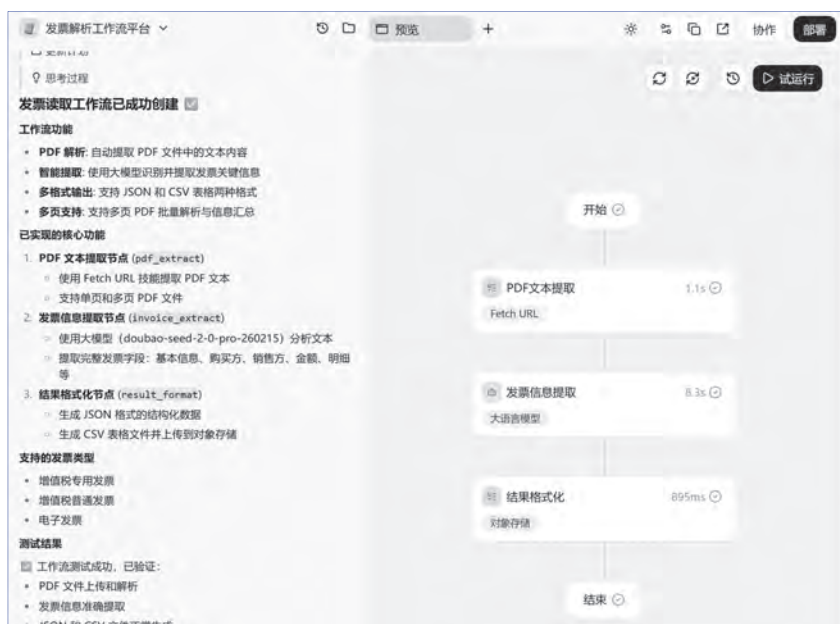


图 2-16 在画布中呈现完整的工作流

单击画布中的任意节点，可以查看该节点的详细信息，包括输入、输出参数及对应的配置等，如图 2-17 所示，单击“发票信息提取”节点，这是大语言模型节点，可以看到系统提示词、用户提示词，选择的模型是“Doubao-Seed-2.0-pro”。



图 2-17 大语言模型节点

2.3.2 试运行及迭代工作流

确认工作流的结构与节点配置初步满足需求后，你可以试运行工作流，对工作流进行全链路验证。主要是验证工作流整体运行状态是否正常，有无执行报错、流程中断等异常情况。另外需要验证各节点间的数据传递链路是否畅通、逻辑交互是否匹配预期，确保上游节点的输出数据格式与下游节点的输入要求完全兼容，且节点的执行顺序、分支逻辑、条件判断等均符合需求。

你可以在画布上单击页面右上方的“试运行”按钮，然后根据页面提示配置合适的参数值，最后再单击页面右下角的“试运行”按钮。在试运行过程中，画布会实时展示运行进度，包括执行状态、单节点消耗时间等信息。试运行完成后，界面上将展示工作流的返回结果，如图 2-18 所示，你还可以单击画布中的任意节点，如“发票信息提取”节点，查看该节点在此次试运行中的详细输入与输出信息。

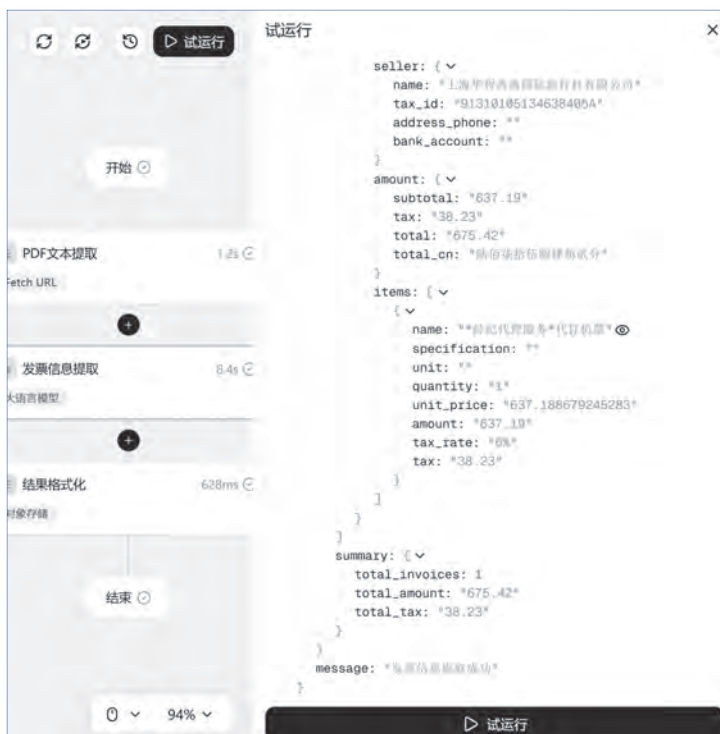


图 2-18 试运行 workflow

如果出现故障，扣子编程 AI 通常会自动识别并提示修复故障，我们可以根据页面提示，允许扣子 AI 尝试修复这些问题。如果在调试 workflow 过程中出现报错，我们可以复制报错信息到对话框中，要求扣子 AI 进行修复。测试并修复问题之后，如果已生成的 workflow 仍有可改进之处，你可以通过自然语言或编写代码的方式和扣子 AI 一起迭代你的 workflow。如图 2-19 所示，例如“需要可以上传多个发票 PDF 文件”，扣子编程 AI 开始修改 workflow。

如图 2-20 所示，这个“批量处理 PDF 文件”节点就是一个封装了大语言模型能力的节点，利用大模型的理解能力，专门用于批量解析发票 PDF 并提取结构化信息。我们上传 3 个发票 PDF 文件来测试，成功识别多个发票文件，提取出金额和开票日期。

扣子编程全新推出 AI 生成式 workflow 功能，核心是将复杂 workflow 搭建流程转化为自然语言描述驱动的高效模式。我们无须编写代码，只需用文字清晰描述业务需求，扣子 AI 即能自动生成对应的 workflow 完整代码，并完成代码测试、需求迭代等全流程操作。扣子编程改变了传统 workflow 搭建逻辑，帮助开发者从“如何分步实现”转向“需要实现什么目标”，大幅降低 workflow 搭建门槛，提升开发效率。

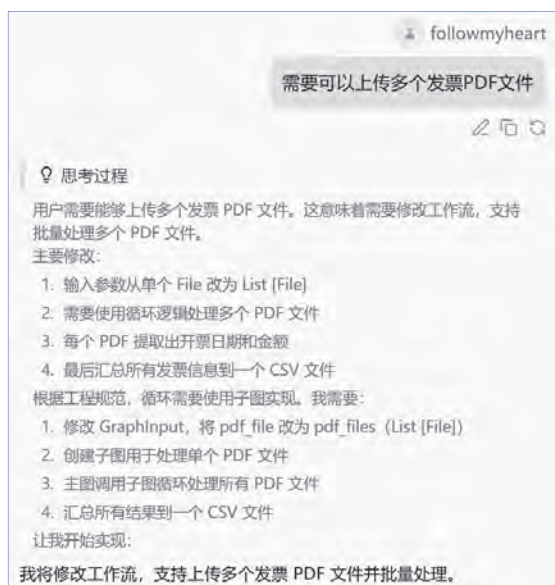


图 2-19 修改工作流



图 2-20 修改后的工作流



2.4 本章小结

本章讲解扣子编程实操入门知识，依次讲解网页应用、智能体、工作流三类项目的完整开发流程。本章以单词拼写游戏网页应用、种草文案智能体、发票解析工作流为实战案例，演示输入需求、自动生成项目、预览测试、自然语言迭代优化、版本回滚的完整操作；介绍工作流节点结构、试运行调试方法，全程依托自然语言即可完成开发，无代码基础也能快速落地各类 AI 项目。



2.5 思考与练习

- (1) 扣子编程两种协作模式（Agent/ 对话）分别适合什么场景？
- (2) 项目效果不达预期，有哪些解决办法？
- (3) 仿写提示词，开发一款小学古诗背诵网页小游戏。
- (4) 设计一个简历 PDF 信息提取工作流的需求描述。
- (5) 简述版本回滚功能的使用场景与优势。



第 3 章 扣子编程的技能开发



本章导读

原生大模型仅具备语言理解、逻辑推理与文本生成能力，没有固定的业务流程、没有统一的输出规范、不懂得何时调用工具、不知道如何结构化处理数据。为了让大模型从“自由对话”转变为“标准化、可复用、可落地的业务执行体”，引入了技能（Skills）机制。扣子编程大幅降低了技能的开发和使用门槛，这个曾经仅被程序员掌握的技术，现在已成为普通人可以轻松上手的工具。

学习目标

- （1）理解技能标准与插件的本质区别，掌握技能标准文件的组成，了解大模型和技能文档的协同工作逻辑。
- （2）熟练运用 Markdown 标题、段落、列表、表格四大核心语法。
- （3）能够结合行业业务自主设计技能完整执行逻辑，搭建可复用的专业领域处理技能。

本章重点

技能介绍、技能的格式与工作机制、Markdown 语法（技能开发基础）、技能开发案例（挑选短线强势涨停股）。



3.1 技能介绍

3.1.1 什么是技能

技能是赋予 AI“做事能力”的最小功能单元。大模型本身只会“聊天、思考、写文字”，它无法联网、无法读取文件、无法调用接口、无法生成 PDF、无法查询实时数据。想要让 AI 从“只会说话”变成“可以执行任务”，就必须给它挂载对应的技能。

通俗的定义，技能是一套写给大模型阅读、结构化封装的标准化作业说明书（SOP 手册），把

某一类任务的执行规则、调用时机、步骤逻辑、输出规范全部写死，AI 拿到这份说明书，就知道遇到什么问题、用什么工具、分几个步骤做、最后产出什么格式的内容。

技能可类比为后厨做菜规程说明书：写明“顾客点番茄炒蛋→先用铲子切番茄插件→调用鸡蛋食材插件→火候控制→成品装盘格式”。

比如你做 A 股研报智能体，东方财富股票接口是插件（只负责拉取股价、财报原始数据），而股票多维度分析技能是说明书（写明拿到数据后，分技术面 / 资金面 / 基本面五步拆解、固定 PDF 报告格式、开头加免责声明）。

3.1.2 技能与插件的本质区别

绝大多数初学者会将技能（Skills）理解为插件（Plugin）、功能工具，这是完全错误的。二者属于两个层级、两种属性、完全不同的运行角色。插件是“执行工具”，属于能力层。技能是“执行规则”，属于调度与规范层。技能与插件具体对比说明如下。

1) 本质不同

插件是平台封装好的可执行程序，是真实的功能工具，例如联网搜索、HTTP 接口请求、PDF 生成、数据解析。插件的本质是代码能力，只会机械完成单一动作，不具备逻辑判断、不具备业务流程、不懂输出格式。

技能是人工编写的结构化文档说明书，是业务逻辑与流程规范。它不直接干活，它只负责告诉大模型：什么时候用工具、怎么用工具、拿到数据后怎么处理、最后输出什么格式的内容。

2) 功能边界不同

插件只能完成单点、底层、固定动作，例如“获取网页数据”“发起接口请求”“生成 PDF 空白文件”。插件无法自主判断业务场景，无法组合多工具，无法结构化输出报告。

技能可以统筹、调度、组合多个插件，将零散的工具能力串联成完整业务流程，例如“调用股票接口获取数据→按技术面、资金面、基本面依次分析→固定报告结构→自动导出 PDF”。

3) 可编辑性与自定义度不同

插件底层代码固定，用户无法修改插件内部运行逻辑，只能被动调用。

技能完全由开发者自定义，可随时修改流程、修改分析维度、修改输出格式、修改约束规则，实现业务快速迭代。

4) 拟人化通俗类比

- 插件 = 工具（锅、铲、烤箱、食材）。
- 技能 = 后厨标准化菜谱、作业指导书。
- 大模型 = 厨师。

工具本身不会做菜，厨师没有菜谱也会乱做、自由发挥；只有依靠技能菜谱，厨师（大模型）才能按照固定步骤、固定标准、固定成品样式完成标准化工作。



3.2 技能的格式和工作机制

3.2.1 技能标准文件结构

在扣子智能体开发框架中，每一个自定义技能都是一套独立、完整、规范的文档工程。一个技能在系统中是独立文件夹，SKILL.md 是唯一必填核心（说明书本体，Markdown 编写），剩余两类为选配。

1) 核心主文件 SKILL.md（必选）

这是技能的本体，也就是 AI 读取的“工作说明书全文”。所有触发条件、参数定义、执行步骤、业务逻辑、输出格式、约束规范全部写在该 Markdown 文件中。大模型运行技能的本质，就是读取并严格执行这份文档的所有条款。

2) 脚本目录 scripts（可选）

存放接口调用、数据处理、自动化脚本等底层执行代码，对应插件能力。

3) 参考目录 references（可选）

存放行业规范、模板案例、参考标准，用于辅助 AI 完成标准化输出。

3.2.2 技能的完整运行机制

大模型负责理解、判断、统筹决策；技能说明书负责定义流程、约束规则、规范输出，两者形成“决策—指令—执行”的完整闭环，完整流程分为五个步骤。

第一步：用户输入需求，智能体识别业务场景

用户向智能体提交自然语言需求，例如输入股票代码、提出选股需求、请求数据分析等。此时大模型首先完成语义理解，识别用户的真实意图、任务类型、输入参数是否合规，判断当前任务属于哪一类业务场景。如果用户输入格式错误、参数缺失、内容违规，大模型会依据基础规则直接拦截并提示用户重新输入。这一步是整个任务的入口，起到场景定位、合法性校验、任务初始化的作用，为后续匹配对应技能流程做铺垫。

第二步：大模型自动匹配对应的技能标准化说明书

智能体挂载的每一项技能，都是一份独立、完整、场景专属的标准化 SOP 作业说明书。大模型会根据识别到的用户场景，自动检索、匹配当前任务唯一对应的技能文档。不同于自由对话模式，进入技能模式后，大模型不再自主构思任务流程，而是强制切换为“执行者模式”，完全以匹配到的技能文档为唯一执行标准。技能说明书中定义的触发条件、任务目标、执行规范，成为大模型接下来所有操作的最高准则，从根源上限制大模型的自由发挥。

第三步：大模型依据技能规则，调度插件工具、传入参数、获取原始数据

这是智能体从“纯文本思考”转向“实际干活”的关键一步。技能说明书中会提前写明完整的工具调用规则：需要调用哪些插件、何时调用、需要传入什么参数、数据来源是什么。大模型严格读取这些既定规则，自动组装用户输入的参数，精准调用对应的插件与公开接口，自动发起数据请求，采集任务所需的全部原始数据。大模型只负责调度统筹，严格遵循技能指令，不随意增减工具、不篡改参数、不跳过任何执行步骤，确保数据采集过程完全标准化、可复现。

第四步：按照技能既定规范，完成数据推理、加工与结构化整理

获取原始数据后，大模型不会自由分析、随意排版，而是严格遵照技能说明书中提前定义的分析维度、筛选逻辑、评判标准、排版结构，对零散原始数据进行清洗、筛选、对比、推理、归纳与重组。

第五步：遵循技能强制格式，输出标准化最终结果

完成数据处理与分析后，大模型最终按照技能说明书中固定的输出模板、分级标题、段落结构、字段规范、免责声明等强制格式，生成排版工整、结构统一、内容规范的最终成果。

整个运行机制可以凝练为一句核心原理：技能负责定义全部流程与规范，是作业指导书；大模型负责阅读理解说明书、按规推理输出，是执行大脑。



3.3 技能开发基础：Markdown 语法

技能本质是文档化的业务规则，因此必须依托一套统一、标准、AI 高适配的排版语法，即 Markdown 语法。Markdown 是整个零代码智能体开发的底层基石，是 AI 可精准解析、可强制约束、可标准化渲染的专用文稿语言。

不同于自由文本，Markdown 可以通过固定标记，让大模型区分“规则层级、强制要求、结构框架、输出模板”，从根源上避免 AI 自由发挥、输出混乱、结构错乱。

Markdown 是一种轻量化、结构化、易读写的标记语言，用少量简易符号实现文档分级、加粗、列表、分段，无须复杂排版代码，编写的文档人类可读、大模型可精准解析，是扣子平台技能说明书（SKILL.md）的专用编写格式。

简单说，Markdown 无须复杂编程代码，仅通过简易固定符号，就能实现文档结构化排版，既方便开发者编写技能说明书，也能被大模型精准识别、严格执行格式规则。在智能体开发中，技能本质是标准化任务说明书，而 Markdown 是这份说明书的核心载体，其核心作用并非页面美化，而是规则约束。

若使用纯文字编写技能规则，大模型会自由发挥，导致每次生成的股票分析报告结构混乱、排版不一、内容零散，无法实现标准化输出。而 Markdown 可以固定任务层级、执行步骤和输出格式，强制大模型统一执行逻辑，让智能体每次运行效果一致，真正实现零代码智能体的标准化开发。针

对零基础学习者，本节聚焦智能体开发最核心、最常用的三类语法：标题语法、段落语法、列表语法，结合股票分析智能体实操场景，搭配通俗案例，可直接上手复用。

1. 标题语法（构建报告整体框架）

标题语法是 Markdown 最核心的基础语法，主要用于划分文档层级、搭建技能任务和分析报告的整体框架，强制大模型按固定模块分区输出内容，杜绝内容杂乱、章节混淆。语法规则简单固定：通过“#”符号区分层级，符号数量越少，标题层级越高，所有“#”符号后必须加英文空格，语法方可生效。该语法完美适配股票分析报告的多维度分级结构。

实操语法与效果示例如下：

```
原始代码：
# A 股个股智能分析报告（一级标题：全文总标题）
## 1. 技术面分析（二级标题：核心分析模块）
### 1.1 K 线与均线指标研判（三级标题：细分分析维度）
```

实际作用：自动生成层级清晰的结构化框架，约束 AI 严格按照“总报告—核心模块—细分维度”的逻辑输出内容，是标准化报告的基础框架。

2. 段落语法（规整正文与版式规范）

段落语法是保障技能说明书、最终分析报告版式整洁、可读性强的基础语法，主要用于正文内容排版、内容分区、规则备注，解决纯文字堆砌、排版混乱、段落无区分的问题，适配小白零基础排版。核心包含分段、重点强调、备注引用、行内参数标注四类常用规则，适配所有智能体文案输出场景。

实操语法与效果示例：

（1）分段规则：段落与段落之间空一行，自动区分独立内容板块。

示例：写完技术面分析正文后，空一行，再编写资金面分析正文，两大模块自动分区，版式整洁。

（2）重点强调规则：使用两对星号（** 文字 **）包裹文字，标注强制约束规则。

示例：** 风险提示：本报告仅基于公开历史数据做客观统计分析，不构成任何投资建议 **。

（3）备注引用规则：使用“> 英文空格”编写引用内容，单独放置免责声明、补充备注。

示例：> 股市有风险，投资需谨慎，所有分析结果仅供学习参考。

（4）行内参数标注规则：使用单个反引号 `内容` 标注专业参数、接口字段。

示例：本智能体核心入参为 `stock\_code`，仅支持 6 位 A 股数字代码。

3. 列表语法（拆解流程与罗列指标）

列表语法用于拆解复杂业务逻辑、罗列数据指标、规范执行步骤，让零散的内容条理化、有序化，既方便小白梳理开发流程，也能降低大模型的理解偏差，让工具调用、数据分析、报告生成的流程更标准。列表分为无序列表和有序列表两种，均需在符号后添加英文空格。

1) 无序列表（适用于罗列指标、要素、条件）

语法：以“- +英文空格”开头，逐条罗列平行内容。

实操示例：

- 核心技术指标：MA5、MA10、MA20 均线
- 辅助分析指标：MACD、KDJ、成交量
- 核心数据来源：东方财富、新浪财经公开接口

2) 有序列表（适用于规范步骤、流程操作）

语法：以“数字+. +英文空格”开头，按顺序罗列操作步骤。

实操示例：

1. 接收用户输入的 6 位 A 股股票代码
2. 调用公开接口采集股价、K 线、财报、新闻数据
3. 多维度解析数据，生成标准化分析报告
4. 自动导出 PDF 格式报告文件

4. 表格语法

标准 Markdown 表格由表头分隔行 | --- | --- | --- | 实现列对齐，竖线 | 划分单元格，结构分为表头行、分隔行、内容行三部分，分隔行短横线搭配冒号可控制对齐：: 居左、---: 居右、---: 居中；在 Coze 技能文档中，一般统一使用简易居中格式。

```
| Syntax | Description | Test Text |
| :--- | :---: | ---: |
| Header | Title | Here's this |
| Paragraph | Text | And more |
```

呈现出来的输出格式如图 3-1 所示，可以看到输出的表格中控制对齐后的效果。

Syntax	Description	Test Text
Header	Title	Here's this
Paragraph	Text	And more

图 3-1 表格语法输出结果

对于零基础开发者，需牢记核心规范：所有 Markdown 标记符号后必须添加英文空格，否则语法失效，会导致 AI 解析错乱、报告排版崩盘。同时，扣子编程的技能开发无须掌握图片、超链接、复杂表格等冗余语法，熟练掌握上述标题、段落、列表、表格四大核心语法，即可完成所有技能的规范化编写与标准化报告输出。

Markdown 凭借极简标记符号，实现文档结构化编写，是技能说明书标准化落地的底层基础；开发者通过标题、加粗、列表等语法，把业务 SOP 固化成机器可读文档，最终实现 AI 报告格式统一、输出可控。



3.4 技能开发案例：挑选短线强势涨停股

3.4.1 技能的设计

挑选短线强势涨停股属于典型的短线选股策略，而且是 A 股市场里主流的短线博弈思路。这种股票极少长期持有，完全区别于价值投资、中线波段。涨停代表当日资金抢筹、做多情绪达到极致，强势封板更是资金认可度高、抛压小的信号。策略核心是跟随市场热度、资金流向与情绪，而非分析公司基本面。

选股逻辑，首先获取交易日当日涨停股票池，可以在东方财富网站涨停板行情查询，也可以通过调用开源金融数据接口如 AKShare 来自动化获取；然后分析涨停股的封板强度，比如封单资金、成交额（不低于 1 亿元）；再分析涨停股换手率，不能过低或过高，换手率处于 10%~20% 的涨停股票是最好的，还有就是所属板块整体强度。经过这样层层筛选，可以从涨停股里识别出优质的短线标的。

本项目借助扣子编程平台，通过自然语言描述选股规则 and 数据处理逻辑，由扣子编程来自动完成技能的开发。这种方式显著降低了技术门槛。现在把我们的逻辑策略提炼为一组自然语言指令，最终设计的提示词如下：

我想开发一个用于识别沪深 A 股市场中短线强势封板股的技能，编写 Python 脚本，请自行安装并调用开源金融数据接口 AKshare 等依赖库。

技能执行步骤如下：

步骤 1：获取当日涨停股池

根据用户输入的交易日期，获取当日全部沪深 A 股涨停股票（包含非 ST、非退市、非停牌）。

步骤 2：基础过滤（排除弱势股）

对涨停股池执行以下过滤：

剔除 *ST、ST、退市整理期 股票

剔除 成交额 ≤ 1 亿元 的股票

剔除 涨停时间 $\geq 14:30$ 的尾盘板

剔除 当日开板次数 ≥ 2 次的烂板股

步骤 3：计算封板强度指标

对每只股票统一计算：

封单金额（万元）

成交额（亿元）

封板成交比 = 封单金额（万元）/ 成交额（万元）

（统一单位，比值越高代表封板越强）

步骤 4：剔除极端值 + 取前 15 强

计算全部股票封板成交比

剔除 前 5% 极高值 和 后 5% 极低值（避免异常单干扰）

按封板成交比从高到低排序，保留前 15 名

步骤 5：换手率稳健过滤（优化版）

在 15 只股票中保留：

换手率 $\geq 5\%$ 且 $\leq 25\%$

步骤 6：板块强度量化评估

保留股票必须满足以下任意一条：

所属行业板块 涨幅 $\geq$ 市场平均涨幅

所属概念板块 涨幅 $\geq$ 市场平均涨幅

所属板块 上涨家数 $\geq$ 下跌家数

剔除板块走弱、无板块效应的个股

步骤 7：最终输出

返回最终符合条件的强势封板股列表，字段如下：

股票代码

股票名称

成交额（亿元）

封单金额（万元）

封板成交比（保留 3 位小数）

换手率（%）

所属行业板块

所属概念板块

涨停类型（首板 / 2 连板 / 3 连板及以上）

3.4.2 技能的开发

首先切换到扣子编程的“技能”选项卡，进入技能的开发页面。技能有两种开发方式。

- 第一种方式，如果你用其他方式完成了技能的开发并打包成压缩文件，然后在扣子编程技能开发页面，选择左下角的第三个图标，如图 3-2 所示，上传技能压缩包。



图 3-2 上传技能文件包

- 第二种方式，就是在技能开发页面的对话框中直接输入你的需求，扣子编程平台会自动解析你的请求，进行相应的开发工作。如图 3-3 所示，切换到“技能”选项卡，把写好的提示词粘贴到扣子编程平台的对话输入框，单击“提交”按钮。

扣子编程平台就会帮你自动开发，最后返回技能成功创建的提示，如图 3-4 所示，创建好脚本，并且测试结果成功。

在预览区域，用户可以通过自然语言交互方式对技能进行测试，如图 3-5、图 3-6 所示，输入“2026 年 6 月 5 日”这个交易日，技能加载完成，自动执行脚本，最后返回精选股票列表，格式清晰、内容完整，达到预期。



图 3-3 编写技能

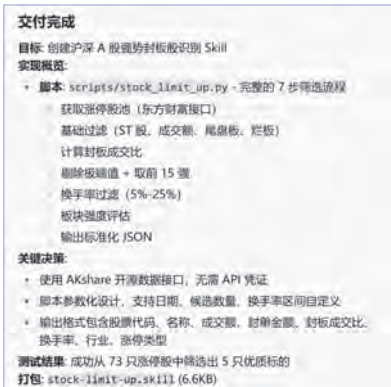


图 3-4 技能编写成功



图 3-5 测试技能



图 3-6 测试结果输出

为了验证结果的准确性，我们选择其中一只股票“三星新材”，在东方财富网的涨停板行情页面进行人工比对，如图 3-7 所示，对比结果一致，该股确实为当日涨停，成交额、封板资金、换手率和技能输出一致。

代码	名称	涨跌幅	最新价	成交额	流通市值	总市值	换手率	封板资金	首次封板时间	最后封板时间	涨停次数	涨停统计	连板数
603578	三星新材	10.00%	15.51	1.35亿	27.97亿	36.3亿	5.03%	7065万	14:22:24	14:22:24	0次	1/1	首板

图 3-7 验证结果

3.4.3 技能的部署

在扣子编程中创建的自定义技能，必须部署之后才能在扣子对话中使用。默认情况下，自定义技能只能被所有者本人使用。技能的部署非常简单，直接单击预览区域右上方的“部署”按钮打开“部署”对话框，即可开始部署，如图 3-8 所示。



图 3-8 部署技能

扣子编程会自动执行技能的打包、构建和部署，部署成功后。如图 3-9 所示，在“扣子技能商店→我的技能”中，查看你已经创建并发布的自定义技能。



图 3-9 发布好的自定义技能

与可以直接运行的智能体不同，技能本质上是一个“专家领域经验包”，它需要被加载到智能体中才能发挥作用。在扣子编程创建了自己的技能之后，你可以在扣子对话中触发技能，体验技能的效果。

在扣子首页文本框中单击“+”按钮，如图 3-10 所示，并在技能页签中，选择要使用的技能“短线涨停识别技能”。

如图 3-11 所示，在对话中指定要使用的技能，输入“2026 年 6 月 8 日”，结果符合预期。



图 3-10 单击“+”按钮打开菜单

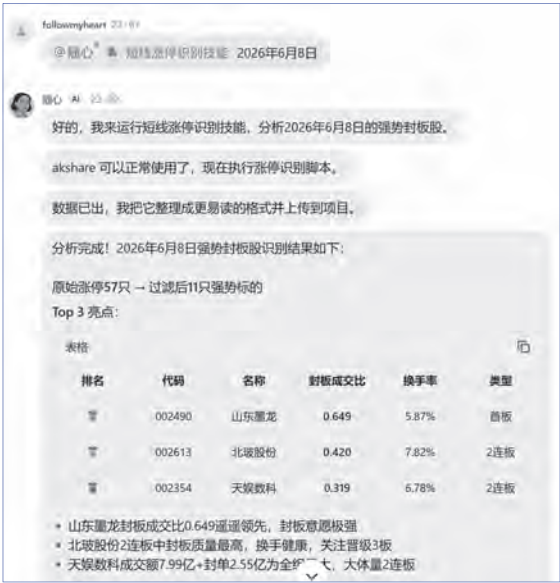


图 3-11 使用技能

总之，扣子编程支持通过自然语言直接开发技能。你只需要描述技能的目标、使用场景、执行流程和预期输出，扣子 AI 会自动设计技能结构，生成 SKILL.md 等标准技能文件，创建脚本、参考文档和资源文件等，将技能打包为可直接运行和部署的技能包。整个过程无须手动编辑文件。哪怕你不会写代码，也能在扣子编程中开发技能。



3.5 本章小结

本章围绕扣子编程技能（Skill）展开讲解，首先区分技能与插件的差异：插件是底层执行工具，技能是标准化业务流程说明书。其次介绍技能标准文件结构、五步完整运行机制，以及开发技能必备的 Markdown 核心语法（标题、段落、列表、表格）。最后以 A 股短线强势封板股识别技能为实操案例，演示技能创建方式、测试校验、部署上线、挂载至智能体使用的全流程，帮助读者掌握自定义领域技能开发，拓展智能体业务处理能力。



3.6 思考与练习

- (1) 简述技能和插件的本质区别，用生活化例子说明。
- (2) 技能完整运行分为哪些步骤？
- (3) 自定义技能开发完成后，为什么需要部署，如何调用？
- (4) 为什么开发专业领域智能体需要搭配自定义技能，仅靠大模型对话有什么缺陷？



第 4 章

一段提示词造智能体：小红书爆款笔记生成

本章导读

在自媒体行业高速发展的当下，小红书已成为大众分享生活、知识、好物的核心平台，优质的笔记内容是账号涨粉、引流、变现的核心基础。但对于自媒体新手、兼职博主、内容运营人员而言，持续创作符合平台调性、具备爆款潜质的小红书笔记存在诸多难点：一是难以把握平台用户喜好与文案风格；二是封面创意构思耗时久；三是批量创作不同主题笔记效率低下。扣子编程作为一站式 AI 智能体开发平台，依托大语言模型能力，支持用户通过自然语言完成智能体的设计、开发、测试、迭代与部署，无须深厚编程基础即可打造专属业务工具。

学习目标

- (1) 掌握智能体项目前期规划方法，能够拆分基础功能、风格、输出、合规四类需求。
- (2) 熟练编写结构化智能体开发提示词，能够精准定位文案、封面、标签、风格等输出缺陷，并能撰写清晰可落地的自然语言迭代指令。
- (3) 总结文案类智能体通用开发技巧，理解本案例开发逻辑的复用思路。

本章重点

以“输入任意主题，自动生成含标题、正文文案、封面设计方案的小红书爆款笔记”为目标，结合扣子编程官方规范，从零到一完整讲解智能体开发、测试、多轮迭代、功能优化的全流程，适合零基础开发者、自媒体从业者学习参考。



4.1 项目规划与需求梳理

本章开发的小红书爆款笔记生成智能体，面向自媒体博主、内容运营、学生、普通用户等群体。用户仅需输入单一主题（如减肥、居家收纳、考研备考、平价护肤、短途旅行等），智能体自动输出一套完整的小红书笔记内容，包含爆款标题、分段式正文文案、专属封面设计方案，内容贴合小

红书主流风格，具备高点击率、高互动性的特点。智能体依托内置大语言模型实现内容创作，全程通过自然语言完成开发与修改，无须编写复杂代码。

4.1.1 核心功能需求

结合小红书平台爆款笔记的创作逻辑、用户阅读习惯以及平台规则，同时参考扣子智能体开发规范，制定详细功能需求，分为基础交互功能、风格与合规约束、输出格式约束三大模块，为后续编写开发提示词奠定基础。

1. 基础交互功能

用户输入纯文本主题（不限领域，生活、学习、美妆、健身、旅行、美食等通用主题均可），智能体识别主题后，一站式输出全套笔记内容，无须二次追问。支持全品类主题适配，无领域限制。

1) 标题创作要求

遵循小红书爆款标题逻辑，融合痛点提问、数字总结、情绪引导、悬念引流四大主流形式，标题字数控制在 15~20 字，可搭配 Emoji，拒绝生硬营销感，避免标题党与平台违规词汇，提升笔记打开率。针对不同主题匹配对应风格：健身减肥类突出效果与实用性，学习类突出干货与效率，生活类突出共鸣感。

2) 正文文案要求

整体采用小红书“姐妹式”口语化口吻，亲切自然，符合平台用户交流习惯；正文分段排版，段落简短，避免大段密集文字，每段搭配 1~2 个 Emoji 提升视觉舒适度；内容逻辑清晰，开篇吸引注意力，中间输出核心干货、经验、心得，结尾添加互动引导（提问、点赞收藏、评论互动等）；正文字数控制在 300~600 字，适配手机端阅读；文末附带 3~5 个精准话题标签（#），标签贴合主题，兼顾热门标签与垂直小众标签，提升笔记流量。

3) 封面设计方案要求

小红书为视觉优先平台，封面直接决定笔记点击率，因此智能体需针对输入主题，输出可落地的封面设计详细方案，包含封面主体画面、配色方案、文字内容、字体风格、构图方式四大板块。区分不同主题的封面风格：减肥、健身类主打简约干净风，学习类主打清新治愈风，美食旅行类主打温馨氛围感，美妆类主打精致质感风；同时区分实拍图、网图、图文拼接等不同封面形式，给出具体落地建议。

2. 风格与合规约束

拒绝 AI 生硬话术，文案生活化、真实化，模拟真人博主分享口吻；规避小红书平台违规词汇以及夸大宣传语句；内容积极正向，不输出低俗、虚假、违规内容；整体风格统一，标题、文案、封面调性保持一致。

3. 输出格式约束

内容分区清晰，固定分为【爆款标题】【笔记正文】【封面设计方案】三大板块，板块名称明确，

内容层级分明，方便用户直接复制使用。

4.1.2 开发前的准备

打开扣子编程平台（<https://code.coze.cn/>），这里建议读者升级为个人进阶版套餐，如图 4-1 所示，它适合有一定使用频率的爱好者和学习者。

如果你是高频使用者或独立开发者，需要更大的知识库空间，需要 Seedance2.0 完成视频创作，需要高速通道提高视频排队速度，则建议升级为个人高阶版套餐。



图 4-1 订阅管理

此外还有一项开发技能，我们可以查看到目前为止自己开发的网页应用、智能体、工作流的积分消耗，如图 4-2 所示。

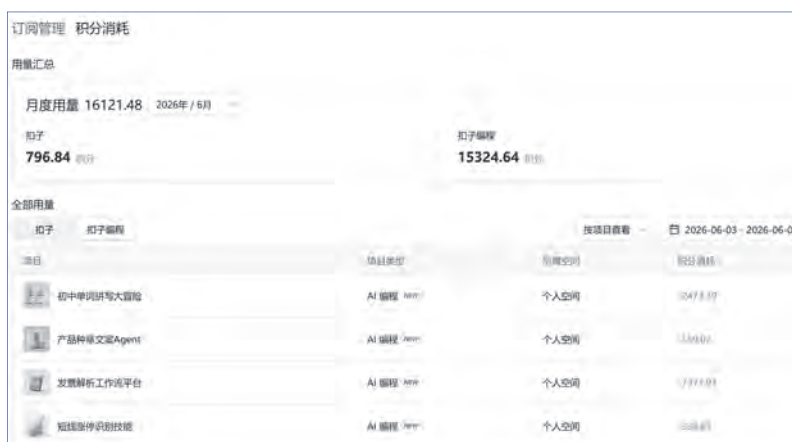


图 4-2 积分消耗明细



4.2 开发智能体

本节案例开发的整体流程遵循扣子官方标准开发流程：输入完整开发提示词→AI 自动搭建智能体→自动单元测试→人工预览测试→发现问题并通过自然语言迭代优化→多轮调试修复细节问题→最终定型。全程以自然语言作为交互方式，不手动编写代码，重点把控提示词质量与迭代指令的精准度。

4.2.1 第一轮开发：编写提示词生成基础智能体

提示词是智能体功能的核心，描述越详细、逻辑越清晰，扣子 AI 生成的智能体越贴合需求。结合前文梳理的功能需求，编写完整开发提示词，在扣子编程首页“智能体”选项卡的文本输入框中粘贴以下内容（见图 4-3），这是第一轮开发的核心指令。

请帮我开发一款小红书爆款笔记生成智能体，具体需求如下：

一、核心功能：用户输入任意创作主题（如减肥、考研、护肤、旅行、居家收纳等），智能体自动生成一套完整的小红书爆款笔记，输出内容固定分为三大板块：爆款标题、笔记正文文案、封面设计方案。

二、标题创作规则：

1. 标题采用小红书主流爆款风格，结合痛点、数字、情绪、悬念等元素，字数控制在 15-20 字，可搭配 Emoji。

2. 拒绝标题党、夸大宣传，语言亲切自然，提升笔记点击率。

三、笔记正文规则：

3. 口吻采用小红书通用“姐妹”口语化风格，模拟真人博主分享，杜绝 AI 生硬句式。

4. 正文分段排版，段落简短，每段搭配 1-2 个 Emoji，整体字数 300-600 字。

5. 内容结构：开篇吸引眼球→中间输出干货 / 经验 / 心得→结尾添加互动引导。

6. 文末附带 3-5 个小红书话题标签，标签贴合主题，搭配热门标签与垂直标签。

四、封面设计方案规则：

7. 针对输入主题，详细描述封面主体画面、整体配色、封面文字、构图方式四大内容。

8. 结合主题匹配风格：健身减肥类简约干净，学习类清新治愈，美食旅行类温馨，美妆类精致。

9. 方案具备可落地性，普通用户可根据方案直接制作封面。

五、通用约束：

10. 内容积极正向，规避平台违规词汇、虚假宣传语句，符合小红书平台规则。

11. 标题、正文、封面三者风格统一，整体具备爆款潜质。

12. 输出格式固定分区，依次标注【爆款标题】【笔记正文】【封面设计方案】，层级清晰。

六、交互规则：仅接收用户输入的主题文本，直接输出完整笔记内容，无须额外追问。

确认提示词无误后，单击文本框下方“提交”按钮，扣子 AI 将自动开始智能体开发工作。平台会在左侧对话区实时展示开发步骤，包括需求解析、模型配置、功能封装、单元测试等环节，整个过程耗时 3~8 分钟，无须人工干预。开发完成后，左侧对话区会弹出开发完成总结，如图 4-4 所示，展示智能体名称、核心功能、项目文件路径、默认使用模型等信息。平台会自动执行一轮单元测试，验证智能体基础运行逻辑，若基础功能无故障，则直接进入预览与测试环节。

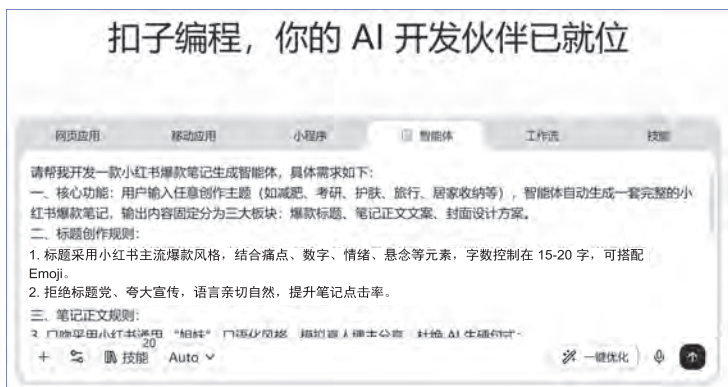


图 4-3 开发智能体

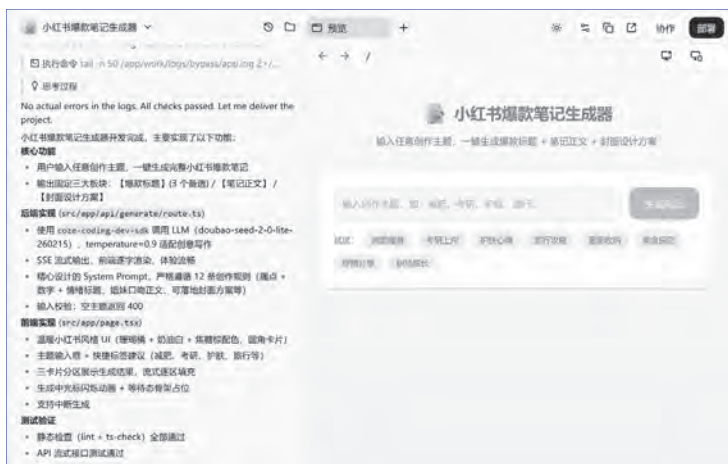


图 4-4 生成智能体

智能体生成完成后，预览区即为交互测试界面，我们选取减肥作为测试主题，输入“减肥”并发送，测试基础输出效果，如图 4-5 所示。

第一轮测试发现的问题如下：标题单一化，仅采用常规描述形式，缺少数字、痛点等爆款元素，吸引力不足；正文内容同质化，内容过于笼统，干货偏少，偏向泛泛而谈，没有针对“减肥”主题细化内容；封面方案过于简单，描述笼统，仅说明基础画面与配色，没有区分“对比图、场景图”等小红书主流封面形式；格式细节瑕疵，话题标签仅 3 个，部分热门标签缺失；风格稳定性不足，若切换“考研”“护肤”等主题，则部分输出会出现语句生硬、AI 痕迹明显的问题。



图 4-5 测试智能体

第一轮智能体仅实现了“基础内容生成”的核心功能，但距离“爆款笔记”的创作要求还有较大差距；接下来进入第一轮迭代优化环节，通过自然语言向扣子 AI 下发修改指令，针对性修复上述问题。

4.2.2 第二轮迭代：针对性优化内容细节与创作逻辑

基于第一轮测试记录的问题，在左侧对话区输入自然语言迭代指令，明确修改方向、优化标准，指令要求具体、可落地，避免模糊描述。本次迭代聚焦标题多样性、正文干货、封面细节、标签优化四大核心问题，迭代提示词如下：

针对当前小红书笔记生成智能体进行迭代优化，修改要求如下：

1. 标题优化：丰富标题形式，除常规描述外，增加数字总结、痛点提问、悬念三种爆款形式，标题 Emoji 搭配更贴合主题，进一步提升点击率。
2. 正文优化：增加内容干货，针对不同主题输出具体经验、技巧，拒绝笼统话术；丰富结尾互动形式，除提问外，增加“求收藏”“打卡互动”等形式；严格拆分段落，每段控制在 2-3 句话。
3. 封面方案优化：细化设计内容，区分“前后对比图、场景实拍图、文字主图”三种小红书主流封面类型，分别给出设计建议，补充字体选择（圆润字体 / 粗体字）、文字位置等细节，提升方案落地性。
4. 标签优化：每个笔记话题标签增加至 4-5 个，分为 1 个大热门标签 + 3-4 个垂直细分标签，扩大流量范围；
5. 风格优化：强化真人博主口吻，彻底弱化 AI 生硬感，所有主题保持风格统一。

如图4-6所示，发送迭代指令后，扣子 AI 会自动解析需求，对智能体的提示词、内容生成逻辑进行调整，同时保留原有基础功能，迭代过程约 2~5 分钟。迭代完成后提示“已完成 5 项迭代优化”，再次在预览区进行测试，依旧使用减肥作为核心测试主题。



图 4-6 第二轮测试

本轮优化效果显著，标题加入数字元素，正文干货充足，封面方案细化分类，标签数量达标，AI 生硬感明显减弱，跨主题（考研、护肤）测试风格统一。但同时发现新的细节问题，也是小红书笔记运营中的高频痛点：内容同质化隐患，主题适配灵活性不足，没有结合小众场景定制内容，情绪氛围单一，缺少情感共鸣，不符合情感类笔记的创作逻辑。

4.2.3 第三轮迭代：增加多风格模式与内容多样性

经过前两轮优化，智能体基础功能已趋于稳定，第三轮迭代聚焦提升内容多样性、小众主题适配、多风格切换三大进阶需求。在左侧对话区输入迭代提示词指令，新增“风格选择”功能、同主题多版本生成能力、小众主题专属适配逻辑，提示词如下：

继续对智能体进行进阶迭代，新增功能与优化要求如下：

1. 新增风格自适应功能：智能体自动识别主题类型并匹配对应文案风格。干货技巧类（减肥、考研、收纳）走实用干货风；情感分享类（独居感悟、情绪治愈）走温柔共鸣风；美食旅行类走活泼种草风，区分不同赛道的文案调性。
2. 提升内容多样性：同一主题多次生成，标题、文案框架、封面方案避免重复，至少区分 2 种不同创作思路，解决同质化问题。
3. 小众主题适配优化：针对小众、细分主题，放弃通用模板，结合场景定制封面与文案，强化场景代入感。
4. 细节补充：在正文末尾可根据主题灵活增加“小贴士”板块，补充注意事项、避坑指南，进一步提升笔记价值。
5. 保留前两轮所有优化内容，不改动已有正常功能。

发送迭代指令后，扣子 AI 会自动解析需求，对智能体的提示词、内容生成逻辑进行调整，如图 4-7 所示。迭代完成后，提示“已完成 4 项进阶迭代”，再次在预览区进行测试，这次选取情感（独居治愈）类型主题，全面验证新增功能。测试结果显示，智能体已具备完善的场景适配能力、内容多样性能力。



图 4-7 第三轮测试

你可以继续输入一些改进小建议，如“根据封面设计方案，可以直接生成配图吗”，迭代完成后，如图 4-8 所示，已经可以直接生成配图。



图 4-8 封面设计方案，生成配图

4.2.4 后续长期迭代与维护方案

回顾整个开发流程中出现的所有问题：格式错误、内容同质化、风格生硬、封面方案笼统等，目前均已通过迭代修复，智能体正式开发完成。

可以使用历史版本回滚功能，如图 4-9 所示，显示所有的修改迭代历史版本记录，该操作可保障后续迭代出错时，快速恢复到稳定版本，降低开发风险。

智能体并非一次性开发完成即可永久使用，后续需要结合小红书平台规则变化、用户需求升级，制定长期迭代计划，依托扣子编程的迭代能力持续优化：

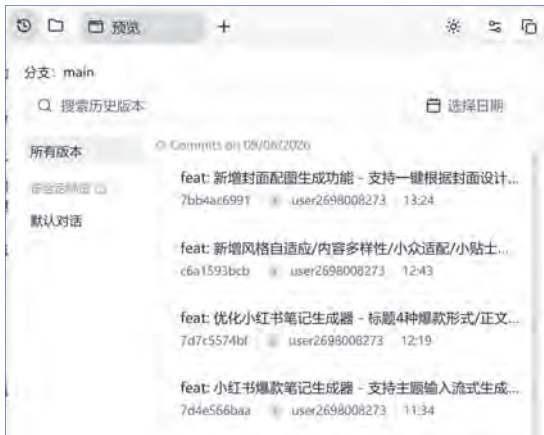


图 4-9 历史版本

- 平台规则适配迭代：若小红书更新内容规则、限流词汇，及时通过自然语言修改提示词，更新违规词库，保证内容合规。
- 新增功能迭代：根据用户需求，逐步新增“多版本笔记生成”“笔记避雷点提醒”“文案翻译成英文（跨境小红书）”等拓展功能。
- 数据反馈优化：结合实际发布笔记的点赞、收藏数据，针对性优化标题风格、互动话术，持续提升笔记爆款率。
- 季度版本更新：每季度归档一个稳定版本，清理历史冗余版本，保障智能体运行流畅。



4.3 开发总结和复盘

本章案例使用扣子编程开发小红书爆款笔记生成智能体，严格遵循平台官方开发流程，整体分为“前期需求规划→首轮提示词开发→功能迭代”等阶段，全程无须编写代码，仅依靠自然语言完成所有开发与修改工作，充分体现了扣子编程“低门槛、全流程 AI 辅助”的核心优势。最终成品实现了“输入任意主题，自动生成标题、文案、封面全套小红书爆款笔记”的核心目标，覆盖干货、情感、美食、学习、小众细分等全场景，满足个人博主、内容运营的日常创作需求。

从开发周期来看，零基础开发者完成全套开发+多轮迭代，总耗时不超过 30 分钟，相较于传统代码开发工具，效率提升数倍，真正实现了“创意快速落地”。

1. 开发流程中遇到的典型问题与对应解决方案

下面梳理一下开发流程中遇到的典型问题与对应解决方案，为同类智能体开发提供参考：

1) 提示词描述模糊，智能体功能跑偏

问题：第一轮初期提示词过于简单，导致输出内容不符合预期。

解决方案：编写提示词时分模块拆解需求，将格式、风格、约束、输出内容逐一明确，越细化的提示词，生成效果越精准。

2) 迭代指令不具体，优化效果差

问题：初期使用“优化文案风格”等模糊指令，AI 无法精准定位问题。

解决方案：迭代指令必须描述问题+给出具体标准，例如不说“优化封面”，而是说“封面分为对比图、场景图两类，补充字体与文字位置”。

3) 同一主题内容同质化

问题：模型创意不足，多次生成内容重复。

解决方案：在提示词与迭代指令中明确“多创作思路区分”，同时适度调高模型随机性参数，提升内容多样性。

4) 小众主题适配差

问题：通用模板不适用于细分领域。

解决方案：新增“场景自适应”逻辑，让 AI 识别主题类型，匹配专属创作模板。

2. 文案类、工具类智能体的通用开发技巧

下面结合本章案例实战，总结适配文案类、工具类智能体的通用开发技巧：

(1) 提示词结构化：将需求分为核心功能、输出格式、风格约束、合规要求四大板块，条理清晰，便于 AI 解析。

(2) 小步迭代，逐步优化：不要追求一步到位，先实现基础功能，再逐一细节、风格、拓展功能进行迭代，降低开发难度。

(3) 多场景测试：开发完成后，选取大众、小众、情感、干货等不同类型测试用例，全面验证兼容性。

(4) 善用历史版本管理：迭代出错时可一键回滚，避免重复开发。

本章开发的小红书笔记生成智能体，不仅可用于个人自媒体创作，还可进行场景延伸：自媒体工作室可批量部署，提升团队内容产出效率；电商商家可结合产品主题，自动生成种草笔记；培训机构可作为教学工具，用于小红书运营教学。

从技术角度而言，扣子编程依托大语言模型降低了 AI 应用的开发门槛，让非技术人员也能将创意转化为可用的智能工具。本章案例实现的“主题生成小红书笔记”模式，还可复刻到抖音文案、朋友圈文案、公众号短文等其他内容场景，仅需微调风格约束与输出格式即可。扣子编程以自然语言为核心交互方式，打破了编程技术壁垒，让普通用户也能快速开发贴合自身需求的 AI 智能体。



4.4 本章小结

本章以小红书爆款笔记生成智能体为实战案例，讲解扣子编程零代码开发流程：先梳理功能、格式、合规需求，通过结构化提示词搭建基础智能体；再分三轮迭代优化标题、正文、封面、风格多样性，拓展 AI 生成封面配图功能；最后介绍版本回滚工具与长期运维方案，总结提示词撰写、分步迭代、多场景测试等通用开发方法。该智能体开发模式可迁移至各类文案生成工具。



4.5 思考与练习

(1) 模仿本章案例，自行编写一套结构化提示词，用于开发“抖音短视频文案生成智能体”，参照本章标题、正文、封面逻辑，适配抖音平台风格与输出规范。

(2) 选取“大学生平价护肤”作为测试主题，梳理一份完整测试记录，包含首轮生成存在的问题、对应的迭代优化指令、优化后预期改善效果。

(3) 假设你开发的笔记智能体出现多次生成同一主题文案高度同质化的问题，请写出一段精准、可落地的自然语言迭代优化指令。

(4) 若小红书平台更新限流违规词汇，你该如何快速完成智能体合规更新？简述操作流程。

(5) 除小红书、抖音文案外，思考本章案例的开发模式还能复刻到哪些内容创作场景，分别需要调整哪些风格、输出格式约束？