

上海工程技术大学

本科生毕业设计论文

题目：上海工程技术大学学士学位论

文 Typst 模板

学 院 电子电气工程学院

专 业 计算机科学与技术

姓 名 这里是作者姓名

指导教师 这里是导师姓名

完成日期 2023 年 12 月

上海工程技术大学

Shanghai University of Engineering Science

毕业设计（论文）

题目：上海工程技术大学学士学位论

文 Typst 模板

学 院	电子电气工程学院
专 业	计算机科学与技术
姓 名	这里是作者姓名
指导教师	这里是导师姓名
完成日期	2023 年 12 月

目录

1 基本功能	1
1.1 标题	1
1.1.1 三级标题	1
1.2 粗体与斜体	2
1.3 脚注	3
1.4 图片	3
1.5 表格	4
1.6 公式	6
1.7 代码块	7
1.8 参考文献	8
参考文献	10
致谢	11

插图目录

1.1 Typst 网页版界面	3
1.2 答辩委员会名单	5

表格目录

1.1 booktab 示例	6
----------------------	---

代码目录

1.1 默认表格	5
1.2 一个简单的 Python 程序	8
1.3 codeblock 命令的实现	8

摘要

这是中文摘要。间起北家。许士业小健明他该有金，升队民国团，理提己？经局由情色于！的经举国有发为题他什常直流望字防多学。讲来破会全闻室；个这提金自办太完作，明然内然我山是？力过究可象定开不展法南顾取什说；候社进。动动里上谢再苦，务的里度成的文电儿说流像状好音！得年跑电白线经合电终也玩青满情子，舞代公续斯会了作电李多。回受轮了随自西同；长样头些实林好分子，维龙布一市由本为会，入令比格带，依为健，竟少个共得家他们集：由德环？很里习用人我方乐员在开亲书突！建我这省们高意生开日一；张他下心开图则所了热呢都难直着受可！速己了很鱼不上痛便阳一旦物月爸适台得科事下，乐着馆模校。

公明议要所布以流式雄道人他早在，的立样多的作整使政把升问，地灵失养中系说到出因由为身取到但大的著做区科过受小取她红看作效花。不就品国家房星生率老自！亚究先常，人的初指一全作保业好是只青国模一两力：以因后提开交馆明画因一装车多小基由，情安前在来孩地几空生是分看原？

关键词： 论文，typst，模板

ABSTRACT

This is English abstract of the thesis. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et.

Keywords: thesis, typst, template

1 基本功能

1.1 标题

Typst 中的标题使用 `=` 表示，其后跟着标题的内容。`=` 的数量对应于标题的级别。

除了这一简略方式，也可以通过 `heading` 函数自定义标题的更多属性。具体可以参考文档中的有关内容。

下面是一个示例：

代码	渲染结果
<code>#heading(level: 2, numbering: none, outlined: false, "二级标题")</code>	二级标题
<code>#heading(level: 3, numbering: none, outlined: false, "三级标题")</code>	三级标题
<code>#heading(level: 4, numbering: none, outlined: false, "四级标题")</code>	四级标题
<code>#heading(level: 5, numbering: none, outlined: false, "五级标题")</code>	五级标题

需要注意的是，这里的样式经过了本模板的一些定制，并非 Typst 的默认样式。

1.1.1 三级标题

这是三级标题下的正文。Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliquam quaerat voluptatem. Ut enim aequaleam animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguere possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et.

1.1.1.1 四级标题

本模板目录的默认最大深度为 3，即只有前三级标题会出现在目录中。如果需要更深的目录，可以更改 `outlinedepth` 设置。

1.2 粗体与斜体

与 Markdown 类似，在 Typst 中，使用 `*` 表示粗体，使用 `_` 表示斜体。下面是一个示例：

代码	渲染结果
遵循 sysu-thesis 的惯例，使用 <code>*黑体*</code> 表示粗体， <code>_楷体</code> 表示斜体。 <code>*bold*</code> and <code>_italic_</code> are very simple.	遵循 sysu-thesis 的惯例，使用 黑体 表示粗体， <i>楷体</i> 表示斜体。 bold and <i>italic</i> are very simple.

由于绝大部分中文字体只有单一字形，这里遵循 sysu-thesis 的惯例，使用**黑体**表示粗体，*楷体*表示斜体。但需要注意的是，由于语法解析的问题，`*...*` 和 `_..._` 的前后可能需要空格分隔，而这有时会导致不必要的空白。如果不希望出现这一空白，可以直接采用 `#strong` 或 `#emph`。

代码	渲染结果
对于中文情形，*使用 * 加粗* 会导致额外的空白， <code>#strong</code> [使用 <code>\#strong</code> 加粗]则不会。	对于中文情形，使用 * 加粗 会导致额外的 空白，使用 <code>#strong</code> 加粗则不会。

1.3 脚注

从 v0.4 版本开始，Typst 原生支持了脚注功能。本模板中，默认每一章节的脚注编号从 1 开始。

代码	渲染结果
Typst 支持添加脚注 <code>#footnote</code> [这是一个脚注。]。	Typst 支持添加脚注 ¹ 。

1.4 图片

在 Typst 中插入图片的默认方式是 `image` 函数。如果需要给图片增加标题，或者在文章中引用图片，则需要将其放置在 `figure` 中，就像下面这样：

代码	渲染结果
<pre>#figure(image("images/chapter1/1-writing-app.png", width: 100%), caption: "Typst 网页版界面",) <web></pre>	 <p>图 1 Typst 网页版界面</p>

¹这是一个脚注。

图 1.1 展示了 Typst 网页版的界面。更多有关内容，可以参考 `about.developers` 中介绍了 Typst 的主要开发者。代码中的 `<web>` 是这一图片的标签，可以在文中通过 `@web` 来引用。

1.5 表格

在 Typst 中，定义表格的默认方式是 `table` 函数。但如果需要给表格增加标题，或者在文章中引用表格，则需要将其放置在 `figure` 中，就像下面这样：

代码	渲染结果
----	------

代码 1 默认表格

```
#figure(  
  tablex(  
    columns: (auto, auto, auto, auto),  
    inset: 10pt,  
    align: horizon,  
    [*姓名*],[*职称*],[*工作单位*],[*职责*],  
    [李四],[教授],[#(学校)],[主席],  
    [王五],[教授],[#(学校)],[成员],  
    [赵六],[教授],[#(学校)],[成员],  
    [钱七],[教授],[#(学校)],[成员],  
    [孙八],[教授],[#(学校)],[成员],  
  ),  
  caption: "答辩委员会名单",  
)<table>
```

姓名	职称	工作单位	职责
李四	教授	上海工程大学	主席
王五	教授	上海工程大学	成员
赵六	教授	上海工程大学	成员
钱七	教授	上海工程大学	成员
孙八	教授	上海工程大学	成员

图 2 答辩委员会名单

对应的渲染结果如图 1.2 所示。代码中的 <table> 是这一表格的标签，可以在文中通过 @table 来引用。

默认的表格不是特别美观，本模板中提供了 booktab 函数用于生成三线表，表 1.1 是一个示例。代码中的 <booktab> 是这一表格的标签，可以在文中通过 @booktab 来引用。

代码	渲染结果
----	------

<pre>#booktab(width: 100%, aligns: (left, center, right), columns: (lfr, lfr, lfr), caption: ['booktab` 示例], [左对齐], [居中], [右对齐], [4], [5], [6], [7], [8], [9], [10], [], [11],)<booktab></pre>	表 1 booktab 示例		
	左对齐	居中	右对齐
	4	5	6
	7	8	9
	10		11

1.6 公式

式 (1.1) 是一个公式。代码中的 <eq> 是这一公式的标签，可以在文中通过 @eq 来引用。

代码	渲染结果
<code>\$ E = m c^2 \$<eq></code>	$E = mc^2$ (1.1)

式 (1.2) 是一个多行公式。

代码	渲染结果
<pre>\$ sum_(k=0)^n k &= 1 + ... + n \ &= (n(n+1)) / 2 \$<eq2></pre>	$\sum_{k=0}^n k = 1 + \dots + n$ $= \frac{n(n+1)}{2}$ <div>(1.2)</div>

式 (1.3) 到式 (1.6) 中给出了更多的示例。

代码	渲染结果
----	------

<code>\$ frac(a^2, 2) \$ <eq3></code>	$\frac{a^2}{2}$	(1.3)
<code>\$ vec(1, 2, delim: "[") \$</code>		
<code>\$ mat(1, 2; 3, 4) \$</code>	$\begin{bmatrix} 1 \\ 2 \end{bmatrix}$	(1.4)
<code>\$ lim_x =</code> <code> <code>op("lim", limits: #true)_x \$ <eq6></code></code>	$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$	(1.5)
	$\lim_x = \lim_x$	(1.6)

1.7 代码块

像 Markdown 一样，我们可以在文档中插入代码块：

代码	渲染结果
<code>```c</code> <code>int main() {</code> <code> printf("Hello, world!");</code> <code> return 0;</code> <code>}</code> <code>```</code>	<code>int main() {</code> <code> printf("Hello, world!");</code> <code> return 0;</code> <code>}</code>

如果想要给代码块加上标题，并在文章中引用代码块，可以使用本模板中定义的 `codeblock` 命令。其中，`caption` 参数用于指定代码块的标题，`outline` 参数用于指定代码块显示时是否使用边框。下面给出的代码 1.2 是一个简单的 Python 程序。其中的 `<code>` 是这一代码块的标签，意味着这一代码块可以在文档中通过 `@code` 来引用。

代码	渲染结果
----	------

<pre>#codeblock(``python def main(): print("Hello, world!") ``, caption: "一个简单的 Python 程序", outline: true,) <code></pre>	<div>代码 2 一个简单的 Python 程序</div> <div> <pre>def main(): print("Hello, world!")</pre> </div>
--	--

代码 1.3 中给出了本模板中定义的 codeblock 命令的实现。

代码 3 codeblock 命令的实现

```
#let codeblock(raw, caption: none, outline: false) = {
  figure(
    if outline {
      rect(width: 100%)[
        #set align(left)
        #raw
      ]
    } else {
      set align(left)
      raw
    },
    caption: caption, kind: "code", supplement: "代码"
  )
}
```

1.8 参考文献

Typst 支持 BibLaTeX 格式的 .bib 文件，同时也新定义了一种基于 YAML 的文献引用格式。要想在文档中引用参考文献，需要在文档中通过调用 bibliography 函数来引用参考文献文件。下面是一个示例：

代码	渲染结果
<p>可以像这样引用参考文献：@wang2010guide 和 @kopka2004guide。</p> <pre>#bibliography("ref.bib", style: "ieee")</pre>	<p>可以像这样引用参考文献： [1] 和 [2]。</p>

注意代码中的 "ref.bib" 也可以是一个数组，比如 ("ref1.bib", "ref2.bib")。

参考文献

- [1] 王晓华, 闫其涛, 程智强, 和 张睿, 《科技论文中文摘要写作要点分析》, 编辑学报, 期 S1, 页 53–55, 2010.
- [2] H. Kopka, P. W. Daly, 和 S. Rahtz, *Guide to LATEX*, 卷 4. Addison-Wesley Boston, MA, 2004.

致谢