

AI工作坊预习作业 - 进阶

从零构建Tibco BW迁移CLI工具的提示词 (<https://github.com/aise-workshop/tibco-movie-practise>)

该练习项目通过多轮提示词练习，指导学员将从零开始构建一个完整的 Tibco BW 到 Spring Boot 的迁移 CLI 工具。这个多轮提示词设计让学员练习使用AI来逐步构建生成一个完整的企业级迁移工具，感受使用AI辅助SDLC的工作特质，并学习到遗留系统迁移的核心理念。

注意该练习已经提供了大量可用的提示词，大家可以使用这些提示词和AI交互，探索尝试借助AI生成代码，辅助完成下面每个阶段的软件开发任务，从阶段1到阶段5，难度不断增加，工程也越来越复杂，学员可根据个人经验，尝试完成下列任务：

- **阶段 1**：项目初始化和基础解析器
- **阶段 2**：BWP 文件解析和业务逻辑转换
- **阶段 3**：CLI 接口和自动化流程
- **阶段 4**：测试验证和问题修复
- **阶段 5**：高级功能和优化

****初级学员****（熟悉 TypeScript/Node.js）：

1. 从阶段 1 开始，重点学习项目架构设计
2. 逐步实现每个解析器和生成器
3. 重点关注代码结构和测试

****中级学员****（有企业级开发经验）：

1. 可以跳过基础架构，直接从阶段 2 开始
2. 重点关注业务逻辑转换的复杂性
3. 深入理解 Tibco BW 和 Spring Boot 的差异

****高级学员****（有迁移工具开发经验）：

1. 可以从阶段 3 开始，重点关注自动化流程
2. 深入研究性能优化和错误处理
3. 扩展工具支持更多的 Tibco BW 特性

AI工作坊环境配置和练习Part1

Git repo link:

Aupro:

<https://github.com/WeiZhang101/agent-backend-demo/tree/main>

Structured Prompts:

<https://github.com/gszhangwei/structured-prompts-driven-development>

该练习基础部分包括:

1. 依据结构化需求生成结构化提示词
2. 依据结构化提示词生成代码

若想100%完成练习并运行, 开发环境满足以下要求:

- **Java:** JDK 21 或更高版本
- **Docker:** 用于运行PostgreSQL数据库

建议根据repo的README去提前调试环境, 按照Quick Start指导去尝试练习

AI工作坊环境配置和练习Part2

Git repo link:

https://github.com/demongodYY/OOCL_langgraph

Instruction:

- 安装 Python 3.12 环境
- 下载仓库代码
- 在仓库代码下启动虚拟环境
- 安装依赖
- 打开 studio practice 文件夹，启动 langgraph 环境，可以对两个 Agent say hello

AI工作坊环境配置和练习Part3

Git repo link

<https://github.com/aise-workshop/jsp2spring-boot-practise>

尝试用提示词利用AI生成该工具。提示词关键信息：

- 自动启动应用（Spring 应用 mvn spring-boot:run）
- 获取出错信息（过滤无关的内容）
- 自动调用 AI Service 修复（SDK: <https://github.com/openai/openai-java>)
 - 分析出错文件
 - 修复出错文件
 - .env => openai_key

不限语言，调用 AI Service 模型相关信息（Java SDK: <https://github.com/openai/openai-java>)