

# Skill Engineering 培训系列

## 从 Chat 使用者, 到 Skill 工程实践者

让一次成功的任务, 沉淀为可复用、可验证、可迭代的团队能力

PEOPLE  
PROCESS  
SKILLS  
BETTER TOGETHER

从尝试  
到沉淀  
让 AI 真正创造持续价值

IDEA  
EXECUTION  
VALIDATION  
REUSE



### 核心公式

TURN SUCCESSFUL TASKS INTO LASTING TEAM CAPABILITIES

**Skill Engineering** = Skill Design + Deterministic Execution + Validation + Evaluation + Reuse

技能设计      确定性执行      结果验证      效果评估      复用传播

### 带来的价值

MORE STABLE · MORE RELIABLE · MORE EFFICIENT



适用于 Qoder / QoderWork 场景下的数据处理、 workflow 复用与团队培训

A MORE CAPABLE TEAM  
WITH AI

# 为什么需要 Skill Engineering



让一次成功的任务，沉淀为可复用、可验证、可迭代的团队能力

## 当前现象

**已经开始使用 Qoder**

不少同事已把 AI 用到日常工作中

**数据处理是主要场景**

整理明细、汇总报表、解释结果

**主要仍通过 Chat 完成工作**

每次重新说明需求，再反复调整

**成功经验尚未稳定沉淀**

换一个 Chat，往往又要从头解释



## 下一步目标

**感知 Skill 的作用**

看见流程和规则复用带来的价值

**一次性完成成为可复用能力**

把成功方法留在完整的能力包中

**个人经验成为团队共同方法**

让另一位同事也能按同一口径完成

**形成共同的工程语言**

知道如何设计、执行、验证和改进

把 Chat 中成功的方法沉淀为 Skill，再通过验证与复用完善它。

# 从一次数据处理任务说起



拿到下面这份工单明细，怎样让 AI 把它整理成 10 月 10 日的日报？

输入：7 行虚构工单明细

| 工单 ID | 创建日期       | 状态  | 耗时 / 分钟 |
|-------|------------|-----|---------|
| T001  | 2026-10-10 | 已完成 | 60      |
| T002  | 2026-10-10 | 待处理 | 0       |
| T003  | 2026-10-10 | 已完成 | 45      |
| T004  | 2026-10-10 | 已完成 | 90      |
| T004  | 2026-10-10 | 已完成 | 90      |
| T005  | 2026-10-10 | 待处理 | 0       |
| T006  | 2026-10-11 | 已完成 | 30      |

T004 重复一行；T006 属于 10 月 11 日。

## 交给 AI 的任务

请统计 10 月 10 日创建的工单。相同 ID 完全一致时保留一条，日期、状态或耗时不一致时停下来等我确认。告诉我总数、完成数、待处理数和完成单平均耗时。

## 应该交回的日报

5 单：完成 3 单，待处理 2 单

完成单平均： $(60 + 45 + 90) \div 3 = 65$  分钟

这份明细要变成一份日报：去重、按创建日期筛选，再统计数量与完成单平均耗时。

# Chat、Skill、Skill Engineering



把一次成功的任务，转化成能跨对话复用、执行关键操作、具备验证机制的工作能力。

Skill Engineering = Skill Design + Deterministic Execution + Validation + Evaluation + Reuse

技能设计

确定性执行

结果验证

效果评估

复用传播

Chat 使用

单次完成任务

提出问题，获得答案  
解决当下的任务



Skill 使用

复用已有能力

调用已沉淀的流程和资源  
按约定方法完成任务



Skill Engineering

工程化沉淀与持续优化

分清规则、实现与验证  
换对话、换数据、换人复用

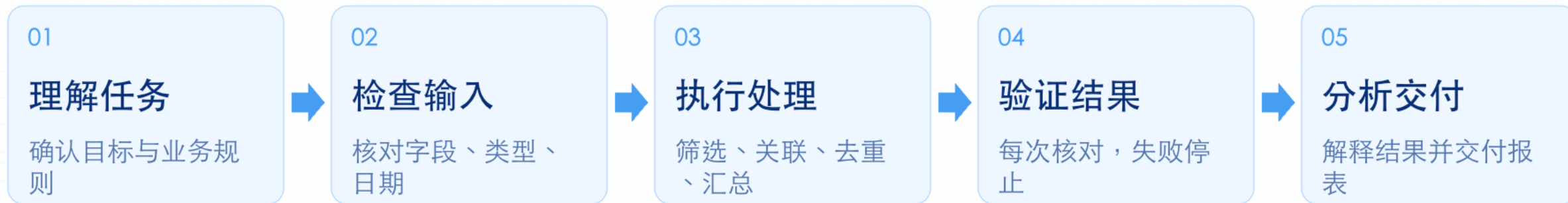
更稳定：新 Chat 也能复现。更可靠：关键计算可验证。更高效：团队可复用、可传播。

# 一个数据处理 Skill 的两个部分



Workflow 安排任务怎样推进，Data Processing 完成数据的解析、转换与计算。

## Workflow：任务怎样推进



### Workflow（工作流）

决定步骤顺序、如何处理歧义、何时继续、何时停止。

### Data Processing（数据处理）

按确认的规则完成运算，并返回结果、异常与校验信息。

Workflow 安排执行与停止；每次运行通过结果校验后，再分析交付。

# AI 推理 vs 确定性执行



在工作流与数据处理两个层次上，分别确定 AI 和固定代码的职责。

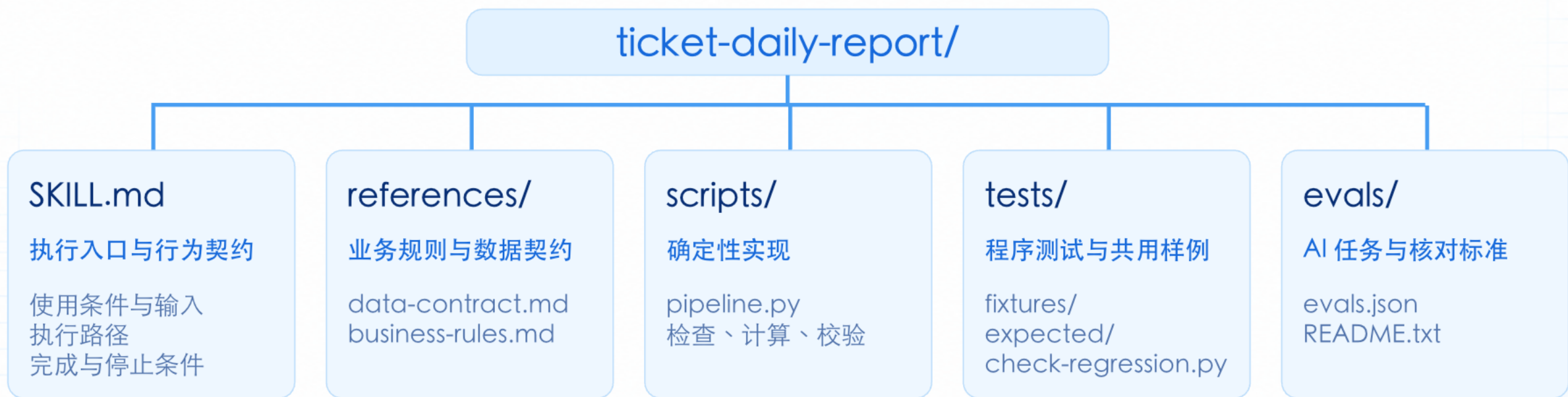
| 工作层次                    | AI 推理负责                    | 确定性实现负责                     |
|-------------------------|----------------------------|-----------------------------|
| Workflow<br>工作流         | 识别任务意图<br>判断异常原因<br>选择处理策略 | 固定步骤顺序<br>状态检查<br>必经校验、停止条件 |
| Data Processing<br>数据处理 | 理解字段含义<br>识别非标准格式<br>解释异常  | 筛选、关联、去重<br>汇总、金额计算<br>格式验证 |

让 Agent 处理不确定性，让代码处理确定性，让验证机制决定结果是否可信。

# 一个好的 Skill 长什么样



参考工单 Skill：入口、规则、实现、测试和评估场景各有用途。



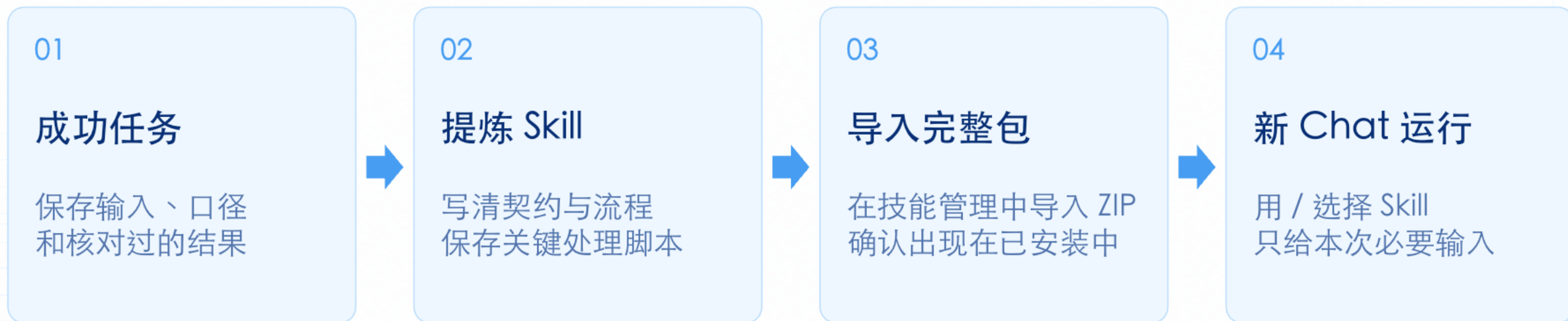
按需增加 assets/ 存放模板；evals.json 保存场景，当前没有自动评估程序。

主文件简洁，细节按需加载（Progressive Disclosure）。简单任务可以只有 SKILL.md。

# Qoder 实操：沉淀一个 Skill



以已经做对的工单日报为起点，把任务方法整理成能独立执行的能力包。



给 Qoder 的提示词：把刚才做对的日报方法整理成完整 Skill ZIP，保存规则和处理脚本，让新 Chat 只给数据、日期和输出位置就能复用。

先手动选择验证执行，再用自然语言验证触发。Qoder 操作入口按当前产品版本核对。

# 如何验证并改进 Skill



Tests 判定任务是否成功，Evals 帮助观察触发、任务表现和真实使用效果。

## Trigger Evals

正确触发，避免误触发

检查真实请求的触发  
属于基础验证



## Task Evals

同一批任务，比较版本

共用 Tests 判定成功  
比较质量、耗时、Token



## Production Evals

真实使用中持续观察

记录失败与人工接管  
跟踪质量、耗时和成本

## Golden Tasks

10 个真实端到端需求

## V1 / V2 (假设)

成功率 60% / 80%

## 相同 Tests 判定

提升 20 个百分点 (假设)

LLM 核对示例：场景 1 + 实际输出 + 执行记录 → 逐项结论与证据（见讲解）。

同一任务、同一 Tests 判定，比较质量与成本。示例需重复运行并控制其他变量。

# 开始实践与问题反馈

把日常工作中的一个成功方法，变成下一次可以继续使用的 Skill。



## 今天就可以开始的一次实践

- 01 选任务并自评  
规则清楚，先确认验收依据
- 02 生成并运行检查  
共用样例，保留实际结果与记录
- 03 换人复用并迭代  
同版本再试，把新问题补进用例

## 问题反馈

遇到问题，联系张坚、李琦

脱敏输入样例  
预期结果与实际结果  
出错截图与校验记录

业务同事沉淀方法与验收标准，技术同事协助固定关键处理，团队共同验证和改进。