

# README

sun123zxy

SunQuarTeX

Write once, present everywhere! 基于 Quarto 的多格式输出中英文学术写作模板库

Github 仓库 · 网页 Demo

## 1 核心功能

Quarto 基础功能：

- 基于 Pandoc's Markdown 的完备学术写作语法
- 强大的交叉引用与定理系统功能
- HTML、PDF/LaTeX、Beamer、Github Flavored Markdown (GFM) 全格式输出；MS Word、PPT 有限支持
- 嵌入 Python 代码生成数据图表（Computation）
- Mermaid、Graphviz 流程图绘制（Diagram）

额外支持：

- TikZ / tikz-cd / quiver 图表绘制
- Lean 代码高亮与源码导入转 Markdown
- RST-style list tables
- Github Actions 自动生成 Demo 站点

推荐在网页 Demo 中阅读本 README.

## 2 基础安装

- 下载并安装 quarto-cli. 本仓库渲染使用 Quarto 版本为 1.7.34.
  - (推荐) 安装 VSCode IDE 并安装 Quarto 插件.
- 创建新文章时使用 Github Template 以本仓库为模板建立新仓库. 您也可以下载本仓库的压缩包或 clone 到本地.
- 仓库根目录命令行执行 `quarto render examples/helloworld.qmd --to=html` 测试安装情况.

PDF / Beamer 输出等可选项安装和使用方法参见后文小节 5. 另外, 纯命令行的自动化 CI 流程可参见本仓库下的 Github Actions 配置文件.

## 3 基础使用

### 3.1 渲染

在仓库根目录命令行执行 `quarto render path/to/your_file.qmd --to=your_format`.

- 使用 `--to` 参数指定输出类型, 包括 `html`, `pdf`, `beamer`, `docx`, `gfm`. 如果已经在文档头中 `format` 选项下列明输出格式, 也可不在命令行中指定该选项.

示例文件请在 `examples/` 目录下查看. 其中或包含可选支持内容, 请安装相应依赖或删除对应内容后渲染.

### 3.2 写作

Quarto 使用的底层 Markdown 方言为 Pandoc's Markdown. 速成可直接参考示例文档或 Quarto 官方教程.

文档中开头部分 `---` 之间的内容称为 YAML 文档头 (YAML Front Matter), 用于设置文档相关元信息, 也用于设置输出格式、样式等. 您在自定义的过程中可能需要修改或添加它们.

针对特定输出格式的设置请在文档头 `format` 下对应格式选项下设置. 希望全局生效的设置 (一般) 可在文档头顶层设置.

(重要) 在文件头声明 `lang=zh` 或 `lang=en` 调整语言. 该选项会影响文档的格式和渲染方式.

## 4 更新

如果您文章的仓库由 Github Template 创建, 或者已经在使用 Git 版本控制, 我们推荐使用如下方式拉取源仓库的更新:

```
git remote add sunquartex git@github.com:sun123zxy/sunquartex.git # 添加 sunquartex 作为第二远程仓库
git pull sunquartex master --allow-unrelated-histories --no-commit # 拉取并尝试合并 sunquartex 的更新
# 手动处理合并冲突
git add .
git commit -m "merge updates from sunquartex" # commit 合并
git push # push 到你的远程仓库
```

- `--allow-unrelated-histories` 选项只有在第一次合并时需要添加.
- `--no-commit` 选项用于防止自动 `commit` 合并. 本仓库更新很不稳定, 建议每次合并都手动处理.

## 5 可选项安装与使用

### 5.1 Python

建议您使用 `uv` 管理 Python 版本. `uv sync` 命令会自动安装所需的 Python 依赖. 您也可以参考 `pyproject.toml` 手动安装.

## 5.2 LaTeX / PDF / Beamer 输出

我们没有直接使用 Quarto 默认的 PDF 输出,而是完全重新设计了输出模板(`_assets/suntemp-art.tex`, `_assets/suntemp-pre.tex`). 大动干戈的目的有个人喜好方面的考量: Quarto 默认使用 Koma-Script 系列的 `scrartcl` 文档类,而我们希望在英文环境下保留 `article` 文档类的原汁原味,也希望在中文环境下使用 `ctexart` / `ctexbeamer` 文档类获得更好的排版格式.

重新设计 PDF 模板,您可以参考 Quarto 的模板自定义教程.

### 安装

安装 Quarto 支持的 LaTeX 发行版. 若无,可使用 `quarto install tinytex --update-path` 安装.

### 使用

正常指定 `format` 即可.

- 可在渲染时使用 `--to=latex` 选项输出中间 `.tex` 文件.

## 5.3 Computation 功能

直接嵌入 Python 代码就可以动态生成数据图表. Quarto 文档

### 安装

- 安装适当版本 Python
- 命令行 `pip install .` 安装 `pyproject.toml` 列明的所需模块

### 使用

使用例:

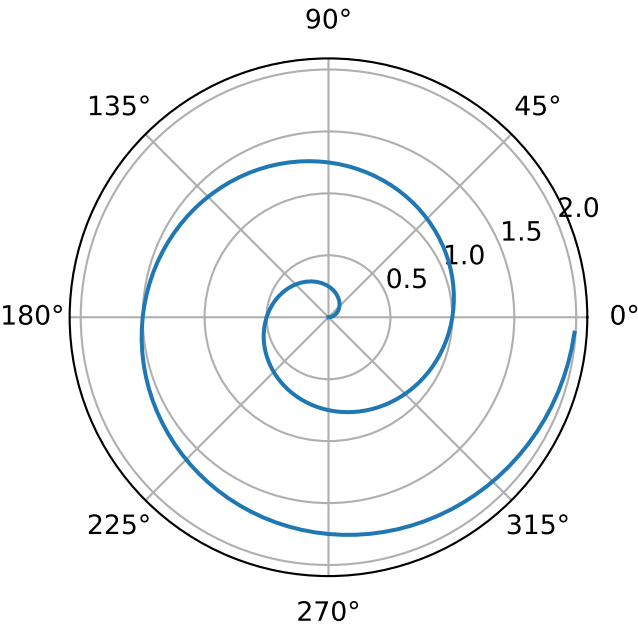


图 1: A line plot on a polar axis

传统 Markdown 表格

$L_i \times C_j$	2	N	$\mathbb{R}$
2	4	N	$\mathbb{R}$
N	N	N	?
$\mathbb{R}$	$\mathbb{R}$	?	$\mathbb{R}$

Markdown grid style tables

a		c	d	e
f	be	ew	a	b
c	d	ewfe		e
f	g	h	r	e

RST-style list tables

我们还支持 RST-style list tables. 该格式可以比较方便地合并单元格.

row 1, column 1	row 1, column 2	row 1, column 3
row 2, column 1	row 2, column 2-3	
row 3, column 1	row 3, column 2	row 3, column 3

亦见 `examples/timetable.qmd`.

Computation based tables

您也可以直接使用代码生成表格：

R ( $\Omega$ )	U (V)	I (mA)	P (mW)	R ( $\Omega$ )	U (V)	I (mA)	P (mW)
0	0.0242	3.6	0.08712	1110	3.75	3.4	12.75
100	0.386	3.6	1.3896	1120	3.77	3.4	12.818
200	0.747	3.6	2.6892	1130	3.78	3.4	12.852
300	1.104	3.6	3.9744	1140	3.8	3.4	12.92
400	1.46	3.6	5.256	1150	3.81	3.3	12.573
500	1.813	3.5	6.3455	1160	3.83	3.3	12.639
600	2.16	3.5	7.56	1170	3.84	3.3	12.672
700	2.51	3.5	8.785	1180	3.85	3.3	12.705
800	2.86	3.5	10.01	1190	3.86	3.3	12.738
900	3.19	3.5	11.165	1200	3.87	3.2	12.384
1000	3.48	3.4	11.832	1210	3.9	3.2	12.48
1100	3.7	3.3	12.21	1220	3.92	3.2	12.544
1200	3.87	3.2	12.384	1230	3.93	3.2	12.576
1300	3.99	3	11.97	1240	3.94	3.2	12.608
1400	4.08	2.9	11.832	1250	3.95	3.2	12.64
1500	4.16	2.7	11.232	1260	3.95	3.2	12.64
2000	4.39	2.2	9.658	1270	3.96	3.1	12.276
4000	4.66	1.13	5.2658	1280	3.97	3.1	12.307
inf	4.85	0.0477	0.231345	1290	3.98	3.1	12.338

(a) (粗)

(b) (细)

表 4: 太阳能电池的负载特性

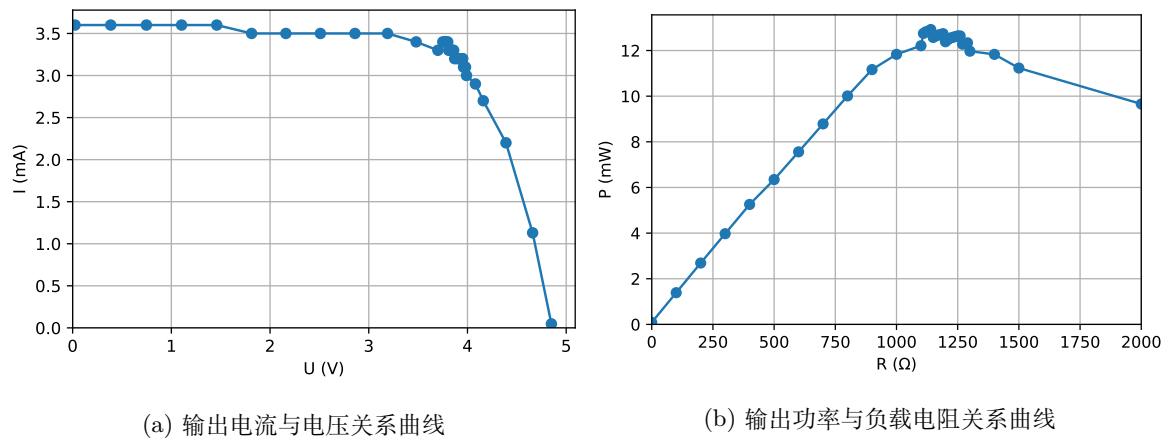


图 2: 太阳能电池的负载特性

交叉引用

在表格外侧包裹 `::: {#tbl-label}` 块。表格 caption 置于块的最后一行。例如：

a		c	d	e
f	be	ew	a	b
c	d	ewfe		e
f	g	h	r	e

表 5: A Table

#### 5.4 Diagram 流程图 (Mermaid、Graphviz 等)

Quarto 文档

##### 安装

非 HTML 格式下需要额外安装 Chrome 或 Chromium.

- 若无, 可使用 `quarto install tool chromium` 安装, 见 Quarto - Diagrams # Chrome Install

#### 5.5 TikZ / TikZ-cd / Quiver 交换图

该功能由 `_assets/tikz.lua` 实现.

##### 安装

如果只是输出到 PDF / Beamer, 除了安装 LaTeX 发行版之外没有别的额外步骤.

如还需输出至其它格式: 请确保 XeLaTeX、dvisvgm、mutool 已在 PATH 中, 且提前安装需要使用的 LaTeX 宏包:

- 安装用于 TikZ 渲染的 LaTeX 宏包:

新建任意空白 Quarto 文档 `temp.qmd` 并执行

```
quarto render temp.qmd --to=pdf --template=_assets/suntemp-tikz.tex
```

随后删除 `temp.qmd`.

如果后续渲染时仍然提示缺少宏包, 请手动安装, 例如手动安装 `standalone` 宏包: 执行 `tlmgr install standalone`

- 使用 Quarto 自带的 TinyTeX 安装 dvisvgm:

- 执行 `tlmgr install dvisvgm` 和 `tlmgr path add` 下载 dvisvgm 并添加至 PATH.

- 安装 mutool:

- (Linux / WSL) 执行 `sudo apt install mupdf-tools`.

- (Windows) 请自行在 MuPDF 官网下载并安装 MuPDF, 并确保 `mutool` 在 PATH 中.

**注记** 关于 mutool 必要性的说明:

As of Ghostscript 10.01.0, this will no longer work due to the introduction of a new PDF interpreter. Therefore, an alternative conversion module based on mutool, a utility which is part of the MuPDF package, has been introduced. It's automatically invoked if Ghostscript can't be used and if a working mutool executable is present in a directory which is part of the system's search path.

来自 dvisvgm manual

## 使用

推荐使用 quiver 在线编辑器生成交换图代码. 使用例:

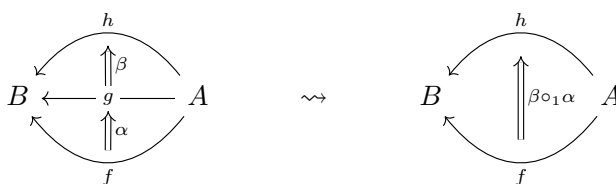


图 3: TikZ-cd / Quiver 示例

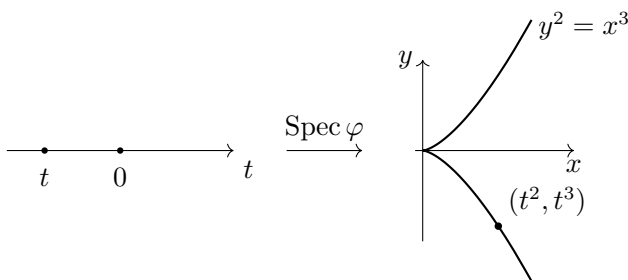


图 4: TikZ 示例

**注记** 在 Beamer 中使用 TikZ 时, 所在幻灯片须添加 `{.fragile}` 标记.

## 5.6 Lean 代码高亮与带注释源码导入

`assets/lean.xml` 用于 Pandoc 的 Lean 代码高亮. 直接使用 `lean` 作为代码块的语言标记即可.

`_assets/lean-include.lua` shortcode 用于直接将带有注释的 Lean 代码导入转换为 Markdown. 使用如下格式即可导入:

```
{{< lean-include path/to/your_file.lean >}}
```

现已支持 PDF / Beamer 输出.

**注记** 目前 HTML 的目录导航定位存在问题.

## 5.7 Github Actions + Github Pages 网站生成

本仓库同时采用 Github Actions + Github Pages 自动生成 Demo 站点.

## 安装

首次使用时，在 Actions 分页中激活 Actions，在本地手动进行第一次网站发布：

- 修改 `_quarto-website.yml` 下 `site-url` 为你的域名（如您使用非 Github Pages 的默认网站域名，请在根目录下额外添加 CNAME 文件）
  - 嫌麻烦的话也可以直接删除这个选项.
- 命令行内设置环境变量 `QUARTO_PROFILE` 为 `website`
- 执行 `quarto publish`
- （清除环境变量）

## 使用

以后的每次 push 都会触发 Github Actions 自动完成的网站生成.

### 5.8 输出为整本书 (Book)

实验性支持 PDF / DOCX 书籍打包. 请调整 `_quarto-book.yml` 的配置并在 `quarto render` 时加入 `--profile=book` 选项渲染.

## 6 样式自定义

修改 YAML 文档头可以自定义部分默认样式.

### 6.1 我要改字号！

目前仅支持 PDF 字号修改. 英文文档默认字号为 11pt，中文文档默认字号为 10.5pt（五号，详见 CTeX 手册）.

```
format:
  pdf:
    fontsize: 12pt
```

### 6.2 我要 / 不想要目录！

```
toc: true # 开启目录
```

该设置全局 / 特定格式下均生效.

### 6.3 我不想给 section 编号 / 我要改 section 编号格式！

```
number-sections: true # section 编号开关，默认关闭
number-depth: 3 # 从 chapter / 一级标题计起的编号深度. 此时 chapter, section, subsection 被编号
toc-depth: 3 # 从 chapter / 一级标题计起的目录深度. 此时 chapter, section, subsection 出现在目录中
```

该设置全局 / 特定格式下均生效.

#### 6.4 我不想给定理编号！ / 我要改定理编号格式！

Quarto 内置的定理编号系统目前无法修改（见 Quarto Discussion #5479），但我们提供自定义 PDF 格式定理编号的可能。（目前仍然无法实现完全关闭 PDF 格式中的定理编号）

```
format:
  pdf:
    custom-theorem:
      numbered-within: section # 开启后将相对于 section（或 subsection, etc.）进行定理编号
      numbered-alike: true # 开启后不同类型的定理将共享编号
```

默认开启 `numbered-alike`，不相对任何标题层级编号。在输出 Book 时，默认相对于 chapter 编号。

注意使用 `numbered-within` 前请先开启 `number-sections`。

#### 6.5 我要改引用格式！

PDF / Beamer 输出使用 BibLaTeX alphabetical，HTML 输出使用 IEEE。如需修改，请自定义 `sun*****.cls` 和 `_format.yml` 和 CSL 文件。

#### 6.6 我要更丰富的 Callout 定理包裹样式！

请移步 [sun123zy/quarto-callout-theorem](https://github.com/sun123zy/quarto-callout-theorem) 学习配置方法。

#### 6.7 我要改 Beamer 幻灯片的颜色！

```
format:
  beamer:
    custom-color:
      define: \definecolor{blueblk}{HTML}{1874D0} # 在这里用 LaTeX 自定义颜色供后面使用
      main: green!40!black # 主色调
      theorem: green!32!black # 各种定理环境颜色
      example: blueblk!50!black # Example / Exercise 环境颜色
      remark: white!15!black # Proof / Solution / Remark 环境颜色
      link: lime!85!black # 链接颜色
```

#### 6.8 PDF / Beamer 宏包不够用，我要自己导入！

```
format:
  html:
    include-in-header:
      text: |
        \(\require{physics}\)
  pdf:
    include-in-header:
```

```
text: |
  \usepackage{physics}
```

亦见 `examples/extra.qmd`. 暂时不支持其它格式下的宏包导入.

## 7 Q&A

### 7.1 一般性的

#### 示例文件编译不了!

示例文件包含了部分可选支持内容, 如未安装相应依赖, 请删除对应内容后渲染.

#### 我不懂 Computer Science, 你能不能讲人话!

请您活用 AI 工具降低学习门槛! 您可以:

- 在网页 Demo 和 AI 聊天提问!
- 使用 VSCode 打开本仓库, 使用自带的 Github Copilot, 将 README 扔进对话框, 提出您的具体需求并获得人话解答.

#### 我想要 XXX 功能! / 我要自己魔改!

仓库主要为自用, 如能为您的生活带来便利欢迎取用. 想要的功能欢迎提 Issue 或 Discussion! (虽然不保证会做:p

我们提供的 YAML 文档头样式只覆盖了极小一部分功能. 更深入的魔改需要您

- 进一步学习底层软件 Quarto, 魔改本仓库的默认配置
- 进一步学习 Pandoc, 编写 LaTeX 模板 / Lua filter

有能力欢迎 Fork 和 Pull Request.

#### 仓库太重, 我想要 standalone 的单文件渲染!

请移步 `quarto-render`, 一个独立开发的小型命令行程序使得单文件 Quarto 渲染更加方便.

### 7.2 写作相关

#### 标题应该用多少个 #?

一级标题被视为整个文档的大标题 (即 Book 中的 chapter title), 文档头标题和一级标题不应同时出现.

二级标题对应 section, 三级标题对应 subsection. Beamer 中四级标题对应 slide. 在 Pandoc 中, 这对应着

```
shift-heading-level-by: -1
slide-level: 4
```

Quarto 对标题层级的处理比较混乱——例如，Book 项目中 Quarto 似乎会额外对标题层级进行 -1 处理；`shift-heading-level-by` 会让小节编号系统无法正常工作。我们做了很多 ad-hoc 的调整（例如 `_assets/promote-h1.lua` 将正文一级标题强制提升至 metadata 标题），因此不建议您再修改这些设置。

参见：

- 相关讨论
- Pandoc `--shift-heading-level-by` 文档
- Pandoc `slide-level` 文档
- Quarto Issue #12048

## 分页符

`{{< pagebreak >}}`. 见官方文档.

## YAML 文档头里的字符串到底打不打引号？

可打可不打。打了的话需要注意特殊字符的转义问题（如 `\`）.

## 用 `$` 包裹行内公式的正确格式

示例：我们有 `$(a + b)^2 = a^2 + 2ab + b^2$`. 证毕.

`$` 内侧应紧接着公式中的非空格字符，外侧与中英文字符之间应有空格，与标点符号、连字符之间不留空格。参考 Pandoc 文档.

## 怎么打出 `TEX`, `LATEX`, `SUNQUARTEX`？

直接写 `\TeX`, `\LaTeX`, `\SunQuarTeX`，不必置于数学环境中。我们编写了 `_assets/fancy-latex-logo.lua` 处理他们在不同格式下的渲染.

## 7.3 输出相关

### 写好的 Beamer 也想输出一份文稿版本的 PDF？

理论上与文档格式兼容，可直接设置 `--to=pdf` 输出文稿版本.

### 我要输出到知乎！

您可以使用 GFM 格式输出，输出内容可复制至 [markdown.com.cn](https://markdown.com.cn) 的在线编辑器转知乎格式.

### PDF 输出，LaTeX 渲染了十遍

您文档的交叉引用可能存在问题。请检查文档头的 `bibliography` 选项和正文中的引用情况.