



# 从零开始使用 L<sup>A</sup>T<sub>E</sub>X 排版论文

上海交通大学图书馆专题培训讲座

李子龙

2022 年 4 月



## 从零开始使用 $\text{\LaTeX}$ 排版论文

<https://github.com/sjtug/sjtulib-latex-talk>

本次讲座帮助学生从零开始跑起来  $\text{\LaTeX}$ ，只用十步了解其基本概念和基础操作。学会使用  $\text{SJTU}_{\text{Thesis}}$  交大学位论文模板，并了解提升排版效率的相关技巧。

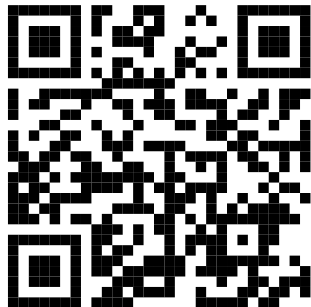
最后更新

2022 年 4 月 3 日

幻灯片下载

Overleaf 

许可证





陈晟祺 等.

如何使用  $\text{\LaTeX}$  排版论文 [EB/OL].

2021. <https://github.com/tuna/thulib-latex-talk>.



曾祥东.

现代  $\text{\LaTeX}$  入门讲座 [EB/OL].

2022. <https://github.com/stone-zeng/latex-talk>.



SJTUG.

SJTUTHESES 用户文档 [EB/OL].

2022. <https://github.com/sjtug/SJTUThesis>.



$\text{\LaTeX}$  Project.  $\text{\CTeX}$  开发小组 译.

[learnlatex.org](https://learnlatex.org)[EB/OL].


2022. <https://github.com/CTeX-org/learnlatex.github.io>.

第一部分


# 学习 L<sup>A</sup>T<sub>E</sub>X

是什么 跑起来 基本结构 扩展 设定格式 逻辑结构 图 表 数学公式 引用

# Learn L<sup>A</sup>T<sub>E</sub>X.org

LearnL<sup>A</sup>T<sub>E</sub>X.org 提供了解 L<sup>A</sup>T<sub>E</sub>X 的 16 篇简短的教程, 并包含了一些可以在线运行的示例, 可以通过亲自动手查看实验效果。本部分主要参考由 C<sub>T</sub>E<sub>X</sub>-org 提供的中文翻译版本 。

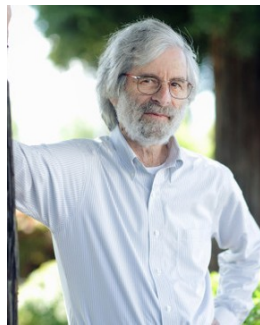


T<sub>E</sub>X 是由斯坦福大学教授高德纳 (Donald E. Knuth) 于 1977 年开始开发的排版引擎。目前仍在更新, 最新版本号为 3.141592653 。



$\text{\LaTeX}$ 

$\text{\LaTeX}$  是最早在 1985 年由现就职于微软的 Leslie Lamport 开发的一种  $\text{\TeX}$  格式<sup>1</sup>，使用一些列宏和扩展宏包来简化  $\text{\TeX}$  的使用。现在由  $\text{\LaTeX}$  Project 的成员维护。现在广泛使用的版本是  $\text{\LaTeX} 2_{\epsilon}$ ，最新的版本为  $\text{\LaTeX} 3$  (2020 年 10 月后默认内置)。



<sup>1</sup>ConTeXt 也是一种  $\text{\TeX}$  格式 .

# pdfL<sup>A</sup>T<sub>E</sub>X

pdfL<sup>A</sup>T<sub>E</sub>X 是为了编译一个 L<sup>A</sup>T<sub>E</sub>X 文档而运行的程序。实际上底层在运行一个叫 pdfT<sub>E</sub>X 的引擎，并预装了对应的 L<sup>A</sup>T<sub>E</sub>X 格式。为了利用临时文件，可能就需要多次运行程序。

```
> pdflatex main.tex
This is pdfTeX,
Version 3.141592653-
2.6-1.40.23 (MiKTeX
21.10)
entering extended mode
LaTeX2e <2021-11-15>
L3 programming layer
<2021-11-22>
```





pdfT<sub>E</sub>X 是编译 T<sub>E</sub>X 文档（以 .tex 结尾）  
的引擎——可以理解 T<sub>E</sub>X 指令的程序。

```
> pdflatex main.tex  
This is pdfTeX,  
Version 3.141592653-  
2.6-1.40.23 (MiKTeX  
21.10) entering  
extended mode  
LaTeX2e <2021-11-15>  
L3 programming layer  
<2021-11-22>
```

表: 主流  $\LaTeX$  程序<sup>23</sup>







引擎	pdf $\TeX$	X $\TeX$	Lua $\TeX$
程序	pdf $\LaTeX$	X $\LaTeX$	Lua $\LaTeX$
特点	直接生成 PDF, 支持 micro-typography	支持 Unicode、OpenType 与复杂文字编排 (CTL)	支持 Unicode, 内联 Lua, 支持 OpenType


pdf $\LaTeX$  不支持 Unicode。为了排版中文, 大部分情况下   应当使用 X $\LaTeX$ , 而 Lua $\LaTeX$  速度相对较慢。 可以在一些情况下使用 pdf $\LaTeX$ 。

<sup>2</sup>(u)p $\TeX$  是日语最常用的引擎, 生成 .dvi, 支持 Unicode。

<sup>3</sup>Ap $\TeX$   具有底层 CJK 支持, 内联 Ruby, Color Emoji。

表: T<sub>E</sub>X 发行版

发行版	MiK <sub>T</sub> E <sub>X</sub> 	T <sub>E</sub> X Live 	MacT <sub>E</sub> X 
特点	只安装必要文件, 依赖 用时更新	所有平台均可使用, 每 年发布一次	Mac 系统专用, 对 T <sub>E</sub> X Live 的进一步打包
推荐平台			

要让 L<sup>A</sup>T<sub>E</sub>X 跑起来, 核心就是要有一套 T<sub>E</sub>X 发行版, 来获取让 L<sup>A</sup>T<sub>E</sub>X 工作所需的一组程序和文件。参考《一份简短的关于 L<sup>A</sup>T<sub>E</sub>X 安装的介绍》 安装想使用的发行版。推荐使用发行版的最新版本<sup>4</sup>, 并使用国内镜像。

<sup>4</sup>老版本 Linux 系统的包管理器自带 T<sub>E</sub>X Live 发行版可能不是最新的 , 尽量使用镜像安装, 并手动将相关环境变量添加到路径 .



**稳定、快速、现代的镜像服务。**  
托管于华东教育网骨干节点上海交通大学。

推荐使用 SJTUG 软件镜像服务 , 离得近, 下得快。

**TeX Live** tlmgr option repository

<https://mirrors.sjtug.sjtu.edu.cn/CTAN/systems/texlive/tlnet>

**MiKTeX** 在 MiKTeX Console 中设置镜像源为

<https://mirrors.sjtug.sjtu.edu.cn>

 可以在 SJTUG 镜像站通知频道  获得更多信息, 加入关联群组参与讨论。



表: 开源编辑器推荐


编辑器	Visual Studio Code L <sup>A</sup> T <sub>E</sub> X Workshop	T <sub>E</sub> Xstudio	T <sub>E</sub> Xworks
特点	搭配 VS Code 使用非常方便, 易扩展	可以使用大量的菜单选项输入代码块, 用户友好	只提供基础的高亮与运行方法, 发行版自带 <sup>5</sup>

使用专为 L<sup>A</sup>T<sub>E</sub>X 设计的编辑器将带来更多便利, 因为它们往往会提供一键编译、内置 PDF 阅读器以及语法高亮等功能。几乎所有现代的 L<sup>A</sup>T<sub>E</sub>X 编辑器都提供 SyncT<sub>E</sub>X 这一强大的功能, 以在 PDF 和代码间对应跳转。

<sup>5</sup>MacT<sub>E</sub>X 打包的是 T<sub>E</sub>XShop 编辑器。

表: 在线协作平台推荐

在线平台	Overleaf 	交大 L <sup>A</sup> T <sub>E</sub> X 助手 
特点	最流行的在线平台, 提供大量的模板, 但国内访问慢	校内平台, 隐私保护有保障, 共享项目限制少

在线平台允许你直接在网页中编辑文档, 无需本地安装即可在后台运行 L<sup>A</sup>T<sub>E</sub>X, 并显示生成的 PDF。可以参照 Overleaf 官方文档学习如何使用在线平台 。

`\documentclass` 命令加载了文档类。 `article` 是由  $\text{\LaTeX}$  提供的用于排版短文档的基本文档类。

`article` 不包含章的短文档

`report` 含有章的单面印刷文档

`book` 含有章的双面印刷文档

`beamer` 幻灯片

### 排版英文最简示例

```
1 \documentclass{article}
2 \begin{document}
3   Together for a Shared Future
4 \end{document}
```

`document` 环境用于指示文档主体的范围。L<sup>A</sup>T<sub>E</sub>X 还有其他的使用 `\begin` 和 `\end` 的搭配，我们称这些为环境。它们将用来设定局部格式命令<sup>6</sup>。

### 排版英文最简示例

```
1 \documentclass{article}
2 \begin{document}
3   Together for a Shared Future
4 \end{document}
```

<sup>6</sup>环境实际上是一个组，只不过通过语义化的形式预装了对应的格式命令。普通的组可以直接使用一对大括号之间的内容 `{...}` 表示。



Together for a Shared Future

### 排版英文最简示例

```
1 \documentclass{article}
2 \begin{document}
3   Together for a Shared Future
4 \end{document}
```

`\usepackage` 用于使用宏包以向  $\text{\LaTeX}$  添加或修改功能，需要在导言区调用。这里使用 `ctex` 宏集以获得中文支持。其调用底层因不同的引擎而不同。

引擎	$\text{pdf\TeX}$	$\text{X\TeX}$	$\text{Lua\TeX}$
程序	$\text{pdf\LaTeX}$	$\text{X\LaTeX}$	$\text{Lua\LaTeX}$
宏包	$\text{CJK}^6$	$\text{xeCJK}$	$\text{luatexja}$
封装		$\text{ctex}$	

### 排版中文

```
1 \documentclass{article}
2 \usepackage{ctex} % 导言区
3 \begin{document}
4     一起向未来
5
6     Together for a Shared Future
7 \end{document}
```

<sup>6</sup> $\text{ctex}$  在  上已经不可以使用  $\text{pdf\LaTeX}$  编译，以及在  上使用该引擎也会变更自动间距调整等功能的默认行为。

CT<sub>E</sub>X 建议对于之前提到的常规文档类，最佳实践是使用该宏集提供的四种中文文档类，以对特定类型提供额外的中文排版适配。

ctexart	ctexrep
ctexbook	ctexbeamer

### 排版中文最佳实践

```
1 \documentclass{ctexart}
2
3 \begin{document}
4     一起向未来
5
6     Together for a Shared Future
7 \end{document}
```

一起向未来  
Together for a Shared Future

1

### 排版中文最佳实践

```
1 \documentclass{ctexart}
2
3 \begin{document}
4     一起向未来
5
6     Together for a Shared Future
7 \end{document}
```

大部分情况下，你都不应当在  $\text{\LaTeX}$  中强制断行：你几乎只是想另起一段，或者是想在段落之间添加空行（使用 `parskip` 宏包就可实现）。只有很少的情况下你需要使用 `\\` 来另起一行而不另起一段（强制断行仍在同一段）。

### 排版中文最佳实践

```
1 \documentclass{ctexart}
2
3 \begin{document}
4   一起向未来
5
6   Together for a Shared Future
7 \end{document}
```

一起向未来  
*Together for a Shared Future*

1

### 样式

```
1 \documentclass{ctexart}
2 \begin{document}
3   \textbf{一起向未来}
4
5   \emph{Together for a Shared Future}
6 \end{document}
```

可以使用显式样式设定命令对片段做加粗、斜体、等宽等等的处理。

<code>\textrm</code>	衬线
<code>\textbf</code>	<b>加粗</b>
<code>\textit</code>	斜体
<code>\texttt</code>	等宽
<code>\textsf</code>	无衬线
<code>\textsc</code>	SMALL CAPS
<code>\textsl</code>	<i>Slanted</i>

### 样式

```

1 \documentclass{ctexart}
2 \begin{document}
3   \textbf{一起向未来}
4
5   \emph{Together for a Shared Future}
6 \end{document}
    
```

与 Word 不同的是， $\text{\LaTeX}$  一般情况下并不需要使用上面的显式命令，而是采用逻辑标记的方法，比如 `\emph` 可以强调文字，以及下面将要提到的目次命令（第 21 页）。这样可以统一管理格式。

## 样式

```
1 \documentclass{ctexart}
2 \begin{document}
3   \textbf{一起向未来}
4
5   \emph{Together for a Shared Future}
6 \end{document}
```



1

一起向未来  
*Together for a Shared Future*

### 大小

```
1 \documentclass{ctexart}
2 \begin{document}
3   {\bfseries\Large 一起向未来\par}
4   {\itshape Together for a Shared Future}
5 \end{document}
```

同样地，你也可以显式地设定字体大小，但是这种命令会更改行文设置，所以需要使用一个组来限定作用范围<sup>6</sup>。

<code>\tiny</code>	极小
<code>\scriptsize</code>	角标大小
<code>\footnotesize</code>	脚注大小
<code>\small</code>	小
<code>\normalsize</code>	正常大小
<code>\large</code>	大
<code>\huge</code>	巨大

## 大小

```

1 \documentclass{ctexart}
2 \begin{document}
3   {\bfseries\Large 一起向未来\par}
4   {\itshape Together for a Shared Future}
5 \end{document}
    
```

<sup>6</sup>注意最后显式地使用 `\par` 在改回大小前结束该段，否则会导致下一行的行间距异常！

也可以使用字体样式对应的更改字体设置的命令，这对于大段文段的设置而言也是很方便的。

```
\textrm \rmfamily
\texttt \ttfamily
\textsf \sffamily
\textbf \bfseries
\textit \itshape
\textsc \scshape
\textsl \slshape
```

## 大小

```
1 \documentclass{ctexart}
2 \begin{document}
3   {\bfseries\Large 一起向未来\par}
4   {\itshape Together for a Shared Future}
5 \end{document}
```

## 无序列表

```
1 \documentclass{ctexart}
2 \begin{document}
3   \begin{itemize}
4     \item 第一项
5     \item 第二项
6     \item 第三项
7   \end{itemize}
8 \end{document}
```

## 有序列表

```
1 \documentclass{ctexart}
2 \begin{document}
3   \begin{enumerate}
4     \item 第一项
5     \item 第二项
6     \item 第三项
7   \end{enumerate}
8 \end{document}
```

## 描述列表

```
1 \documentclass{ctexart}
2 \begin{document}
3   \begin{description}
4     \item[第一] 文本
5     \item[第二] 文本
6     \item[第三] 文本
7   \end{description}
8 \end{document}
```

- 1

- 1

1

$\text{\LaTeX}$  可以使用目次命令将文档划分层级<sup>6</sup>，并自动设定对应字体样式和大小。

命令	含义	层次
<code>\chapter</code>	章 <sup>7</sup>	0
<code>\section</code>	节	1
<code>\subsection</code>	小节	2
<code>\subsubsection</code>	小小节	3

## 目次

- 1 `\documentclass{ctexart}`
- 2 `\begin{document}`
- 3 `\section{概念}`
- 4 `\subsection{\LaTeX{}}`
- 5 `\LaTeX{}` 是一个用以排版高质量作品的文档准备系统。
- 6 `\end{document}`

<sup>7</sup>章这一级只在 `report` 和 `book` 文档类（包括对应的中文文档类）有定义。还有不常用的 `\part` (`0@article/-1@report&book&beamer`) 以及更低层次的 `\paragraph` (4) 与 `\subparagraph` (5)。

`\tableofcontents` 用来生成对于目次命令的目录。如果你想设定显示到哪个层级，在这个命令前使用

`\setcounter{tocdepth}{层次}`  
如果你想在目录中使用更短的标题：

`\section[短标题]{长标题}`  
如果你想让本目次的标题不显示在目录中：

`\section*{目录没这个标题}`

### 主文档

```
1 \documentclass{ctexrep}
2 \includeonly{learnlatex, sjtuthesis}
3 \begin{document}
4   \tableofcontents
5   \include{learnlatex}
6   \include{sjtuthesis}
7 \end{document}
```

对于大型文档而言，使用多个文件管理源文件通常是更方便的。而 `\include` 和 `\input` 都以相对路径的方式插入其他  $\text{T}_{\text{E}}\text{X}$  文档。区别在于，`\include` 命令会从新页开始并做一些内部调整，这基本上只对 chapter 这一级有用。而 `\input` 会原样插入源代码。

## 主文档

```
1 \documentclass{ctexrep}
2 \includeonly{learnlatex, sjtuthesis}
3 \begin{document}
4   \tableofcontents
5   \include{learnlatex}
6   \include{sjtuthesis}
7 \end{document}
```



但是 `\include` 插入的文档可以使用 `\includeonly` 管理当前要排印哪一部分的内容，利用所有章节辅助文件的同时，减少编译时间并专注于该部分的内容。

### 主文档

```
1 \documentclass{ctexrep}
2 \includeonly{learnlatex, sjtuthesis}
3 \begin{document}
4   \tableofcontents
5   \include{learnlatex}
6   \include{sjtuthesis}
7 \end{document}
```

## learnlatex.tex

```
1 \chapter{学习 \LaTeX{}}
2 \section{概念}
3 \subsection{\LaTeX{}}
4 \LaTeX{} 是一个用以排版高质量作品的文档准备系统。
```

子文件中就不需要添加 **document** 环境了<sup>8</sup>。

## 主文档 main.tex

```
1 \documentclass{ctexrep}
2 \includeonly{learnlatex,sjtuthesis}
3 \begin{document}
4   \tableofcontents
5   \include{learnlatex}
6   \include{sjtuthesis}
7 \end{document}
```

<sup>8</sup>如果想强制指定子文档的主文档，可以在文件第一行输入魔术命令：`% !TeX root = main.tex`

## 插入单图

```
1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \graphicspath{{figs/}{pics/}}
4 \begin{document}
5 \begin{figure}[ht]
6   \centering
7   \includegraphics[width=4cm]{sjtug}
8   \caption{SJTUG 徽标}\label{fig:sjtug}
9 \end{figure}
10 \end{document}
```



## 插入单图

```
1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \graphicspath{{figs/}{pics/}}
4 \begin{document}
5 \begin{figure}[ht]
6   \centering
7   \includegraphics[width=4cm]{sjtug}
8   \caption{SJTUG 徽标}\label{fig:sjtug}
9 \end{figure}
10 \end{document}
```

为了插入外部图片，需要使用 `graphicx` 宏包。之后在文档主体便可以使用 `\includegraphics` 插入图片。导言区也可以加入 `\graphicspath` 指定图片文件夹<sup>9</sup>。

<sup>9</sup>其命令参数每个为一个以 `/` 结尾的文件夹，每个文件夹需要使用大括号包裹起来。

## 插入单图

```
1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \graphicspath{{figs/}{pics/}}
4 \begin{document}
5 \begin{figure}[ht]
6   \centering
7   \includegraphics[width=4cm]{sjtug}
8   \caption{SJTUG 徽标}\label{fig:sjtug}
9 \end{figure}
10 \end{document}
```

`\includegraphics` 命令便以相对路径的方式插入图片，如果无同名图片，那么后缀名可以省略。可以使用可选参数指定插入的图片尺寸，最佳实践是使用 `\textwidth` 或 `\linewidth` 的相对值指定尺寸大小，以在未来可能的布局更改中保留一定的灵活性。

## 插入单图最佳实践

```
1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \graphicspath{{figs/}{pics/}}
4 \begin{document}
5 \begin{figure}[ht]
6   \centering
7   \includegraphics[width=0.4\textwidth]{
8     sjtug}
9   \caption{SJTUG 徽标}\label{fig:sjtug}
10 \end{figure}
11 \end{document}
```

也可以通过键值对的方法设置图片的其他属性。

width	宽度
height	高度
scale	缩放
angle	角度

```
1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \graphicspath{{figs/}{pics/}}
4 \begin{document}
5 \begin{figure}[ht]
6     \centering
7     \includegraphics[width=0.4\textwidth]{
8         sjtug}
9     \caption{SJTUG 徽标}\label{fig:sjtug}
10 \end{figure}
11 \end{document}
```

**figure** 为一个浮动体环境 (**table** 也是)，可以将其移动到  
其他位置上以减少行文中的空白。  
可以添加可选参数以指定如何放  
置浮动体，最多可以使用四种位  
置描述符：

h	Here	尽可能在这里
t	Top	页面顶部
b	Bottom	页面底部
p	Page	浮动体专页

还可以只使用 `float` 宏包提供的 `H` 描述符以强制置于此处。

## 插入单图最佳实践

```
1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \graphicspath{{figs/}{pics/}}
4 \begin{document}
5 \begin{figure}[ht]
6   \centering
7   \includegraphics[width=0.4\textwidth]{
8     sjtug}
9   \caption{SJTUG 徽标}\label{fig:sjtug}
10 \end{figure}
11 \end{document}
```

采用 `\centering` 命令而不是 `center` 环境来水平居中图片。这将避免多余的纵向间距。



## 插入单图最佳实践

```
1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \graphicspath{{figs/}{pics/}}
4 \begin{document}
5 \begin{figure}[ht]
6   \centering
7   \includegraphics[width=0.4\textwidth]{
8     sjtug}
9   \caption{SJTUG 徽标}\label{fig:sjtug}
10 \end{figure}
11 \end{document}
```

使用 `\caption` 命令输入题注，如果这个命令写在插入图片前面，题注将会在上方（中文中一般对 `table` 环境这么做）。后面将会看到如何对留有标记（`\label`）的图片进行引用。

## 插入双图

```
1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \graphicspath{{figs/}{pics/}}
4 \begin{document}
5   \begin{figure}[ht]
6     \begin{minipage}{0.48\textwidth}
7       \centering
8       \includegraphics[height=2cm]{sjtug}
9       \caption{SJTUG 徽标}\label{fig:sjtug}
10    \end{minipage}\hfill
11    \begin{minipage}{0.48\textwidth}
12      \centering
13      \includegraphics[height=2cm]{sjtugt}
14      \caption{SJTUG文字}\label{fig:sjtugt}
15    \end{minipage}
16  \end{figure}
17 \end{document}
```

在 `figure` 环境里使用 `minipage` 小页制作列盒子，以并排两图并分别编号，需要设定强制参数以指定列宽。两个小页之间添加 `\hfill` 使两个小页两端对齐。

## 插入双图

```
1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \graphicspath{{figs/}{pics/}}
4 \begin{document}
5   \begin{figure}[ht]
6     \begin{minipage}{0.48\textwidth}
7       \centering
8       \includegraphics[height=2cm]{sjtug}
9       \caption{SJTUG 徽标}\label{fig:sjtug}
10    \end{minipage}\hfill
11    \begin{minipage}{0.48\textwidth}
12      \centering
13      \includegraphics[height=2cm]{sjtugt}
14      \caption{SJTUG文字}\label{fig:sjtugt}
15    \end{minipage}
16  \end{figure}
17 \end{document}
```

在每个小页内部分别使用  
`\caption` 添加标签。

## 插入双图

```
1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \graphicspath{{figs/}{pics/}}
4 \begin{document}
5   \begin{figure}[ht]
6     \begin{minipage}{0.48\textwidth}
7       \centering
8       \includegraphics[height=2cm]{sjtug}
9       \caption{SJTUG 徽标}\label{fig:sjtug}
10    \end{minipage}\hfill
11    \begin{minipage}{0.48\textwidth}
12      \centering
13      \includegraphics[height=2cm]{sjtugt}
14      \caption{SJTUG文字}\label{fig:sjtugt}
15    \end{minipage}
16  \end{figure}
17 \end{document}
```



图 1: SJTUG 徽标

**SJTUG**

图 2: SJTUG 文字

```

1 \documentclass{ctexart}
2 \usepackage{graphicx}
3 \usepackage{subcaption}
4 \graphicspath{{figs/}{pics/}}
5 \begin{document}
6   \begin{figure}[ht]
7     \begin{subfigure}{0.48\textwidth}
8       \centering
9       \includegraphics[height=2cm]{sjtug}
10      \caption{徽标}
11    \end{subfigure}\hfill
12    \begin{subfigure}{0.48\textwidth}
13      \centering
14      \includegraphics[height=2cm]{sjtugt}
15      \caption{文字}
16    \end{subfigure}
17    \caption{SJTUG}\label{fig:sjtug}
18  \end{figure}
19 \end{document}

```



subcaption 宏包提供了 **subfigure** 环境（以及 **subtable**），可以用于以子图的形式插入，编号会增加一级。也可以为子图添加子级引用编号。

```

1 \documentclass{ctexart}
2 \usepackage{array,booktabs}
3 \begin{document}
4 \begin{table}[ht]
5   \centering
6   \caption{北京冬奥会吉祥物}
7   \begin{tabular}{lp{3cm}}
8     \toprule
9     吉祥物 & 描述 & \\
10    \midrule
11    冰墩墩 & 2022 年北京冬季奥运会吉祥物 & \\
12    雪容融 & 2022 年北京冬季残奥会吉祥物 & \\
13    \bottomrule
14  \end{tabular}
15 \end{table}
16 \end{document}

```

使用 **tabular** 环境绘制表格。需要添加参数（称为**表格导言**）以确定每一列的对齐方式。引入 **array** 宏包来使用更多的引导符。

<b>l</b>	向左对齐
<b>c</b>	居中对齐
<b>r</b>	向右对齐
<b>p{3cm}</b>	固定列宽，两端对齐
<b>m{3cm}</b>	p + 垂直居中对齐
<b>&gt;{\bfseries}</b>	后一列单元格前加命令
<b>*{3}{1}</b>	三个左对齐列

```
1 \documentclass{ctexart}
2 \usepackage{array, booktabs}
3 \begin{document}
4 \begin{table}[ht]
5   \centering
6   \caption{北京冬奥会吉祥物}
7   \begin{tabular}{lp{3cm}}
8     \toprule
9     吉祥物 & 描述 & \\
10    \midrule
11    冰墩墩 & 2022 年北京冬季奥运会吉祥物 & \\
12    雪容融 & 2022 年北京冬季残奥会吉祥物 & \\
13    \bottomrule
14  \end{tabular}
15 \end{table}
16 \end{document}
```

booktabs 宏包提供了标准三线表格所需要的行分割线：`\toprule`，`\midrule`，`\bottomrule`。也可以使用 `\cmidrule{1-2}` 来部分地绘制行分割线。一般不推荐在专业表格中使用纵向分割线。

```
1 \documentclass{ctexart}
2 \usepackage{array,booktabs}
3 \begin{document}
4 \begin{table}[ht]
5   \centering
6   \caption{北京冬奥会吉祥物}
7   \begin{tabular}{lp{3cm}}
8     \toprule
9     吉祥物 & 描述 & \\
10    \midrule
11    冰墩墩 & 2022 年北京冬季奥运会吉祥物 & \\
12    雪容融 & 2022 年北京冬季残奥会吉祥物 & \\
13    \bottomrule
14  \end{tabular}
15 \end{table}
16 \end{document}
```

每行内容使用 `\\` 分隔开，每行中的单元格使用 `&` 分隔开。



```
1 \documentclass{ctexart}
2 \usepackage{array,booktabs}
3 \begin{document}
4 \begin{table}[ht]
5   \centering
6   \caption{北京冬奥会吉祥物}
7   \begin{tabular}{lp{3cm}}
8     \toprule
9     吉祥物 & 描述 \\
10    \midrule
11    冰墩墩 & 2022 年北京冬季奥运会吉祥物 \\
12    雪容融 & 2022 年北京冬季残奥会吉祥物 \\
13    \bottomrule
14  \end{tabular}
15 \end{table}
16 \end{document}
```

1


表 1: 北京冬奥会吉祥物	
吉祥物	描述
冰墩墩	2022 年北京冬季奥运会吉祥物
雪容融	2022 年北京冬季残奥会吉祥物

## 表头居中

```
1 \documentclass{ctexart}
2 \usepackage{array,booktabs}
3 \begin{document}
4 \begin{table}[ht]
5   \centering
6   \caption{北京冬奥会吉祥物}
7   \begin{tabular}{lp{3cm}}
8     \toprule
9     \multicolumn{1}{c}{吉祥物} &
10    \multicolumn{1}{c}{描述} \\
11    \midrule
12    冰墩墩 & 2022 年北京冬季奥运会吉祥物 \\
13    雪容融 & 2022 年北京冬季残奥会吉祥物 \\
14    \bottomrule
15  \end{tabular}
16 \end{table}
17 \end{document}
```


`\multicolumn` 命令不仅可以用于合并同行的单元格，还可以用于临时地屏蔽表格导言对该列的对齐方式定义。这里用于居中表头。

`\multicolumn{格数}{对齐方式}{内容}`

跨页表格考虑使用 `longtable` 宏包。带标注的表格可以考虑使用 `threeparttable` 宏包。考虑使用在线工具生成表格代码 .

输入数学公式是  $\text{\LaTeX}$  的绝对强项，很多常见的公式服务依赖于一些轻量级渲染引擎比如 MathJax,  $\text{\KATEX}$ 。但是它们实际上使用的是  $\text{\LaTeX}$  语法变种，也就是说并没有使用  $\text{\LaTeX}$  后端。所以不要期望有完全一致的输出。

为了更好的获得数学公式输入支持，请使用  $\mathcal{AMS}$ math 宏包。数学模式分为两种：

**行内模式** 一般通过一对美元符号 ( $\$ \cdots \$$ ) 标记，可以使用编辑器内建的符号表输入数学符号，也可以使用在线工具手写识别 .

**行间模式** 一般通过 `equation` 环境<sup>9</sup>输入。如果需要使用多行公式，请使用 `align` 环境，并按照类似表格输入的方式，使用 `&` 对齐符号，`\\` 换行。如果不想手动居中，可以考虑多行自动居中的 `gather` 和单个大型公式首尾两端对齐 `multline`。

<sup>9</sup>这是有编号环境，其加星号的变种 `equation*` 用于生成无编号环境。

$$PV = nRT$$

$$\sum_{i=1}^k i^2 = \frac{n(n+1)(2n+1)}{6}$$

$$T(n) = aT\left(\left\lceil \frac{n}{b} \right\rceil\right) + \mathcal{O}(n^d)$$

$$\frac{x_1+x_2+x_3}{3} \geq \sqrt[3]{x_1x_2x_3}$$

$$n = (\underbrace{1 \cdots 1}_{k \text{ of } 1\text{'s}})_2 = 2^{k+1} - 1$$

$$\nabla f(P) = \left( \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right) \Big|_P$$

$$\int_a^b f(x) \, dx = \lim_{|P| \rightarrow 0} \sum_{i=1}^n f(\xi_i) \Delta x_i$$

$$T(n) = \begin{cases} \mathcal{O}(n^d), & \text{if } d > \log_b a, \\ \mathcal{O}(n^d \log n), & \text{if } d = \log_b a, \\ \mathcal{O}(n^{\log_b a}), & \text{if } d < \log_b a. \end{cases} \quad (1)$$

$$Q^T A = R \quad (2)$$

$$\begin{pmatrix} q_1^T \\ q_2^T \\ q_3^T \end{pmatrix} (a_1 \quad a_2 \quad a_3) = R \quad (3)$$

正如之前所提到的， $\text{\LaTeX}$  中使用 `\label` 标记，然后可以使用 `\ref` 来引用这个标记。`\label` 可以放在使用计数器的对象之后。

## 图

```
1 \begin{figure}
2   \caption{示例}\label{fig:example}
3 \end{figure}
```

## 表

```
1 \begin{table}
2   \caption{示例}\label{tab:example}
3 \end{table}
```

## 目次

```
1 \section{示例}\label{sec:example}
```

## 公式

```
1 \begin{equation}
2   a = b + c
3 \label{eq:example}
4 \end{equation}
5 如公式 \eqref{eq:example} 所示，
```

为了使得对公式编号的引用带有括号，推荐使用  $\mathcal{M}$ Smath 宏包中的 `\eqref` 命令。对于多行公式环境，每一个换行符前都可以添加一个 `\label` 用于引用该行公式。

## 图

```
1 \begin{figure}
2   \caption{示例}\label{fig:example}
3 \end{figure}
```

## 表




```
1 \begin{table}
2   \caption{示例}\label{tab:example}
3 \end{table}
```

## 目次

```
1 \section{示例}\label{sec:example}
```

## 公式

```
1 \begin{equation}
2   a = b + c
3 \label{eq:example}
4 \end{equation}
5 如公式 \eqref{eq:example} 所示，
```

L<sup>A</sup>T<sub>E</sub>X 可以通过专用数据库文件 .bib 自动生成参考文献，很多的文献管理文件比如 EndNote , Zotero , JabRef  都可以直接导出这种格式的文件用于 L<sup>A</sup>T<sub>E</sub>X 论文中的引用。一般不需要手写数据库文件，你只需要注意每一个文献会在数据库中有一个主键，这个类似于上文的 `\label` 标记，只是要引用数据库中的文献需要使用 `\cite` 命令。

ref.bib

```
1 @article{devoftech, % 类型为期刊论文，主键为 devoftech
2   title={新时期我国信息技术产业的发展},
3   author={江泽民},
4   year={2008}
5 }
```

而让  $\text{\LaTeX}$  处理 `.bib` 数据库文件与引用有两种工作流。你可能需要查清楚如何在编辑器中设置对应的工作流，或者采用后面所提到的高级编译方式 `latexmk`。

### $\text{\BibTeX}$ + `natbib`

一般期刊提交使用这种方法，`natbib` 宏包将提供命令 `\citet`（文本引用）和 `\citep`（括号引用）。

### `biber` + `biblatex`

这是更容易自定义的方法，与  $\text{\BibTeX}$  的运作方式稍有不同。`biblatex` 提供了更加智能的引用命令。

### $\text{\BibTeX}$ + `gbt7714`

中文引用可以直接使用 `gbt7714` 宏包，但是角模式和正文模式不能混用。

### `biber` + `biblatex-gb7714-2015`

而中文引用可以使用 `biblatex` 宏包的样式 `gb7714-2015`，需使用  $\text{\XeLaTeX}$  编译。



## BibTeX + gbt7714

```
1 \documentclass{ctexart}
2 \usepackage{gbt7714}
3 \bibliographystyle{gbt7714-
   numerical}
4 % \citestyle{numbers} % 正文模式
5 \begin{document}
6   他指出了...\cite{devoftech}
7   \bibliography{ref}
8 \end{document}
```

## biber + biblatex-gb7714-2015

```
1 \documentclass{ctexart}
2 \usepackage[backend=biber,style=
   gb7714-2015]{biblatex}
3 \addbibresource{ref.bib}
4 \begin{document}
5   他在文献 \parencite{devoftech}
6   指出了...\cite{devoftech}
7   \printbibliography
8 \end{document}
```

参考文献

1

他指出了...<sup>[1]</sup>

### 参考文献

- [1] 江泽民. 新时期我国信息技术产业的发展[J]. 上海交通大学学报, 2008, 42(10): 1589.

参考文献

1

他在文献 [1] 指出了...<sup>[1]</sup>,

### 参考文献

- [1] 江泽民. 新时期我国信息技术产业的发展[J]. 上海交通大学学报, 2008, 42(10): 1589.

第二部分

SJTUThesis

- 最早由韦建文于 2009 年 11 月发布 0.1a 版
- 2018 年起由 SJTUG 接手维护
- 2019 年 6 月吴伟健重构了整个宏包的代码，升级版本号为 1.0
- 现在主要由吴伟健与张驰准备其 2.0 版本
- 最新版：1.1.0 (2022/3/26)
- 支持本科、硕士、博士学位论文以及课程论文的排版



sjtug/SJTUThesis

上海交通大学 X<sub>Y</sub>LaTeX 学位论文及课程论文  
模板 | Shanghai Jiao Tong University  
X<sub>Y</sub>LaTeX Thesis Template



2.4k



55



701

## Markdown

- + 技术文档流行
- + 语法简单
- 不内置格式控制

## Word

- + 通用论文格式
- + 所见即所得
- 进阶排版仍困难

## $\text{\LaTeX}$ SJTUThesis

- + 学术论文格式
- + 内容样式分离
- 上手有门槛

## InDesign

- + 专业杂志排版
- + 精细调整
- 过于繁琐专业

## Markdown

- + 技术文档流行
- + 语法简单
- 不内置格式控制

## Word




- + 通用论文格式
- + 所见即所得
- 进阶排版仍困难


## $\text{\LaTeX}$ SJTUThesis


- + 学术论文格式
- + 内容样式分离
- 上手有门槛


## InDesign



- + 专业杂志排版
- + 精细调整
- 过于繁琐专业



 R Markdown  
(Bookdown) 模板  
 



 配套 MathJax 渲染公式


 数学公式可以直接通过  $\text{\LaTeX}$  格式转换

 也就是 Unicode Math 输入方式

  $\text{\TeX}$  的可视前端  
LyX  Overleaf  
Rich Text 模式


  $\text{\TeX}$  的可视性改良  $\text{\TeX}_{\text{MACS}}$  

 传说用了  $\text{\TeX}$  的一些算法 

下载 推荐安装 Git  后，克隆 SJTUG 镜像仓库



```
git clone https://mirror.sjtu.edu.cn/git/SJTUThesis.git/
```

编译 推荐使用 latexmk 编译<sup>10</sup>，在不能够利用自带的 .latexmkrc 配置文件的情况下，需要查清楚在对应的编辑器中如何使用 X<sub>Y</sub>TEX + biber 编译<sup>11</sup> .






```
latexmk -xelatex main
```

在线 Overleaf 用户可以下载压缩包后，上传并采用 X<sub>Y</sub>TEX 编译方式。

<sup>10</sup>MiKTeX 用户需要手动安装 Perl 解释器  才能使用 latexmk。

<sup>11</sup>这种情况下，你可能需要查清楚如何全局安装该文档类，并刷新文件名数据库。

**第一次编译失败** 如果没有办法通过通常方式编译成功, 请尝试使用文件夹内附带   Makefile 和  Compile.bat 进行编译。

**统计字数** 编写过程中也可以使用对应的命令调用 T<sub>E</sub>Xcount 来统计正文字数。



```
make all
make clean
make cleanall
make wordcount
```



```
./Compile.bat thesis
./Compile.bat clean
./Compile.bat cleanall
./Compile.bat wordcount
```



编译论文  
清理中间文件  
↩ + 删除论文  
统计字数



## 无法使用 latexmk 578

MiKTeX 需要安装 Perl 解释器。

## CT<sub>E</sub>X 套装无法编译 446

使用最新 T<sub>E</sub>X 发行版。

## pdf<sub>L</sub>A<sub>T</sub>E<sub>X</sub> 无法编译 444

请使用 latexmk，或更改编辑器设置以 X<sub>Y</sub>L<sub>A</sub>T<sub>E</sub>X 编译。

## 缺少字体 564 598

更换字体集，或者安装对应字体。


## 缺少汉字 533 617

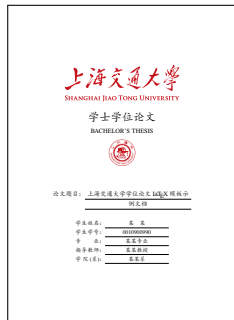
去除使用 fandol 字体集的设定。或者是安装字体后，改用 cjk-font=adobe 或 cjk-font=founder。

## README

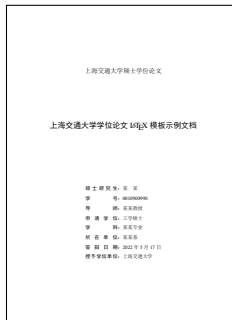
不同编辑器的设置请首先参阅 README  文档。

## Wiki

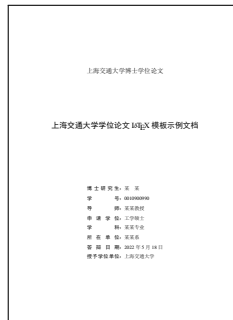
其他编译问题推荐查阅 Wiki  的使用说明部分。



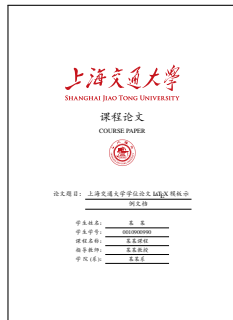
(a) 学士



(b) 硕士

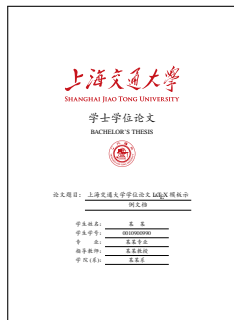


(c) 博士

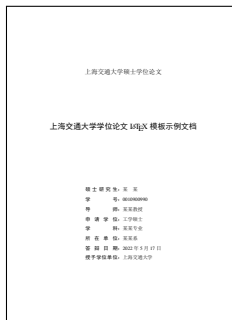


(d) 课程

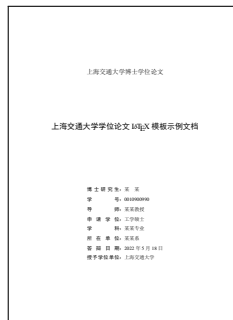
图：论文类型示例



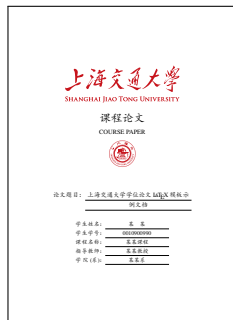
(a) bachelor



(b) master



(c) doctor



(d) course

图：论文类型示例 type

## SJTUThesis













-  bibdata
-  contents
-  figures
-  scans
-  texmf/tex/latex/sjtuthesis
  -  fd
  -  vi
  -  sjtuthesis-graduate.ltx
  -  sjtuthesis-undergraduate.ltx
  -  sjtuthesis.cls
-  main.tex
-  setup.tex








表: 文档类选项

选项	含义	相关
type=	指定论文类型	第 42 页
cjk-font=	指定字体	第 41 页
review	开启盲审模式	<a href="#">✔195</a> <a href="#">✔686</a>
twoside	双页模式	<a href="#">✔554</a>
oneside	单页模式	<a href="#">✔694</a>
openright	章从奇数页开始	<a href="#">🔍724</a>
openany	章从任意页开始	<a href="#">✔446</a>

- ├── bibdata
- ├── contents
- ├── figures
- ├── scans
- ├── texmf/tex/latex/sjtuthesis
  - ├── fd
  - ├── vi
  - ├── sjtuthesis-graduate.ltx
  - ├── sjtuthesis-undergraduate.ltx
  - └── sjtuthesis.cls
- ├── main.tex
- └── setup.tex

```
>_
git clone https://mirror.sjtu.edu.cn/git/SJTUTeX.git
cd SJTUTeX
git checkout v1
cd source
# 🐧 🍏
make build
# 🪟
xetex sjtuthesis.ins
latexmk -xelatex sjtuthesis.dtx -f
```

## SJTUThesis








-  bibdata
-  contents
-  figures
-  scans
-  texmf
-  **main.tex**
-  setup.tex

在 `main.tex` 中引入 `setup.tex` 来导入主要的信息录入与宏包加载配置。

### main.tex

```
12 % 论文基本配置，加载宏包等全局配置
13 \input{setup}
14
15 \begin{document}
16
17 %TC:ignore
18
19 % 标题页
20 \maketitle
```

## SJTUThesis








-  bibdata
-  contents
-  figures
-  scans
-  texmf
-  **main.tex**
-  setup.tex

其中 `\sjtusetup` (第 45 页) 中的 `info` 将会修改封面的信息设置。

### main.tex

```
12 % 论文基本配置，加载宏包等全局配置
13 \input{setup}
14
15 \begin{document}
16
17 %TC:ignore
18
19 % 标题页
20 \maketitle
```

## SJTUThesis

-  bibdata
-  contents
-  figures
-  scans
-  texmf
-  main.tex
-  setup.tex

## setup.tex

```
3 \sjtusetup{
4   info = {
5     title   = {上海交通大学学位论文 \LaTeX{} 模
6              板示例文档},
7     title*  = {A Sample for \LaTeX-based SJTU
8              Thesis Template},
9     author  = {某\quad{}某},
10    author* = {Mo Mo},
11  },
12  style = { header-logo-color = red,
13  },
14  name = {
15    publications = {攻读学位期间完成的论文},
16  },
17 }
```



## SJTUThesis

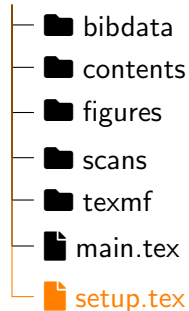


表: info 域

命令作用	中文对应选项	英文对应选项
论文标题	title	title*
关键字列表	keywords	keywords*
作者姓名	author	author*
申请学位名称	degree	degree*
院系名称	department	department*
专业名称	major	major*
导师	supervisor	supervisor*
副导师	assisupervisor	assisupervisor*
日期		date
学号		id

- └─ bibdata
- └─ contents
- └─ figures
- └─ scans
  - └─ sample-copyright-old.pdf
- └─ texmf
- └─ main.tex
- └─ setup.tex

`\copyrightpage` 可以用于插入版权页。也可接受一个可选参数，用于直接使用扫描件。📄<sup>473</sup>

```
22 % 原创性声明及使用授权书
23 \copyrightpage
24 % 插入外置原创性声明及使用授权书
25 % \copyrightpage[scans/sample-copyright-
    old.pdf]
```


## 前置部分

```
graph LR
    bibdata[bibdata]
    contents[contents]
    abstract[abstract.tex]
    nomenclature[nomenclature.tex]
    figures[figures]
    scans[scans]
    texmf[texmf]
    main[main.tex]
    setup[setup.tex]
```

前言从 \frontmatter 处开始，页码设置为大写罗马数字，主要包含摘要和目录内容。

```
main.tex
27 % 前置部分
28 \frontmatter
29
30 % 摘要
31 \input{contents/abstract}
32
33 % 目录
34 \tableofcontents
35 % ...
```

SJTUThesis

—  bibdata

—  contents

—  app\_flow\_chart.tex


—  app\_maxwell\_equations.tex

—  floats.tex


—  intro.tex

—  math\_and\_citations.tex

—  summary.tex

—  figures

—  scans












—  texmf

正文从 `\mainmatter` 处开始，页码设置为正常数字，包含正文、参考文献、附录内容。

main.tex

```
47 % 主体部分
48 \mainmatter
49
50 % 正文内容
51 \input{contents/intro}
52 \input{contents/math_and_citations}
53 \input{contents/floats}
54 \input{contents/summary}
55
56 %TC:ignore
57
58 % 参考文献
59 \printbibliography[heading=bibintoc]
```

SJTUThesis

-  bibdata
-  contents
  -  achievements.tex
  -  acknowledgements.tex
  -  digest.tex
  -  publications.tex
  -  resume.tex
-  figures
-  scans
-  texmf
-  main.tex

结尾从 `\backmatter` 处开始，页码设置为正常数字，包含致谢等相关情况。

main.tex

```
71 % 结尾部分
72 \backmatter
73
74 % 用于盲审的论文需隐去致谢、发表论文、科研成果、简历
75
76 % 致谢
77 \input{contents/acknowledgements}
78
79 % 发表论文及科研成果
80 % 盲审论文中，发表论文及科研成果等仅以第几作者注明即可，不要出现作者或其他人姓名
81 \input{contents/achievements}
```

SJTUT<sub>HESIS</sub> 定义了常用的数学环境（需要引入 `ntheorem` 或者 `amsthm` 宏包）。

## SJTUThesis

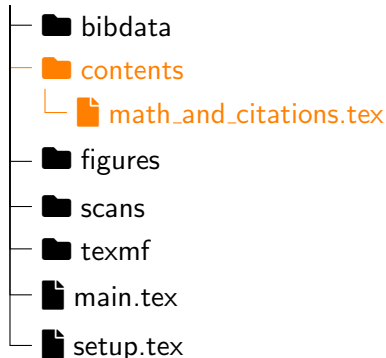










表: SJTUT<sub>HESIS</sub> 定义的数学环境

assumption	假设	lemma	引理
axiom	公理	problem	问题
conjecture	猜想	proof	证明
corollary	推论	proposition	命题
definition	定义	remark	注
example	例	solution	解
exercise	练习	theorem	定理




SJTUT<sub>HESIS</sub> 可以通过 `unimath` 选项使用 `unicode-math` 进行数学输入，注意与传统方式的区别。 [☑555](#)

## SJTUThesis






-  **bibdata**
  -  **thesis.bib**
-  **contents**
-  **figures**
-  **scans**
-  **texmf**
-  **main.tex**
-  **setup.tex**

### setup.tex

```
111 % 使用 BibLaTeX 处理参考文献
112 % biblatex-gb7714-2015 常用选项
113 % gbnamefmt=lowercase 姓名大小写由输入
    信息确定
114 % gbpub=false 禁用出版信息缺失
    处理
115 \usepackage[backend=biber,style=gb7714
    -2015]{biblatex}
116 % 文献表字体
117 % \renewcommand{\bibfont}{\zihao{-5}}
118 % 文献表条目间的间距
119 \setlength{\bibitemsep}{0pt}
120 % 导入参考文献数据库
121 \addbibresource{bibdata/thesis.bib}
```

- ② 跟 Word 的参考实现略有不同
- ③ 毕设论文的格式只要不违背《上海交通大学关于本科生毕业设计（论文）工作的指导意见》 [✔<sup>621</sup>](#)、《上海交通大学博士、硕士学位论文撰写指南》 [✔<sup>652</sup>](#) 即可，其他细节上的修改可以先搜索解决方案，再反馈给我们。
- ② 我需要转为 Word 文档
- ③ PDF 转为 Word 文档属于逆向工程 ，从  $\text{\LaTeX}$  源代码出发的转换可以使用其他工具实现 [✔<sup>480</sup>](#) [✔<sup>500</sup>](#)。



- 🗨 日常模板或  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  使用问题可以前往 Discussions  提问 (解决后别忘了  Mark as answer
- 🕒 如果是 SJTUT<sub>HESIS</sub> 项目本身的 bug 和 feature request 可以通过 Issues  反馈。
- 👤 如果你有好点子，可以贡献代码向 SJTUT<sub>E</sub>X(v1)  存储库发 PR，而后把解包结果同步到 SJTUT<sub>HESIS</sub>。
- 🏷 如果你对正在基于  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}3$  开发的新版感兴趣，也欢迎向 SJTUT<sub>E</sub>X(v2)  发 PR。
- 👤 也欢迎在 QQ 群即时讨论。



## 第三部分

# 更进一步



RECKDAHL K. 王磊 译.

L<sup>A</sup>T<sub>E</sub>X 插图指南 [EB/OL].

第三版. 2017. url<https://github.com/WenboSheng/epslatex-cn>.



包太雷.

L<sup>A</sup>T<sub>E</sub>X Notes[EB/OL].

第二版. 2021. <https://github.com/huangxg/lnotes>.



刘海洋.

L<sup>A</sup>T<sub>E</sub>X 入门 [M].

北京: 电子工业出版社, 2013.



OVERLEAF.

Learn  $\text{\LaTeX}$  in 30 minutes[EB/OL].

[https://www.overleaf.com/learn/latex/Learn\\_LaTeX\\_in\\_30\\_minutes](https://www.overleaf.com/learn/latex/Learn_LaTeX_in_30_minutes).



M.R.C. VAN DONGEN.

$\text{\LaTeX}$  and Friends[M/OL].

New York: Springer, 2012.

<https://link.springer.com/book/10.1007/978-3-642-23816-1>



MITTELBACH F., GOOSSENS M., BRAAMS J., et al.



The  $\text{\LaTeX}$  Companion[M].

2nd ed. Boston: Addison-Wesley Professional. 2004.

- 首先可以查阅宏包的附带文档。在命令行输入



```
>_
```

```
texdoc <pkg>
```

- 其次可以考虑向 T<sub>E</sub>X StackExchange  寻找解答或提出问题。
- 中文相关的问题可以在 C<sub>T</sub><sub>E</sub>X 临时论坛  搜寻。
- 提问时请附上最小工作案例（MWE）与日志文件。

5 月 18 日还会有一讲来讨论  $\text{\LaTeX}$  更加注重视觉效果的相关知识。

 SJTUBEAMER 

 图：像模像样 L<sup>A</sup>T<sub>E</sub>X 



# Happy T<sub>E</sub>Xing!

李子龙 · 从零开始使用 L<sup>A</sup>T<sub>E</sub>X 排版论文