

IMPRINT · 印记

从草稿到成品

Imprint 是怎么把 Markdown 变成出版社级 PDF 的

目 录

从草稿到成品.....	3
三个数字	4
为什么做这件事.....	5
怎么做到极致.....	6
一个真实的例子.....	7
给 Agent 用	8
边界.....	9

从草稿到成品

“

大多数 AI 生成的 PDF 丑、乱、不专业。Imprint 的目标很简单：让机器排版，让机器验收，让成品可以直接交付。

Imprint 是一个 AI 原生的印刷级 PDF 生成器：Markdown 进，PDF 出，每一份成品都自带 0 - 100 印刷级质检报告。这不是“好看的 PDF 渲染器”——它按出版物的标准工作：宋体正文、黑体标题、首行缩进、标点避头尾、目录页码精确、表格跨页重复表头、代码永不截断。

三个数字

数字	含义
24	套主题，从学术书卷到深夜蓝纸暗色系，全部通过 WCAG 印刷对比度检查
8	套模板：报告 / 书籍 / 论文 / IEEE 双栏 / 红头公文 / 简历 / 技术文档 / 信函
0 – 100	机器给每一份 PDF 打分：页边距、字体、对比度、标点禁则、孤行寡行……逐项给出证据

为什么做这件事

2026 年，任何大模型都能写 Markdown。但"写出来"和"交付一份能直接发给客户、评委、出版社的 PDF"之间，隔着一巨大的鸿沟：

- 默认渲染器丢样式：公式断行、表格溢出、代码被截断
- 没有人验收：排出来的 PDF 是不是符合排版规范，全靠人眼
- 中文字排版的坑没人管：标点避头尾、首行缩进、宋体/黑体的混排层次

Imprint 的答案是把排版规范和验收标准都写进机器。生成之后，机器自动检查 11 项印刷级指标，打分并给出证据；不达标就改，改到达标才交付。

怎么做到极致

每个环节都不是拍脑袋，而是对照社区最佳实践逐项实测：

- 标准：对照 GB/T 9704（公文）、IEEE 模板（论文）、传统书籍版式（正文），逐项落实
- 字体：系统字体自动探测，缺字体时诚实回退并给出警告，绝不静默替换
- 配色：每套主题的正文、小字、装饰对比度分别通过 WCAG $\geq 7:1$ / $\geq 4.5:1$ / $\geq 3:1$
- 图表：mermaid 流程图直接渲染成矢量图，不糊、不溢出
- 无障碍：PDF/UA 结构化标签，机器可读
- 验收：渲染后逐页检查——页数、书签、字体、对比度、禁则、孤行，输出 0 – 100 分

一个真实的例子

把这份报告本身作为输入：

```
B A S H
```

```
npx imprint-pdf 从草稿到成品.md
```

一行命令，自动安装引擎、选主题、渲染、质检。输出结尾就是这份 PDF 的质检报告——分数、每一项的证据、每一页的检查结果，全部可见。

建议

换肤只需换一个 `--theme` 参数，正文和结构完全不变；拿不准主题时用 `modern`。

给 Agent 用

Imprint 内置 MCP server (4 个工具: `list_themes` / `render_markdown` / `validate_pdf` / `new_document`)。Claude Code、Codex、Cursor 只要说"把这份报告排成 PDF", Agent 自己会: 选主题 → 渲染 → 质检 → 达标交付。

B A S H

```
pip install "imprint-pdf[mcp] @ git+https://github.com/263311487-ux/imprint-pdf.git"
imprint-mcp
```

边界

Imprint 只管生成与排版。读取、提取、编辑已有 PDF 不在这条产品线里——把一件事做到极致，比什么都做但都不好更重要。

本文档由 Imprint 自己渲染。打开这份 PDF 的那一刻，它就已经通过了印刷级质检。