

中文文档模板示例

张三

这是一个展示 ILM 模板中文支持的示例文档。本模板基于原版 ILM 模板，增加了对中文排版的支持，包括中文字体、字号定义以及相关的排版工具函数。

2024 年 03 月 19 日

感谢使用这个模板 ♥,
希望您喜欢它 😊

目录

1. 中文支持功能	4
1.1. 中文字体支持	4
1.2. 中文字号系统	4
1.3. 使用示例	4
1.3.1. 不同字体和字号的组合	4
1.3.2. 中文段落缩进	5
1.3.3. 代码块的中文注释支持	5
1.3.4. 中英文混排	5
1.4. 表格中的中文内容	5
1.5. 插图示例	5
1.6. 程序清单示例	5
2. 配置说明	8
2.1. 基本配置	8
2.2. 自定义字体	8
2.3. 字体优先级	8
2.4. 参考文献和引用	8
2.4.1. 引用示例	8
2.4.2. 不同类型的文献	8
参考文献	10
插图索引	11
表格索引	11
程序清单索引	11

1. 中文支持功能

本模板在原有 ILM 模板基础上增加了完整的中文支持，主要包括以下功能：

1.1. 中文字体支持

模板提供了四种主要的中文字体类型：

- 宋体：适用于正文内容，具有衬线特征
- 黑体：适用于标题和强调内容，无衬线字体
- 楷体：适用于特殊场合，书法风格
- 仿宋：适用于引用和特殊文本

每种字体都包含了完整的中英文混排支持，自动选择合适的英文字体进行搭配。

1.2. 中文字号系统

模板定义了完整的中文字号体系，从初号到小七号：

字号名称	大小	字号名称	大小
初号	42pt	小四	12pt
小初	36pt	五号	10.5pt
一号	26pt	小五	9pt
小一	24pt	六号	7.5pt
二号	22pt	小六	6.5pt
小二	18pt	七号	5.5pt
三号	16pt	小七	5pt
小三	15pt		
四号	14pt		

1.3. 使用示例

1.3.1. 不同字体和字号的组合

这是三号宋体文本

这是小二黑体文本

这是四号楷体文本

这是小四仿宋文本

1.3.2. 中文段落缩进

中文段落通常需要首行缩进两个字符。使用 `chinese-indent` 可以实现这一效果：

这是一个带有标准中文缩进的段落。中文排版中，段落的首行通常需要缩进两个汉字的宽度，这样可以使文档看起来更加规范和美观。

另一个段落的示例。通过使用预定义的缩进函数，可以确保所有段落都具有一致的缩进效果。

1.3.3. 代码块的中文注释支持

```
# 这是中文注释
def hello_world():
    print("你好，世界！") # 输出中文问候语

# 支持中英文混合的注释
def calculate_sum(a, b):
    """计算两个数的和 - Calculate the sum of two numbers"""
    return a + b
```

1.3.4. 中英文混排

模板能够很好地处理中英文混排的情况。例如：我们使用 Typst 排版系统来制作这个中文文档，它具有出色的 Unicode 支持和灵活的字体配置功能。

1.4. 表格中的中文内容

姓名	年龄	职业
张三	25	软件工程师
李四	30	产品经理
王五	28	UI 设计师
赵六	32	数据分析师

表 1 中文表格示例

1.5. 插图示例

模板支持插图的自动编号和索引功能。以下是一个简单的示例图形：

输入文档 → 字体处理 → 排版引擎 → 输出 PDF

图 1 中文排版流程示意图

1.6. 程序清单示例

除了内联代码，模板还支持完整的程序清单，并能自动生成索引：

```

# 中文文本处理示例
import re

def process_chinese_text(text):
    """
    处理中文文本的函数
    参数: text - 输入的中文文本
    返回: 处理后的文本
    """

    # 移除多余的空白字符
    text = re.sub(r'\s+', '', text.strip())

    # 标准化中文标点符号
    punctuation_map = {
        ',': '，', # 全角逗号
        '.': '。', # 全角句号
        '?': '？', # 全角问号
        '!': '！', # 全角感叹号
    }

    for old, new in punctuation_map.items():
        text = text.replace(old, new)

    return text

# 使用示例
sample_text = "你好，世界！这是一个中文处理的例子。"
processed = process_chinese_text(sample_text)
print(f"处理结果: {processed}")

```

代码 1 Python 中文处理示例

另一个展示 Typst 代码的程序清单：

```
// 中文文档配置示例
#import "@preview/ilm:1.4.1": *

#set text(lang: "zh", font: 字体.宋体)

#show: ilm.with(
  title: [学术论文标题],
  author: "作者姓名",
  fonts: (
    宋体: ("Source Han Serif SC", "SimSun"),
    黑体: ("Source Han Sans SC", "SimHei"),
  ),
  figure-index: (enabled: true, title: "插图索引"),
  table-index: (enabled: true, title: "表格索引"),
  listing-index: (enabled: true, title: "程序清单索引"),
)

// 使用中文排版工具
#chinese-indent 这是带有标准缩进的中文段落。

#黑体(size: 字号.小二)[重要提示]
#宋体[这里是正文内容。]
```

代码 2 Typst 中文模板配置

2. 配置说明

2.1. 基本配置

要使用中文支持，请在文档开头设置语言：

```
#set text(lang: "zh")
```

2.2. 自定义字体

如果需要使用自定义字体，可以通过 `fonts` 参数传递：

```
#show: ilm.with(  
  fonts: (  
    宋体: ("Custom Song Font", "Source Han Serif SC"),  
    黑体: ("Custom Hei Font", "Source Han Sans SC"),  
  ),  
  // 其他配置...  
)
```

2.3. 字体优先级

模板定义的字体具有完整的回退机制：

1. 首先尝试系统中安装的专业中文字体
2. 如果不可用，则使用开源的思源字体系列[1]
3. 最后回退到系统默认字体

这确保了文档在不同环境下都能正确显示中文内容。

2.4. 参考文献和引用

模板完全支持中文参考文献的管理和引用。在中文学术写作中，正确的引用格式至关重要[2]。

2.4.1. 引用示例

在现代排版系统中，Unicode 标准为中日韩（CJK）文字提供了统一的编码方案[3]。基于这一标准，现代排版工具如 Typst 能够很好地处理中文内容[4]。

2.4.2. 不同类型的文献

模板支持多种类型的中文文献引用：

- 学术期刊文章：[4]
- 会议论文：[3]
- 专业书籍：[2]
- 开源项目和网络资源：[1]

这些引用将在文档末尾的参考文献部分自动生成完整的书目信息。

通过这些功能的集成，ILM 模板现在可以很好地支持中文文档的创作。无论是学术论文、技术文档还是其他类型的中文内容，都可以使用这个模板来获得专业的排版效果。

参考文献

- [1] Adobe Systems, 《思源字体系列：开源中日韩文字体解决方案》. 见于: 2024 年 3 月 19 日. [在线]. 载于: <https://github.com/adobe-fonts/source-han-serif>^o
- [2] 李明, 中文排版技术指南. 北京: 清华大学出版社, 2023.
- [3] 刘建国 和 陈雅丽, 《Unicode CJK 字符集在现代排版系统中的应用》, 收入 第十届中文信息处理学术会议论文集, 上海: 中国计算机学会, 2023, 页 45 – 52.
- [4] 王小红 和 张三, 《基于 Typst 的中文文档排版系统研究》, 计算机科学与技术, 卷 45, 期 2, 页 123 – 135, 2024.

插图索引

图 1 中文排版流程示意图	5
---------------------	---

表格索引

表 1 中文表格示例	5
------------------	---

程序清单索引

代码 1 Python 中文处理示例	6
代码 2 Typst 中文模板配置	7