

Context Engineering Intro

📅 2025年8月14日 ⌚ 1 分钟阅读

#Context Engineering

#Template

Context Engineering Intro 是一个全面的模板，用于实现上下文工程，这是通过为 AI 编码助手提供完整上下文来更高效地处理任务的技术。

该项目名为 **Context Engineering Intro**，提供了一个全面的模板，用于实现上下文工程（Context Engineering），这是通过为 AI 编码助手提供完整上下文来更高效地处理任务的技术。相比传统的提示工程（Prompt Engineering），上下文工程可减少失败率、提高一致性、支持复杂功能并具备自我修正能力。

项目功能和结构

快速开始：

克隆模板并可选配置项目规则（编辑 `CLAUDE.md`）。

在 `examples/` 目录添加代码示例。

使用 `INITIAL.md` 描述需求，然后通过指令生成并执行 PRP（产品需求提示）。

核心概念：

提供全面上下文（文档、规则、代码示例等），超越简单的任务提示。

将“PRP”作为实现功能的详细蓝图，定义清晰的实现步骤、验证规则和测试要求。

模板结构：

`.claude/`：定义命令和设置（如 `generate-prp.md` 和 `execute-prp.md`）。

`PRPs/`：存储生成的需求提示和模板。

`examples/`：存放代码模式与实践的关键目录。

目录

文章信息

字数

阅读时间

发布时间

更新时间

标签

#Context Engineering

#Template

`CLAUDE.md`：定义项目级别规则。

`INITIAL.md` 和 `INITIAL_EXAMPLE.md`：需求模板及示例。

使用流程：

定义全局规则 (`CLAUDE.md`)。

描述初始功能需求 (`INITIAL.md`)。

生成 PRP (`/generate-prp`)。

执行 PRP 实现功能 (`/execute-prp`)。

最佳实践：

提供丰富的代码示例和文档。

使用验证机制确保代码成功运行。

自定义符合项目要求的规则 and 标准。

语言使用

主要语言为：Python (51.4%) 和 TypeScript (44.9%)。

许可证

MIT 许可证。

此项目专注于利用 Claude Code 来实施上下文工程，但其方法可适用于任何 AI 编码助手。

分享这篇文章



相关文章推荐

Context Engineering

Context Engineering 是...

从“念咒语”到“造世...

从“念咒语”到“造世界”：提示工...