

zL^AT_EX manual

Eureka

2024 年 6 月 22 日

目录

第一部分 Document	1		
第一章 zL^AT_EX 文档类	2		
1.1 基本介绍	2		
1.2 Set Up zL ^A T _E X	2		
1.2.1 兼容情况	2		
1.2.2 加载 zL ^A T _E X	2		
1.2.3 额外设置	3		
1.2.4 最小工作示例	3		
1.2.5 在线体验	4		
1.3 宏包机制	4		
1.3.1 基本宏包	4		
1.3.2 语言类宏包	4		
1.3.3 数学字体类宏包	5		
1.3.4 超链接宏包	6		
1.4 文档类选项与命令	6		
1.4.1 导言区	6		
1.4.2 快捷命令	11		
1.4.3 zl _{at} exSetup	12		
1.4.4 zl _{at} exColorSetup	12		
1.4.5 zl _{at} exFramed	13		
1.5 zL ^A T _E X 接口	13		
1.5.1 命令声明	14		
1.5.2 盒子接口	14		
1.5.3 绘图接口	14		
1.5.4 算法与数据结构	14		
1.5.5 计数器	15		
1.5.6 接口完善	15		
1.6 自定义	15		
1.6.1 封面	15		
1.6.2 目录	16		
1.6.3 子目录	17		
1.6.4 页眉页脚	17		
1.6.5 章节格式	17		
1.7 字体配置	19		
1.7.1 一些术语	19		
1.7.2 New Computer Modern	19		
1.7.3 模板预设	19		
1.7.4 查看系统字体	20		
1.7.5 读懂字体警告	22		
1.7.6 字体族声明	26		
1.7.7 Emoji	27		
1.7.8 数学字体	27		
1.7.9 常见字体问题	28		
1.7.10 Furthermore	30		
1.8 数学环境	30		
1.8.1 常用数学环境	30		
1.8.2 使用方法	31		
1.8.3 定理类环境样式	32		
1.8.4 证明类环境	34		
1.8.5 注意事项	35		
1.8.6 自定义数学环境	36		
1.9 标签与引用	36		
1.9.1 Footnote	36		
1.9.2 cleveref	36		
1.9.3 figure and table	37		
1.10 列表环境	37		
1.11 文献引用	37		
1.11.1 基本使用	37		
1.11.2 使用 tip	38		
1.12 索引	38		
1.12.1 使用方法	38		
1.12.2 Bug	39		
第二部分 索引与参考	41		
第二章 部分命令/名词索引	42		
第三章 参考文献	43		

第三部分 Implement	44		
第四章 zL^AT_EX Implement	45		
4.1 文档选项	45		
4.1.1 Key-Value	45		
4.1.2 Load Options	45		
4.2 class load	47		
4.3 compile engine	48		
4.4 Layout	48		
4.4.1 geometry	48		
4.4.2 Slide mode	49		
		4.4.3 fancyhdr	50
		4.5 math	51
		4.5.1 math env theme	51
		4.5.2 math environment	52
		4.5.3 Math alias	56
		4.6 structure Style	57
		4.6.1 hyperref/index and bib	57
		4.6.2 cref name	57
		4.6.3 contents	58
		4.6.4 Chapter and section	58

第一部分

Document

zL^AT_EX 文档类

1.1 基本介绍

zL^AT_EX 文档类默认基于 article 文档类, 但是你仍然可以在加载本文档类时选择加载其他的文档类, 通过设置选项 `<class>` 的值为 article 或者是 ctexbook 即可. 通过更换默认的文档类, 从而满足使用者的不同需求, 目前本模板可以用于以下场景:

- 撰写书籍或者笔记
- 讨论班的 Slide 制作 (可以和 article 无缝切换)

zL^AT_EX 的制作初衷便是: 让使用者可以方便进行书籍和笔记的撰写以及日常汇报 slide 的无缝切换. zL^AT_EX 全部由 L^AT_EX3 进行编写, 采用 `key-value` 的方式进行选项配置, 对于作者来说: 方便后续的模板拓展和维护; 对于用户来说: 使用键值对可以减轻用户记忆命令参数这一负担, 方便用户使用命令. 如果使用者熟悉 L^AT_EX, 那么花费不到 10min 的时间, 你便可以轻松使用本文档类完成如上任务, 减少不必要的工作.

1.2 Set Up zL^AT_EX

1.2.1 兼容情况

目前本文档类 zL^AT_EX 还没有登陆 CTAN, 未来可能也没有这个打算, 本模板还没有完全开发完成. 由于本文档类使用的部分 L^AT_EX3 命令在老版本下并不存在, 所以如果你的 T_EXLive 过于老旧, 则可能出现无法编译的情况. 目前已知 zL^AT_EX 文档类在各平台的兼容情况为:

Windows : T_EXLive 最低版本 2022

Linux : T_EXLive 最低版本 2022

MacOS : 兼容 MacT_EX2024(老版也应兼容)

1.2.2 加载 zL^AT_EX

由于 zL^AT_EX 还没有传入 CTAN(未来可能会考虑), 所以想要使用此文档类, 可以有如下的两种方法:

- 把此文档类-zlatex.cls 放入当前项目文件夹下

- 在命令行运行命令: `kpsewhich -var-value=TEXMFHOME`, 然后把 `zlatex.cls` 放入此路径下的 `tex/latex/` 子目录下. 在 Windows 上一般是: `C:/Users/<name>/texmf/`, 在 Linux 下一般是 `~/texmf/`, 具体路径以自己的实际情况为准.

1.2.3 额外设置

z \LaTeX 默认只加载部分的基础宏包, 所以想要实现其它的功能还请自行引入相关的宏包; z \LaTeX 引入的宏包机制请参见表 1.1. 在默认情况下本模板即可呈现一个比较好的效果, 所以不熟悉 \LaTeX 的用户不用担心本模板配置选项过于复杂. 想要马上开始体验? 请参见“[小节 1.2.4](#)”的最小写作示例.

1.2.4 最小工作示例

z \LaTeX 的最小工作示例如下¹. 首先是中文写作示例, 默认加载 `article` 文档类, 如果喜欢使用 `book` 文档类, 可以在加载文档类时指定 `class=book`.

```
% compile engine: xelatex
\documentclass[lang=cn]{zlatex}

\begin{document}
% some preface
% \tableofcontents

% wrting your document here ...
\end{document}
```

其次是英文写作示例 (此时为 `book` 文档类), 你需要修改的地方只有两处; 首先就是把语言选项改为 `lang=en`, 其次便是把编译方式改为 `pdflatex`.

```
% compile engine: pdflatex
\documentclass[class=book]{zlatex}

\title{<title>}
\author{<author>}
\date{<date>}
\begin{document}
\maketitle
\frontmatter
% some preface
% \tableofcontents
% some claim etc.
\mainmatter
```

¹ 导言区的配置可能需要根据自己的实际情况加以调整, 详细配置请参见后文

```
% writing your document here ...
\end{document}
```

1.2.5 在线体验

为了让部分用户可以直接体验到 $\text{zL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ，免去“繁杂”的环境配置，我已将本模板部署在 Overleaf 上，地址为: [Overleaf ZLaTeX Project](#)，直接打开此地址即可体验。Overleaf 上的项目内容为本手册的源码，不含有 zTikZ 文档。由于部分的技术原因， zTikZ 请在本地体验。

1.3 宏包机制

$\text{zL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 文档类会根据用户指定的选项自动处理和加载对应的宏包，所以 $\text{zL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 文档类在不同的导言区选项声明下加载的宏包和命令是不同的。本文档类内置导言区选项输出命令: `\zlatexOptions`，用于打印此时文档类 $\text{zL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 接收到的选项。比如此时文档类接收到的选项为:

Class Options: cn- oneside - 10pt

以下为详细的宏包加载信息:

1.3.1 基本宏包

基本宏包，意味着不管你的导言区如何的配置，这些宏包都是会加载的。宏包列表如下:

expl3	l3keys2e	framed	geometry
fancyhdr	amsfonts	amsmath	amsthm
xcolor	biblatex	indextools	sidenotes
cleveref	graphicx	float	titletoc
titlesec	esint		

表 1.1: $\text{zL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 文档类基本宏包

1.3.2 语言类宏包

根据不同的文档类语言， $\text{zL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 会加载不同的和语言相关的宏包，在 `lang=en(cn)` 下的宏包加载列表分别为:

lang=en	inputenc(pdfTeX)	fontenc
lang=cn	fontspec	ctex

表 1.2: $\text{zL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 文档类语言宏包

在 L^AT_EX 排版中,正确使用引号的方法是分别使用 ```和`'` 排版单引号 ‘和’, 分别用 ````和`''` 排版双引号 “和”. ````和`'`会被转换成 opening quotation marks, 后两者会被转化成 closing quotation marks. 另外, 在排版要求中, 当两层双引号嵌套使用时, 其外层需要使用双引号, 而内层则应该使用单引号, 并且不同的语言中, 用于表示引号的字符也不完全一致. 为解决这些排版中的问题, 可以使用 csquotes 宏包实现引号的灵活排版², 比如要使用双引号文本, 那么此时你就可以使用命令 `\enquote {<your text>}`.

但是你也可以直接在源文件中输入对应的 unicode, 如果你使用的是 Xe_TE_X 或者是加载了 inputenc 的 utf8 选项.

1.3.3 数学字体类宏包

本模板会根据导言区选项加载不同的数学字体. 为方便新手较使用不同数学字体, 本模板预设了几个简单的数学字体. 用户只需配置导言区即可实现不同数学字体的加载. 具体的宏包加载规则如下:

- `mathSpec={font=<none>}`: 不加载任何的数学字体宏包, 采用默认数学字体
- `mathSpec={font=newtx}`: 加载宏包 newtxmath
- `mathSpec={font=euler}`: 加载命令 `\RequirePackage[OT1,euler-digits]{eulervm}`
- `mathSpec={font=mtpro2}`: 加载命令 `\RequirePackage[lite,subscriptcorrection,slantedGreek,nofontinfo]{mtpro2}`
- `font=mathpazo`: 加载命令 `\RequirePackage{mathpazo}`, 同时保留了文档类默认的正文字体样式.

如果使用者在导言区还指定了选项 `mathSpec={alias=true}`, 那么 zL^AT_EX 此时还会额外加载的宏包: amssymb, mathtools, bm.

除此之外, zL^AT_EX 还默认加载了宏包 esint, 用于提供部分积分符号变体. 下面仅举几例, 更多的人信息请参见 esint 宏包文档:

$$\iiint \quad \int \cdots \int \quad \oint \quad f \quad \int$$

其实还有一个包和 esint 类似, 叫做 pdfMsym. 关于后者的使用请参见 pdfMsym 官方文档: [pdfMsym](#).

如果用户想自己设置文档的数学字体, 那么请参见”字体配置[节 1.7](#)”中的数学字体配置部分.

²本内容摘自 [l^atexstudio](#)

1.3.4 超链接宏包

zL^AT_EX 文档类现在不默认加载宏包 `hyperref`, 但是在加载文档类时指定 `hyper=true` 即可加载此宏包以及对应的配置. 其实还是考虑到编译速度的问题, 一般在初稿阶段: 作者是不需要这部分超链接功能的, 此宏包引入后需要的多次编译也十分不必要, 所以我便把 `hyperref` 从基本宏包中移除.

1.4 文档类选项与命令

本模板具有丰富的配置选项, 采用键值对 [`<key 1>=<value 1>`, `<key 2>=<value 2>`] 的形式对各个选项就行指定, 和具体的指定顺序无关, 具体的可配置项和可用的配置值参见本节后续内容以及图 1.1. 现在先简要说明 zL^AT_EX 文档类的主要配置内容:

- 模板语言: 目前本模板类仅支持中文和英文两种语言.
- 文档类选择: 可以在加载本文档类时指定 `article`, `book`, `ctexbook`.
- 展示方式: 默认情况下以普通的文档展示, 也可以指定 `layout={slide}` 选项后把文档转化为 `beamer`, 详细使用请参见: 小节 1.4.1.
- 页面设置: 页边距, 版心, 页眉页脚等的指定; 目前本模板还未整理好对应的接口, 但用户可以通过 `geometry/fancyhdr` 宏包提供的接口进行设置.
- 边注(margin): 包含编注图片或文字注释, 其中边注图片依赖宏包 `sidenotes`.
- 数学字体: 本模板预设了 5 种数学字体配置.
- 字体配置: 用户可根据自己喜好配置相应的中英文字体, 具体配置方法参见: 节 1.7.

1.4.1 导言区

目前 zL^AT_EX 的配置选项可以在文档类加载时指定, 也可以通过命令 `\zlatexSetup` 进行指定. 键值对被划分为两个层级: 第一级中主要的键为: `<layout>`, `<mathSpec>`, ``, `<bib>`, `<classOption>`, `<lang>`, `<toc>`. 其中前四个键 (key) 具有自己的独立的子键 (sub-key), 后两者不具有自己的子键, 直接指定即可. 关于各层 key-value 的关系请见图 1.1.

从图 1.1 可以看出, zL^AT_EX 的配置是比较复杂的, 但是刚接触本文档类时, 用户也不必担心, 默认选项配置下已经能够得到一个效果比较好的文档了. 下面详细说明各个键的指定方式和作用.

■ `classOption`

这里的 `<classOption>` 表示你加载的对应文档类的合法 options. 比如 `article` 文档类, 合法的一个设置为: `classOption = {10pt, oneside, fleqn}`. 如果你在加载文档类时指定的 `<class>` 为 `ctexbook`, 那么此时你可以在 `<classOption>` 中填入如下像: `zihao=5, 12pt, heading=false, linespread=1.3` 等 `ctex` 手册中说明的合法选项. 若对于默认的 `article`, `book` 文档类不熟悉, 可以参见: [latex-documentclass-options](#).

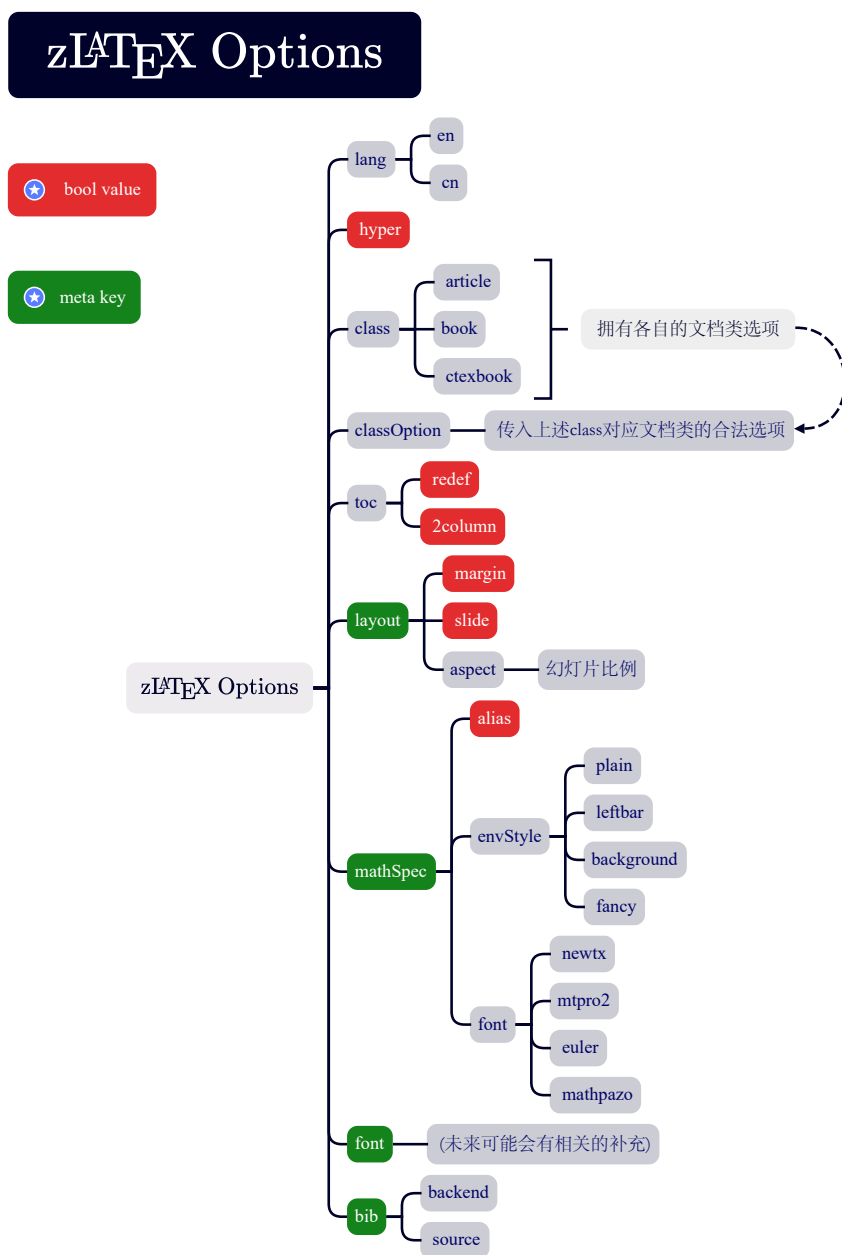


图 1.1: zlatex options map

➡ lang

键 `<lang>` 的合法值为 `en`, `cn`; 第二个关于文档语言的指定是比较容易理解的, 但是还是得说明对应的一些注意事项。`lang=cn` 或者 `class=ctexbook` 时仅支持为 $\text{X}\text{\LaTeX}$ 编译; 在指定 `lang=en` 时, $\text{pdf}\text{\LaTeX}$, $\text{X}\text{\LaTeX}$ 二者都是可以接受的, 但是建议使用 $\text{pdf}\text{\LaTeX}$. 因为在指定 `lang=en` 时部分的西文宏包可能会有冲突的危险. 当指定 `lang=en`, 并且采用 $\text{pdf}\text{\LaTeX}$

进行编译时, $\text{z}\text{L}\text{A}\text{T}\text{E}\text{X}$ 会引入宏包 `inputenc`, 然而此宏包对 $\text{X}\text{L}\text{A}\text{T}\text{E}\text{X}$ 是没有适配的.

► `toc`

键 `<toc>` 的合法值为 `redef`, `2column`, 这一选项主要用于文档的目录格式设置, 可以设置目录页对应的标题, 目录的栏数. `<toc>` 选项可以同时指定 `redef`, `2column`, 分别代表重定义目录名 (居中加粗), 以及排版双栏目录. 后续可能会把这个键 `<toc>` 放入 `<layout>` 中. 所以目前此接口还不稳定, 请谨慎使用.

► `bib`

本模板的默认参考文献后端为 `biber`, 可以在导言区通过 `backend=bibtex` 来更改默认后端为 `bibtex`. 这说明, 在默认配置下, 你编译文档时应使用 `biber` 命令, 而非 `bibtex` 命令. 同样的, 可以通过 `<bib>` 的子键 `<source>` 来指定参考文献对应的文件名, 默认文件名为: `ref.bib`.

► `layout`

`<layout>` 键可用于指定整个文档布局: 版心, 页边距, 边注间距等, 也可以指定文档最终的呈现方式: 普通的文档, 还是展示所用的 `slide`. 目前模板提供子键 `<margin>`, `<slide>` 用于设置对应布局. 此键合法值为 `bool` 值, 指定方式如下:

```
% {margin} <=> {margin=true}
\documentclass[
  layout={margin, slide}
]{zlatex}
```




默认情况下 `<margin>` 的值为 `false`. 如果使用了上述键值指定, 那么此时的文档就会产生一个边注. 一般情况下, 如果不是撰写相应的书籍, 使用者是不需要加载这个选项的. **需要注意:** `margin=false` 只有在 `oneside` 时才有效, 否则会抛出警告. 如果原文档中含有 `\marginpar` 命令, 那么在指定 `margin=false` 后, 对应的 `\marginpar` 环境, 会被替换为 `framed` 宏包提供的 `leftbar` 环境.

注记 1.4.1 默认情况下 `twoside` 选项对应的 `margin` 和 `margin` 选项对于的 `margin` 宽度是不同的, 后者比前者宽一些.

如果你需要将最终的文档编译为演示文档, 除了必须加载 `<slide>` 选项外, 你还可以指定幻灯片比例, 使用 `<aspect>` 子键进行设置, 默认的比例为 `16:9`, 默认的长度单位为厘米 (`cm`). 如果你想要设置幻灯片的比例为 `12:9`, 那么导言区按照如下设置即可:

```
\documentclass[
  layout={slide, aspect=12|9},
]{zlatex}
```

也就是使用符号 “|” 进行两个维度上距离的分割. 关于 `slide` 模式下的主题: 目前还不支持自定义, 后续可能会进行拓展. 如果仅仅是想要修改 `slide` 中不同元素对应的颜色, 那么只需重新定义如下三个颜色:

- `slideRed:` 
- `slideGray:` 
- `slideWhite:` 

下面为一默认 `slide` 模式样例:

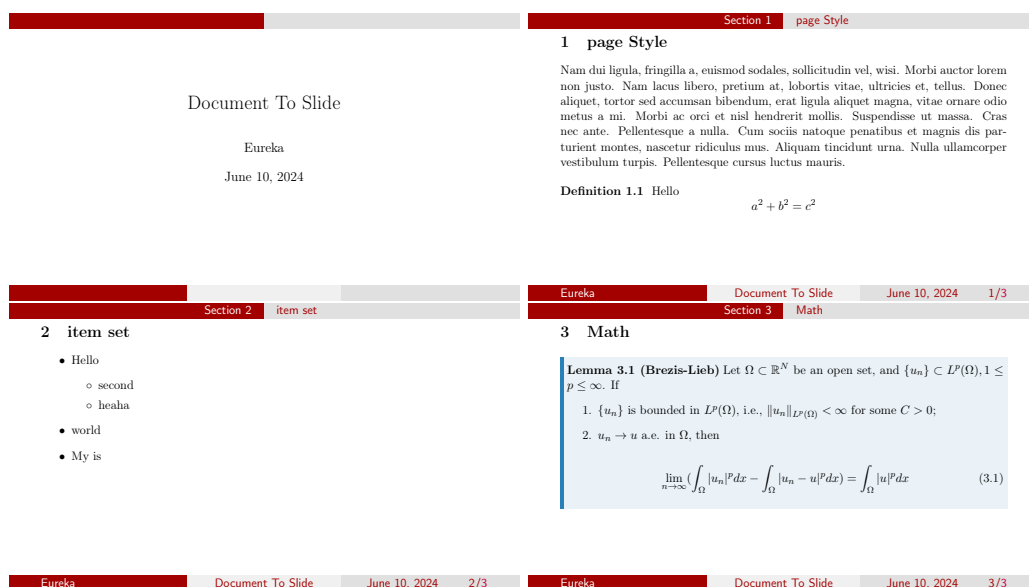


图 1.2: Slide 样例

目前 `slide` 模式还处于一个很不成熟的阶段, 请谨慎使用. 在正式场合请使用默认的 `beamer` 文档类.

注记 1.4.2 由于本模板可以切换单双面布局, 所以为了保证 `titlepage` 中文字的正确居中, 本文档类对原始命令: `\maketitle` 进行了重定义. 除了本文档类定义的 `\maketitle` 命令外, 本文档类还提供了命令 `\Maketitle` 用于排印简洁的封面页, 效果相当于原始命令 `\title` + `\maketitle`. 本模板保留原始命令为 `\orimaketitle`, 那么这三个命令到底该怎么用? 下面给出一个情况下的应用规则:

- 在 `slide mode` 下使用 `\orimaketitle` 命令
- 在 `oneside` 布局下使用 `\maketitle`, `\maketitle` 均可
- 在 `twoside` 布局下使用 `\Maketitle` 命令排印原始封面页 + 居中 + 换页, `\maketitle` 命令排印自定义的封面页 + 换页

mathSpec

此选项可用于设置文档类对应的数学字体等内容, 本文档类预设了几种数学字体和定理环境样式以及部分数学命令别称. 详细信息如下:

- 数学字体 (`font=name`): 可用 `name:newtx`, `mtpro2`, `euler`, `mathpazo`
- 定理类环境样式 (`envStyle=name`): 可用 `name:plain`, `leftbar`, `background`, `fancy`
- 字体命令 `alias` 加载 (`<alias>`): 可以直接输入 `alias`, `alias=false` 或者是输入 `alias=true`, `alias=false`

一个简单的例子: 如果你想设置键 (`<mathSpec>`) 对应子键: `font`, `envStyle`, `alias` 的值 (`value`), 其中最后一个 `key` 为 `bool` 值. 那么你可以在导言区中这样设置:

```
\documentclass[
  mathSpec={
    alias,
    envStyle=background,
    font=newtx
  },
]{zlatex}
```

上述的声明表示, 文档类的定理类数学环境样式为: `background`, 数学字体为 `newtxmath`. 同时加载了数学命令别名. 其实所谓的命令别名就是如下常用命令的缩写:

```
\mathbb{X}   \mathcal{X}   \mathrm{X}   ...
```

可以替换为如下更加简洁的形式:

```
\B{X}   \C{X}   \R{X}   ...
```

对于 \LaTeX , 新手千万不要秉持着: “少些就是赚到” 的观念, 你保持不住. 但是你完全可以定义空格为你的 `delimiter`, 以此来划分你的宏参数. 所以你可以写这种东西:

```
% define macro
\def\upb#1 #2 {\mathbb{#1}^{\#2}}
% use macro
\upb R N^2 = \upb R N \times \upb R N \cdots
```

这个结果和你添加括号无异, 但是一旦你落下了某个重要的空格, 那么轻则编译结果不正确, 重则编译不通过.

关于 `alias` 配置的宣传明细节, 请参见章节“快捷命令”[小节 1.4.2](#). `alias` 选项可根据个人习惯进行选择加载, 默认情况下并不会加载. 但在加载此选项后, 默认的两个 \LaTeX 宏 `\S`, `\ll` 会被重定义, 分别被更名为 `\ss`, `\LL` : (\S , \ll).

需要注意: 本模板提供的部分数学字体并不一定符合每人审美, 且由于本模板在 `Linux` 环境下开发, 所以在平台迁移后部分字体可能会有细微的差别. 其中的 `mtpro2` 字体并非免费字体, 请不要用于任何的商业用途. 若用户想自己配置数学字体, 那么可以参见后文“`Unicode-math`: [小节 1.7.8](#)”

关于 `envStyle` 的详细信息请参见：“常用数学环境”[小节 1.8.1](#)”

1.4.2 快捷命令

在你决定加载选项 `alias` 前的进一步说明：默认自定义命令并不一定符合你的习惯，所以请一定谨慎加载此选项。在 `alias=true` 后， \LaTeX 会进行如下命令的声明/重定义，以及部分宏包的加载：

```
\RequirePackage{amssymb, mathtools}
\RequirePackage{bm}
% Math Font
\newcommand{\dd}{\mathrm{d}}
\newcommand{\C}[1]{\ensuremath{\mathcal{#1}}}
\let\ss\S
\renewcommand{\S}[1]{\ensuremath{\mathscr{#1}}}
\newcommand{\B}[1]{\ensuremath{\mathbb{#1}}}
\newcommand{\FF}[1]{\ensuremath{\mathbf{#1}}}
\newcommand{\F}[1]{\ensuremath{\bm{#1}}}
\newcommand{\R}[1]{\ensuremath{\mathrm{#1}}}
\newcommand{\K}[1]{\ensuremath{\mathfrak{#1}}}
% Math Arrow
\newcommand{\lr}{\ensuremath{\longrightarrow}}
\let\LL\ll
\renewcommand{\ll}{\ensuremath{\longleftarrow}}
\newcommand{\equ}{\ensuremath{\Longleftrightarrow}}
\newcommand{\sr}{\ensuremath{\longmapsto}}
\newcommand{\lrr}[2][\ensurmath{\xrightarrow{#1}{#2}}]{\ensurmath{\xrightarrow{#1}{#2}}}
\renewcommand{\lll}[2][\ensurmath{\xleftarrow{#1}{#2}}]{\ensurmath{\xleftarrow{#1}{#2}}}
\newcommand{\ns}{\ensuremath{\varnothing}}
\newcommand{\A}{\ensuremath{\forall}}
% Math Operator
\newcommand{\alt}{\ensuremath{\mathrm{Alt}}\;}
\newcommand{\sgn}{\ensuremath{\mathrm{sgn}}\;}
\newcommand{\curl}{\ensuremath{\mathrm{curl}}\;}
\newcommand{\grad}{\ensuremath{\mathrm{grad}}\;}
\newcommand{\trace}{\ensuremath{\mathrm{trace}}\;}
\renewcommand{\div}{\ensuremath{\mathrm{div}}\;}
```

若用户认为上述的某些数学运算符声明并不是那么规范，如果有必要，可以自行进行修改，可参见命令 `\DeclareMathOperator` , `\DeclareMathAlphabet` .

1.4.3 zlatexSetup

本模板同样也提供了命令`\zlatexSetup` 用于在导言区对整个文档进行设置，其使用方法和前面的几乎是一致的。如下两段声明等价：

```
\documentclass[
  mathSpec={envStyle=background, alias, font=mathpazo},
  toc={redef, 2column},
  bib={backend=bibtex}
]{zlatex}
```

```
\zlatexSetup{
  mathSpec={
    envStyle=background,
    font=mathpazo,
    alias
  },
  toc={redef, 2column},
  bib={backend=bibtex}
}
```

但是命令`\zlatexSetup` 并不能对所有键进行指定，如 `<class>`, `<lang>` 这两者只能在加载文档类时声明，不能在后续使用此命令声明。同时，本命令不包含本文档类的颜色主题设置，和颜色主题设置相关类容请参见命令：`\zlatexColorSetup`，关于本文档类的默认色彩设置请参见“模板配色小节 1.4.4”。

如果你加载了宏包 `ctex`，那么此时你也可以使用此宏包提供的命令`\ctexset` 进行对应元素的设置。

1.4.4 zlatexColorSetup

zL^AT_EX 提供了命令`\zlatexColorSetup` 用于整个模板的配色设置，可供用户配置的选项：

- `hyperref` 宏包对应的颜色，对应的键：`<link>`, `<url>`, `<cite>`，三者颜色默认不同。
- `chapter` 章节计数器颜色，对应的键：`<chapter>`。
- `chapter` 章节 `chapter ruler` 的前景色，对应键：`<chapter-rule>`。
- 所有数学环境对应的颜色，对应的键：`<math-env-name>`，如 `<axiom>`, `<definition>`, `<theorem>`, `<remark>` 等。

下面给出一个设置文档类配色的示例代码：

```
\zlatexColorSetup{
  link           = purple,
  chapter-rule   = black,
  axiom          = purple,
  definition      = blue
}
```

本文档类的默认配色如下:













结构元素	chapter	chapter-rule	link	url	cite		
颜色							
数学环境	axiom	definition	theorem	lemma	corollary	proposition	remark
颜色							

表 1.3: zL^AT_EX 文档类默认配色

1.4.5 zlatexFramed

目前本模板还提供了命令`\zlatexFramed {<name>}[<color>]`用于创建类似 Markdown 的引用环境. 参数中 `<name>` 表示声明环境名称³, `<color>` 表示此环境的背景颜色. 一个简单的使用样例如下:

```
% 环境 'refer' 声明
\zlatexFramed{refer}[orange]
% 使用环境 'refer'
\begin{refer}%
% something wrting here
\end{refer}
```

As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

劳仑衣普桑, 认至将指点效则机, 最你更枝. 想极整月正进好志次回总般, 段然取向使张规军证回, 世市总李率英茄持伴.

在上面的 refer 环境开始时插入一个% 可以用于消除多余空格

1.5 zL^AT_EX 接口

zL^AT_EX 的接口正在不断的完善中, 所以目前的接口可能并不是那么稳定 (我已然尽力).

³如果此环境已存在, 那么该环境会被 Override

1.5.1 命令声明

后续可能会考虑声明一个用于定义键值对命令的接口，而不是默认的位置参数或者是 `xparse` 提供的可选参数等。依托于 \LaTeX 3 的 `key-value` 模块，这样的接口会更加的灵活，更加的方便用户使用。

1.5.2 盒子接口

通过 \LaTeX 3 的盒子接口进行接口的定制可能并不是后续的一个模板开发重点。如果要开发，也是主要基于宏包 `xcoffins` 已提供命令的再次封装，在万不得已时我不会做这部分的底层工作。一般情况下有了 `xcoffins` 宏包后便可以进行大部分的操作了，如果使用者还需要绘制部分复杂的图形只能够借用 `tikz` 宏包，不妨先在另一个文档类中生成对应的 pdf 文件，然后在正文中直接插入 pdf 即可。目前 `xcoffins` 宏包的接口已经稳定，详细信息请参见

- [l3experimental](#)
- [interface of xcoffins](#)

一般盒子操作对于 \LaTeX 的普通使用者来说可能难度比较大，所以要提供一个便于用户使用，同时还要保证接口的统一性的系列命令是不容易的，所以 $\mathrm{z}\LaTeX$ 盒子接口还没有开始编写。其实如果用户能够完成盒子的声明，盒子 (join box) 简单拼接等操作，那么这就已经足够了。更进一步，若用户能够对盒子进行一些常规的操作：包括盒子的 `resize`, `scale`, `rotate`。那么大部分的复杂操作你都能够自己实现了 (除去少部分必须使用 `tikz`)。

1.5.3 绘图接口

本文档类在将来并不打算引入 `tikz` 宏包，一方面个人觉得 `tikz` 宏包过于庞大，很多功能并不都是一个文档类所应必须拥有的。也许会在后续引入 `l3draw` 模块用于替代部分的 `tikz` 命令。

因为本文档类可能会抛弃 `tikz`, `tcolorbox` 等宏包，所以在后续版本中可能需要加入 `xcoffins`, `l3draw`, `l3opacity` 等隶属于 `l3experimental` 的宏包。基于这些宏包定制属于 $\mathrm{z}\LaTeX$ 的接口，从而提供对应的用户接口，完善图形绘制功能等。这些宏包大部分还处于 `experiment` 阶段，自然 $\mathrm{z}\LaTeX$ 的这部分接口自然也可能在不久的将来改变，请见谅。同样的，`l3draw` 其实还不能完全实现 PGF 的功能，如 `shade` 操作。

本文档类配套的 `zTikZ` 模块 (宏包) 提供了丰富功能: 如 `TikZ` 绘图, 数值计算, 符号计算以及部分的图像处理功能. 具体使用请参见 `ztikz_manual.pdf`.

1.5.4 算法与数据结构

在 \LaTeX 3 中其实已经提供了很多的常用数据结构: 列表 (`token list`, `sequence`, `clsit`), 字典 (`prop list`), 整数数组 (`int array`), 浮点数组 (`fp array`). 其实其中也提供了基本的堆, 栈, 队列等常用数据结构. 使用已有的这些元素来构造如: 图, 树等结构也不是特别困难的。

在 L^AT_EX3 中实现常用的算法可以么？当然可以，而且有了 L^AT_EX3 的宏展开机制，相较于在原本的 L^AT_EX2e 中实现，在现阶段要简单很多。但是可能需要稍微深入理解 L^AT_EX3 的宏展开机制。所以最好还是从命令 `\expandafter` 以及 T_EX 的 token 处理上手吧 >_<。其实也不用深入到 T_EX 的 token 处理。

1.5.5 计数器

当用户在 zL^AT_EX 中加载不同的文档类，如 `article`, `book`, `ctexbook` 后，本文档类自动继承了这一部分计数器。至于定义环境部分，则是使用 `amsthm` 宏包所定义的数学环境计数器，如 `theorem`, `definition`, `corollary`, `example`, `axiom`, `remark` 等。用户在使用过程中可以按照自己的需要重定义这部分计数器。

zL^AT_EX 提供命令 `\zlatexUpdateCounterAfter` 用于设置计数器的更新，使用格式为：

```
\zlatexUpdateCounterAfter{<child>}{<father>}
```

也就是让上述的 `<child>` 计数器随着 `<father>` 父计数器的更新而更新，本命令原始声明：

```
\NewDocumentCommand{\zlatexUpdateCounterAfter}{mm}{
  \@addtoreset{#1}{#2}
}
```

本文档类的公式计数器默认跟随 `section` 计数器进行更新，在 zL^AT_EX 中源码声明为：

```
\counterwithin{equation}{section}
```

1.5.6 接口完善

目前 zL^AT_EX 接口还没有稳定，尤其是其中的 Hook(钩子) 声明部分还没有完全的处理妥当。但是一般情况下，用户需要配置的选项是比较少的，只要能够把导言区设置规范，那么剩下的内容就交给 zL^AT_EX 罢。

1.6 自定义

1.6.1 封面

本文档类并没有内建复杂的封面格式，只是简单的重定义了 `\maketitle` 命令用于生成封面。声明如下：

```
\renewcommand{\maketitle}{
  \begin{titlepage}
```

```

\vfll\vspace*{40pt}
\noindent\hspace*{134pt}\rule[-75pt]{6pt}{95pt}{\hspace*{10pt}\leavevmode\parbox
[t]{25em}{\fontsize{25}{25}\selectfont\bfseries\@title}}\par
\vspace*{-15pt}
\noindent\hspace*{150pt}{\leavevmode\parbox[t]{20em}{\Large\bfseries\@author}}\p
ar
\vfll
\noindent\hspace*{150pt}{\Large\textcolor{gray}{\@date}}
\end{titlepage}}

```

如果使用者想要使用更加美观的封面,请手动加载 `tikz` 宏包,然后重定义上述的`\maketitle`命令,以此实现自己的定制目标.

1.6.2 目录

➡ 自定义 toc

尽管在前面的“`zLATEX` 的加载选项”一节便已经说明了 `zLATEX` 文档类的部分页面元素格式. 但 `zLATEX` 提供的加载选项 `<toc>` 以及的其可选值 `redef`, `2column` 其实并不是由 `titletoc` 宏包提供. 但是用户仍然可以将其用于目录的格式定制,只是在本文档类中并没有采用其提供的命令而已.

所以,如果用户想定制目录格式,那么可使用默认加载的 `titlesec` 宏包或自行加载 `tikz` 宏包对目录进行定制. 这里给出基于第一种方案下的一个定制样例,效果可见图 1.3:

```

% \RequirePackage{zhnumber}
\titlecontents*{chapter}[0pc]{\large\bfseries}
  {\textcolor{\tl_use:N \l__zlatex_link_color_tl}{\thecontentslabel}}{}
  {\normalfont\titlestyle*[1pc]{.}\contentspage}[]
\titlecontents{section}[1.8pc]{\addvspace{3pt}\tl_use:N\g__sec_symbol_tl\,}
  {\textcolor{\tl_use:N \l__zlatex_link_color_tl}{\S\;\thecontentslabel}}{}
  {\titlestyle*[1pc]{.}\contentspage}%[{\tocsym[sec]{}]}
\titlecontents{subsection}[3pc]{\addvspace{3pt}\tl_use:N\g__subsec_symbol_tl\,}
  {\textcolor{\tl_use:N \l__zlatex_link_color_tl}{\thecontentslabel}}{}
  {\titlestyle*[1pc]{.}\contentspage}%[{\tocsym[subsec]{}]}
\titlecontents*{subsubsection}[5.8pc]{\small}
  {\textcolor{\tl_use:N \l__zlatex_link_color_tl}{\thecontentslabel}}{}
  {(\thecontentspage)}[\quad]

% append of prepend item to toc
% switch left indent for subsection (none->3pc)
\tl_new:N \g__sec_symbol_tl
\tl_new:N \g__subsec_symbol_tl
\tl_set:Nn \g__sec_symbol_tl {}
\tl_set:Nn \g__subsec_symbol_tl {}
\NewDocumentCommand\tocsym{0{sec}0{}}{

```

```

\str_case:nnF {#2}{
  {star}{ \tl_set:cn {g__#1_symbol_tl}{\ding{73}} }
  {=}{ \tl_set:cn {g__#1_symbol_tl}{\ding{104}} } %*
}{\tl_set:cn {g__#1_symbol_tl}{#2}}
}

```

需要注意的是,有时单独设置 toc 中的目录样式往往是不够的,可能还需要设置其计数器 (章节) 的格式,参见[小节 1.6.5](#).

1.6.3 子目录

在 z \LaTeX , 用户可以在 chapter/section 后插入本章/节下的所有子目录. 此功能基于 `titletoc` 宏包, 本文档类中将其封装为命令 `\partialToc[<depth>]`, 其中 `<depth>` 表示目录的深度, 默认为 2. 一个使用示例如下:

```

\chapter{Test}
\partialToc

```

当然你也可以把子目录放入边注中, 类似于 Memoir 文档类中提供的效果: 只需把 `\partialToc` 命令放入 `\marginpar` 环境即可.

注记 1.6.1 如果要在边注中放入子目录, 那么可以考虑加载 `margin` 选项, 否则 `margin` 过窄会导致子目录显示不协调, 甚至是显示不全.

1.6.4 页眉页脚

本文档类采用 `fancyhdr` 进行页眉页脚的定制, 如果用户想自定义页眉页脚样式, 可以直接使用 `fancyhdr` 提供的命令 `\fancyhead`, `\fancyfoot` 重定义这部分内容, 抑或是声明对应的页面样式 (pagestyle). 由于本文档类的 `<slide>` 选项的存在, 后续会考虑添加设置 slide 中页面元素的接口.

1.6.5 章节格式

z \LaTeX 目前没有提供定制章节格式的接口. 使用者可以使用宏包 `titlesec`, `tocloft` 提供的命令进行自定义. z \LaTeX 文档类默认加载了 `titlesec`, `titletoc` 宏包进用于 chapter 格式的定制以及输出子目录功能. 如果用户仍不满足于目前的章节格式, 请直接使用 `titlesec` 宏包的 `\titleformat` 命令覆盖本模板的原始定义, 或者加载其它宏包进行定制.

在前面我们已经提到可以使用 `titletoc` 宏包对目录进行定制, 这里给出与之相匹配的计数器设置:

```

% title format
\titleformat{\section}

```

```

{\bfseries\centering\Large}

{\thesection}

{0ex}{-}{-}

\titleformat{\subsection}

{\bfseries\large}

{\thesubsection}

{0ex}{-}{-}

\titleformat{\subsubsection}

{\bfseries}

{\thesubsubsection}

{0ex}{-}{-}

% (sub)section number styl
\setcounter{secnumdepth}{5}

\renewcommand{\thesection}{\thechapter.\arabic{section}\;}

\renewcommand{\thesubsection}{\zhnum{subsection}.\hspace*{-1ex}}

\renewcommand{\thesubsubsection}{\alph{subsubsection}.\;}

```

最终可得到如下样式目录:

目录			
第一章 一元函数微分	1		
§ 1.1 预备	1		
一、几个记号	1		
二、几个微分的不等式	4		
三、Weier 公式	6		
四、中值定理与上、下导数	7		
§ 1.2 用定积分证明微分中值定理	9		
一、用定积分证明微分中值定理	9		
二、用 Cauchy 定理证明微分中值定理	14		
三、洛必达法则及“0/0”和“∞/∞”的极限	15		
四、利用中值定理证明微分中值定理	17		
五、微分中值定理、微分中值定理与微分中值定理	18		
六、微分中值定理	19		
七、中值定理 1.2	20		
§ 1.3 泰勒公式与泰勒展开式	22		
一、泰勒公式与泰勒展开式	22		
二、泰勒公式与泰勒展开式	24		
三、泰勒公式与泰勒展开式	25		
四、泰勒公式与泰勒展开式	26		
五、泰勒公式与泰勒展开式	28		
六、泰勒公式与泰勒展开式	30		
§ 1.4 多元函数微分	30		
一、多元函数微分	30		
二、多元函数微分	30		
三、多元函数微分	30		
四、多元函数微分	30		
五、多元函数微分	30		
六、多元函数微分	30		
七、多元函数微分	30		
八、多元函数微分	30		
九、多元函数微分	30		
十、多元函数微分	30		
十一、多元函数微分	30		
十二、多元函数微分	30		
十三、多元函数微分	30		
十四、多元函数微分	30		
十五、多元函数微分	30		
十六、多元函数微分	30		
十七、多元函数微分	30		
十八、多元函数微分	30		
十九、多元函数微分	30		
二十、多元函数微分	30		
二十一、多元函数微分	30		
二十二、多元函数微分	30		
二十三、多元函数微分	30		
二十四、多元函数微分	30		
二十五、多元函数微分	30		
二十六、多元函数微分	30		
二十七、多元函数微分	30		
二十八、多元函数微分	30		
二十九、多元函数微分	30		
三十、多元函数微分	30		
三十一、多元函数微分	30		
三十二、多元函数微分	30		
三十三、多元函数微分	30		
三十四、多元函数微分	30		
三十五、多元函数微分	30		
三十六、多元函数微分	30		
三十七、多元函数微分	30		
三十八、多元函数微分	30		
三十九、多元函数微分	30		
四十、多元函数微分	30		
四十一、多元函数微分	30		
四十二、多元函数微分	30		
四十三、多元函数微分	30		
四十四、多元函数微分	30		
四十五、多元函数微分	30		
四十六、多元函数微分	30		
四十七、多元函数微分	30		
四十八、多元函数微分	30		
四十九、多元函数微分	30		
五十、多元函数微分	30		
五十一、多元函数微分	30		
五十二、多元函数微分	30		
五十三、多元函数微分	30		
五十四、多元函数微分	30		
五十五、多元函数微分	30		
五十六、多元函数微分	30		
五十七、多元函数微分	30		
五十八、多元函数微分	30		
五十九、多元函数微分	30		
六十、多元函数微分	30		
六十一、多元函数微分	30		
六十二、多元函数微分	30		
六十三、多元函数微分	30		
六十四、多元函数微分	30		
六十五、多元函数微分	30		
六十六、多元函数微分	30		
六十七、多元函数微分	30		
六十八、多元函数微分	30		
六十九、多元函数微分	30		
七十、多元函数微分	30		
七十一、多元函数微分	30		
七十二、多元函数微分	30		
七十三、多元函数微分	30		
七十四、多元函数微分	30		
七十五、多元函数微分	30		
七十六、多元函数微分	30		
七十七、多元函数微分	30		
七十八、多元函数微分	30		
七十九、多元函数微分	30		
八十、多元函数微分	30		
八十一、多元函数微分	30		
八十二、多元函数微分	30		
八十三、多元函数微分	30		
八十四、多元函数微分	30		
八十五、多元函数微分	30		
八十六、多元函数微分	30		
八十七、多元函数微分	30		
八十八、多元函数微分	30		
八十九、多元函数微分	30		
九十、多元函数微分	30		
九十一、多元函数微分	30		
九十二、多元函数微分	30		
九十三、多元函数微分	30		
九十四、多元函数微分	30		
九十五、多元函数微分	30		
九十六、多元函数微分	30		
九十七、多元函数微分	30		
九十八、多元函数微分	30		
九十九、多元函数微分	30		
一百、多元函数微分	30		

图 1.3: ToC Example

本文档类默认不加载 tikz, pgf 宏包, 想要使用这两个宏包定义更加复杂章节样式, 请手动加载, 并设置自己喜欢的章节格式. 也许后续我会在 \LaTeX 的加载选项中添加一个 tikz 宏包选项, 从而可以让用户自定义章节格式.

1.7 字体配置

1.7.1 一些术语

在字体上，国内环境可以说是一团浆糊，再加之 M\$ word 对用户的不断误导与溺爱，导致无数的用户不知道字体，字体族，字体系列，字形的区别。下面给出一些常见注意事项：

- 可以粗略的认为一个字体中包含了一些字体族，一些字体系列。做字体是一个吃力不讨好的活，没多少人愿意做。
- 一般这个字形是对于西文字体而言的，对于中文无效。中文所谓的斜体概念都是 word 这个毒瘤产生的。部分字体本来就不是给中文设计的，所以不要想着把它们应用在中文上。
- 深受 M\$ Office 迫害的用户很多喜欢 Times New Roman 字体，但是它并没有对应的数学字体 (`newtxmath` 不是)。
- **注意字体版权：** Windows 默认的字体全部都是有商业版权的。原则上，只有正版 Windows 用户方才可以用于非商业用途。如需用于商业，必须获得授权

1.7.2 New Computer Modern

目前在 XeTeX 下已有一个比较完善的字体包 `newcomputermodern`，这个包兼容原始的 CM 字体，并且对部分的细节进行了优化。最重要的是，它支持多种语言：希腊语，俄语，中文，日文，韩文，西里尔语和梵文等。这个包的使用方法如下：

```
% way 1
\usepackage{newcomputermodern}

% way 2
\usepackage[default]{fontsetup}
```

推荐使用第二种方式进行调用，具体的使用细节请参见：[New Computer Modern](#)。

1.7.3 模板预设

字体配置一直以来都是一个比较头疼的问题，尤其是对于中文文档。z^hTeX 目前具有两套字体设置：英文和中文。英文 (`lang=en`) 下的字体配置为：

```
\RequirePackage[utf8]{inputenc}
\RequirePackage[T1]{fontenc}
\RequirePackage[english]{babel}
```

在中文 (`lang=cn`) 环境下的字体配置为：

在设置全文的中(日韩)字体时也请使用命令`\setCJKmainfont` 命令, 格式和上述的命令一样. 也许在设置文档的中文字体时你还需要在导言区加载如下命令:

```
\renewcommand\CJKrmdefault{}
```

这几个命令可以用于设置整个文档的字体, 但是同样的是, 我们可以在一个文档中使用多个字体. 在讨论这个的时候, 我们遇到了一个问题? 怎么查看自己电脑上有哪一些可用的字体(包括数学字体)? 在 Windows 和 Linux 均可以通过如下命令进行查看:

```
fc-list
```

然后你就会看到如下的类似输出, 在 Linux 下可能是这样的:

```
/usr/share/fonts/100dpi/lubI24.pcf.gz: B&H LucidaBright:style=Italic
/usr/share/fonts/100dpi/UTBI__12-IS08859-1.pcf.gz: Adobe Utopia:style=Bold Italic
/usr/share/fonts/75dpi/luIS08-IS08859-1.pcf.gz: B&H Lucida:style=Sans Italic
/usr/share/fonts/75dpi/luIS10-IS08859-1.pcf.gz: B&H Lucida:style=Sans Italic
/usr/share/fonts/75dpi/luIS14-IS08859-1.pcf.gz: B&H Lucida:style=Sans Italic
/usr/share/fonts/100dpi/luRS18-IS08859-1.pcf.gz: B&H Lucida:style=Sans
/usr/share/fonts/75dpi/courB18-IS08859-1.pcf.gz: Adobe Courier:style=Bold
/usr/share/fonts/100dpi/lubBI24.pcf.gz: B&H LucidaBright:style=Italic
/usr/share/fonts/100dpi/courO12.pcf.gz: Adobe Courier:style=Oblique
```

在 windows 上可能是这样的(如果安装了 T_EXLive):

```
C:/WINDOWS/fonts/times.ttf: Times New Roman:style=Regular,Normal, ...
C:/WINDOWS/fonts/GeoSlab703 Md BT Bold.ttf: GeoSlab703 Md BT:style=Bold
C:/WINDOWS/fonts/vgasysr.fon: System:style=Regular
C:/WINDOWS/fonts/seguibl.ttf: Segoe UI, Segoe UI Black:style=Black,Regular
C:/texlive/2024/texmf-dist/fonts/opentype/public/drm/drmsl11.otf: drmsl11:style=Regular
C:/texlive/2024/texmf-dist/fonts/opentype/public/fonts-tlwg/Garuda-Oblique.otf:
Garuda:style=Oblique
C:/texlive/2024/texmf-dist/fonts/truetype/public/junicodevf/JunicodeVF-Roman.ttf:
Junicode VF:style=Exp Bold
C:/texlive/2024/texmf-dist/fonts/opentype/public/qualitytype/QTTehtone-BoldItalic.otf:
QTTehtone:style=BoldItalic
C:/texlive/2024/texmf-dist/fonts/opentype/public/tempora/Tempora-Italic.otf:
Tempora:style=Italic
C:/texlive/2024/texmf-dist/fonts/opentype/public/drm/drmsym11.otf:
drmsym11:style=Regular
```

我们可以通过“font name”和“file name”两种形式来调用字体. 比如我们想要调用“Times New Roman”字体, 在第一个输出示例中, 可以看到“Times New Roman”对应的文件名为“times.ttf”, 那么我们可以通过如下命令进行调用:


```
% by file name
\setmainfont{times.ttf}

% by font name
\setmainfont{Times New Roman}
```

如果你不想在你的本地安装过多的字体，我常常建议使用“file name”的格式。只需要把对应的字体文件放到你的项目文件夹下就行了，免去了字体安装这一个步骤。这个时候你就需要在可选参数中指定键 `<Path>` 对应的路径值，对应的是项目下的字体路径。这里给出一个示例：假如你把字体放到了项目根路径下的 `Fonts` 文件夹，那么声明格式为：

```
\setmainfont[Path = Fonts/]{<font name or file name>}
```

在安装字体后务必使用如下命令刷新系统字体缓存，否则 `fontspec` 或 `xeCJK` 宏包无法找到系统中对应的字体：

```
fc-cache -f -v
```

通过上面的两条命令我们便可以轻松的设置文档中 `\textrm{<content>}` 对应的字体⁴ 为“Times New Roman”。

1.7.5 读懂字体警告

但是你可能也会遇到如下的编译警告：

```
LaTeX Font: Font shape `TU/AlegreyaSans-ExtraBoldItalic.otf(0)/b/n' undefined
(Font)^^Iusing `TU/AlegreyaSans-ExtraBoldItalic.otf(0)/m/n' instead.
```

上述的字体是我在导言区设置了如下字体命令导致的：

```
\setmainfont{AlegreyaSans-ExtraBoldItalic.otf}
```

为什么会有这个警告？其实就是出在一个字体加粗命令 `\textbf` 上，我并没有指定这个新的文章中字体族 (rmfamily) 所对应的粗体系列字体。所以 $\text{T}_{\text{E}}\text{X}$ 采用了默认的“m(edium)”，而不是“b(old)”。详细信息可以参见 [3]：The $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ Companion(3rd) – Font Selection and Encodings。这里给出其中两个关于关于字体设置的表格：

⁴默认情况下，文档中的字体族即为 roman family

Family	Series	Shapes	Typeface Examples
<i>Computer Modern Roman</i> (Encodings: T1, TS1, OT1)			
cmr	m	n, it, sl, sc, ui	Computer Modern Roman, <i>italic</i> , <i>slanted</i> , and SMALL CAPS
	bx	n, it, sl, sc	bold extended , <i>italic</i> , <i>slanted</i> , and SMALL CAPS
	b	n	bold medium width (no other shapes)
<i>Computer Modern Sans</i> (Encodings: T1, TS1, OT1)			
cmss	m, bx	n, (it), sl	Computer Modern Sans bold extended , <i>italic</i> , and <i>slanted</i>
	sbc	n	Computer Modern Sans semibold condensed (no shape variants)
<i>Computer Modern Typewriter</i> (Encodings: T1, TS1, OT1)			
cmtt	m	n, it, sl, sc	Comp. Mod. Typewriter, <i>italic</i> , <i>slanted</i> and SMALL CAPS
cmvtt	m	n , it	Comp. Mod. Typewriter proportional upright and italics
<i>Computer Modern Fibonacci</i> (Encodings: T1, OT1)			
cmfib	m	n	Computer Modern Fibonacci
<i>Computer Modern Dunhill</i> (Encodings: T1, OT1)			
cmdh	m	n	Computer Modern Dunhill

Values in **blue** are not or only partly offered in OT1 encoding!

Table 9.5: Classification of the Computer Modern font families

It is followed by a discussion of \LaTeX 's standard support packages for input and font encodings including a discussion of how to use an extended set of text symbols. The section concludes by describing a package for tracing \LaTeX 's font processing and another package for displaying glyph charts (a package the author used extensively while preparing the later parts of this chapter).

9.5.1 Computer Modern, Latin Modern — The \LaTeX standard fonts

Original \TeX font encoding

Along with \TeX , Donald Knuth developed a family of fonts called Computer Modern; see Table 9.5. Until the early 1990s, essentially only these fonts were usable with \TeX and, consequently, with \LaTeX . Each of them contains 128 glyphs (\TeX was working with 7 bits originally), which does not leave room for including accented characters as individual glyphs. Thus, using these fonts means that accented characters have to be produced with the `\accent` primitive of \TeX , which in turn means that automatic hyphenation of words with accented characters is impossible. While this restriction is acceptable for English documents that contain few foreign words, it is a major obstacle for other languages.

Not surprisingly, these deficiencies were of great concern to the \TeX users in Europe and eventually led to a reimplementaion of \TeX in 1989 to support 8-bit

Family	Series	Shapes	Typeface Examples
<i>Latin Modern Roman</i>		Latin Modern Roman Latin Modern Roman Caps	(Encodings: T1, TS1, OT1, T5, LY1, and TU)
lmr	m	n, it, sl, sc, scl , ui	Latin Modern Roman upright, <i>italic</i> , <i>slanted</i> , SMALL CAPS and SMALL CAPS SLANTED
	bx	n, it, sl	Latin Modern bold extend, <i>italic</i>, and <i>slanted</i>
	b	n, (it) , sl	Latin Modern bold <i>and slanted</i>
<i>Latin Modern Sans</i>		Latin Modern Sans	(Encodings: T1, TS1, OT1, T5, LY1, and TU)
lmss	m	n, (it) , sl	Latin Modern Sans upright <i>and slanted</i>
	(b) , bx, sbc	n, (it) , sl	Latin Modern Sans bold extended <i>and slanted</i>, and semibold condensed <i>and slanted</i>
<i>Latin Modern Typewriter</i>		Latin Modern Mono	(Encodings: T1, TS1, OT1, T5, LY1, and TU)
lmtt	m	n, it, sl, sc, scl	Latin Modern Typewriter upright, <i>italic</i> , <i>slanted</i> , SMALL CAPS and SMALL CAPS SLANTED
	l , lc , b , (bx)	n, (it) , sl	Latin Modern Typewriter light, <i>slanted</i>, and light condensed, <i>slanted</i>, bold and bold <i>slanted</i>
<i>Proportional Latin Modern Typewriter</i>		Latin Modern Mono Prop	(Encodings: T1, TS1, OT1, T5, LY1, and TU)
lmttt	m	n, (it) , sl	Latin Modern Typewriter proportional <i>and slanted</i>
	l , b , (bx)	n, (it) , sl	Latin Modern Typewriter light, light <i>slanted</i>, bold and bold <i>slanted</i>
<i>Latin Modern Dunhill</i>		Latin Modern Roman Dunhill	(Encodings: T1, TS1, OT1, T5, LY1, and TU)
cmdh	m	n, (it) , sl	Latin Modern Dunhill upright <i>and slanted</i>

Shapes or series value colored in blue are additions in the Latin Modern family not available in the Computer Modern. Values given in parentheses indicate substitutions; e.g., *(it)* produces **sl**, and **(b)** becomes **bx**, or vice versa. Note that the CM proportional typewriter has an **it** shape while the LM proportional typewriter has an **sl** shape instead. The font names with gray background are for use with the fontspec package in Unicode engines.

Table 9.6: Classification of the Latin Modern font families

additional shapes and weights were added [171]. In particular, there are now bold and light versions of the typewriter fonts that can be useful in certain situations.

However, without accompanying math fonts, they were not yet a full-fledged alternative to Computer Modern or European Modern fonts. This changed in 2011 with the release of the Latin Modern Math fonts. Additionally, all fonts in the Latin Modern families were made available in OpenType font format, which made them available to be used in Unicode T_EX engines (or even Word, if so desired).

Unicode engines

In fact, they are now L^AT_EX's default fonts if you typeset using X_YL^AT_EX or LuaT_EX.

在阅读了上述关于“字体的选择”内容之后, 解决这个问题就不难了, 给出两个方法:

- 通过“font name”进行设置的情况下, 如果你的本地字库够全, 那么 \TeX 会自动处理这个问题.
- 通过“file name”进行设置的情况下, 你可以通过如下命令进行设置, 这里以 `comic.ttf` 为例:

```
\setmainfont{comic.ttf}[
  BoldFont=comicbd.ttf,
  ItalicFont=comici.ttf,
  BoldItalicFont=comicz.ttf
]
```

如果你在 Windows 命令行使用如下命令:

```
fc-list | rg comic | rg WINDOWS
```

那么可以得到如下的输出:

```
C:/WINDOWS/fonts/comicbd.ttf: Comic Sans MS:style=Bold
C:/WINDOWS/fonts/comici.ttf: Comic Sans MS:style=Italic
C:/WINDOWS/fonts/comicz.ttf: Comic Sans MS:style=Bold Italic
```

说明在本地的名为“Comic Sans MS”的字体族具有“Bold, Italic, Bold Italic”三种字形. 通过下面的命令我们便可以简单的通过第一种方式解决这个字体加粗问题:

```
\setmainfont{Comic Sans MS}
```

就像上述备注的第二种“file name”方式的好处外, By “font name”也有上述的优点. 尽管对于字体的加粗和斜体你可以偷懒, 在字体族声明的可选参数中加入 `AutoFakeBold`, `AutoFakeSlant` 参数, 它会自动同时实现中英文伪粗体和伪斜体, 不用你自己单独去指定. 但是我并不推荐. 还是给出一个示例:

```
\setCJKfamilyfont{AutoFakeBoldSlant}{<font-name>.ttf}[AutoFakeBold , AutoFakeSlant]
```

在通过“font name”进行字体的声明时, 可以考虑把字体放到如下的位置, 这样可以避免书写字体的路径:

- MacOS: `~/Library/Fonts`
- Windows: `C:\Windows\Fonts`
- Linux 或 Windows: 放在 TEXMF tree 对应的路径, 见前文.

1.7.6 字体族声明

上述的三个命令都是设置全文对应的字体族, 如果你想要改变局部的字体族, 那么可以使用 `fontspec` 宏包提供的命令 `\newfontfamily` 和 XeCJK 宏包提供的 `\setCJKfamilyfont` 分别进行英文与中日韩文字的字体族声明. 二者的声明方式为:

```
\newfontfamily{\FamilyA}{FontA.ttf}
\setCJKfamilyfont{FamilyB}{FontB.ttf}
```

上述两条命令声明了 `FamilyA`, `FamilyB` 两个字体族, 分别对应了 `FontA.ttf`, `FontB.ttf` 两个字体文件. 想要使用这两个字体族, 在文档中按照如下方式使用:

```
{\FamilyA <Your Content>}
{\CJKfamily{FamilyB} < 你的内容 >}
```

比如你需要在文档中输入俄语, 由于 `article` 文档类默认的字体 `Computer modern` 并没有包含这些俄语字母字形 (Glyph). 所以你可能就需要使用一个含有这类字形的字体了, 推荐 `CMU Serif`. 可以安装进系统也可以放在项目文件夹下, 这里放在项目文件夹 `./Fonts/` 下为例:

```
\newfontfamily{\russia}[Path=./Fonts/]{cmunrm.ttf}[
  BoldFont=cmunbx.ttf,
  ItalicFont=cmunbi.ttf
]
```

然后使用如下格式调用:

```
{\russia < 俄语内容 >}
```

Жизнь, как прекрасная мелодия, только текст перепутались.
生命是首美丽的曲子, 只是歌词有些纠结

当然了, 你也可以通过 `\setmainfont` 命令, 从而进行全局字体设置. 最后还是给出声明中文字体族的一个简单示例:

```
\setCJKfamilyfont{fangsong}{simfang.ttf}[
  Path=./Fonts/,
  BoldFont=simfangbold.ttf
]
```

在声明了对应的字体族之后, 使用如下格式进行调用:

```
{\CJKfamily{fangsong} < 你的中文内容 >}
```

1.7.7 Emoji

如果用户想要使用 Emoji, 在阅读完上述的字体配置后, 相信读者已经能够独自解决此问题了. 这里给出大致的解决思路:

- 把对应的 Emoji 对应的 unicode 放到一个支持显示表情包的字体组环境中.
- 加载 `fontawesome5` 宏包.
- 更换编译引擎为 Lua \TeX .

1.7.8 数学字体

目前 \LaTeX 文档类提供了几种常见的数学字体宏包接口, 分别是 `computer modern math`, `newtxmath`, `eulervm`, `mtpro2`. 其中 `computer modern math` 即为默认的数学字体, 推荐刚上手的用户使用. 这里需要说明: 数学字体并不仅仅只包含我们直接看到的那部分 Glyph 本身, 一套数学字体往往还需要对应的一个 kern tables(距离表), 用于指定各个数学符号之间的距离. 数学字体还包含很多复杂的东西, 并没有我们认为的那么简单.

不想要折腾的用户可以直接看 [Free-Math-Font](#) 这个文档, 其中提供了多种数学字体的调用方法. 即使是你配置不来, 但是至少你能够直接参考文档中的案例实现自己的需求. 但这里不去介绍上述对应宏包的使用方法, 下面重点介绍宏包 `unicode-math`.

`unicode-math` 宏包的基本使用格式为:

```
% after \usepackage{fontspec}
\usepackage{unicode-math}
% set main math font (necessary)
\setmathfont{Latin Modern Math}
```

上述的命令 `\setmathfont` 由宏包 `unicode-math` 提供, 用于设置文档中数学字体. 在指定文档的数学字体后, 你可以再声明一系列的数学字体命令用于指定局部数学字体样式. 但是在这之前, 请允许我先介绍一下怎么查看你本地系统中的数学字体.

以 Windows 为例, 在命令行运行:

```
fc-list | rg Math
```

然后你可以得到大致如下的结果:

```
C:/WINDOWS/fonts/eumat2.ttf: Euclid Math Two:style=Regular
C:/WINDOWS/fonts/eumat2b.ttf: Euclid Math Two:style=Bold
C:/WINDOWS/fonts/cambria.ttc: Cambria Math:style=Regular
```

```
C:/texlive/2024/texmf-dist/fonts/opentype/public/xits/XITSMath-Bold.otf: XITS
Math:style=Bold
C:/texlive/2024/texmf-dist/fonts/opentype/public/xits/XITSMath-Regular.otf: XITS
Math:style=Regular
C:/texlive/2024/texmf-dist/fonts/opentype/public/tex-gyre-math/texgyretermes-math.otf:
TeX Gyre Termes Math:style=Regular
```

和前文的设置正文字体一样，可以通过“font name”或者是“file name”进行数学字体的设置，所以在这里不再赘述。这里介绍一下怎么设置局部数学字体。通过如下命令：

```
\setmathfont[range={\mathscr,\mathbfscr}]{TeX Gyre Termes Math}
```

在此命令之后的所有`\mathscr`、`\mathbfscr` 命令均会使用“TeX Gyre Termes Math”字体。其他不变，比如`\mathrm`、`\mathbf` 等。后续如果还想使用其他数学字体，只需要再次声明不同的`\setmathfont` 命令即可。

值得注意的是：在加载宏包 `unicode-math` 后，数学字符的加粗请采用`\boldsymbol` 或`\symbf`、`\symbfit`、`\symbfup` 等命令。

关于 `unicode-math` 宏包的更多功能，请参见对应的宏包手册。

1.7.9 常见字体问题

在阅读了上述字体配置的基础知识介绍后，你应该能看懂大部分模板中的字体配置了。在看懂这些命令后，也能够说明一些常见字体问题背后原因了：

- 为什么 Windows 下中文加粗变成了黑体？斜体变成了楷体？
- 为什么有时 L^AT_EX 会报字体缺失的警告或错误？
- 怎么在 Windows 上使用 Linux 下的字体？或者反过来使用？
- 怎么使用 Founder 字体？
- ...

在 Windows 下文字加粗后变黑体的问题⁵，如果你去看 `ctex-fontset-windows.def` 这个文件，有如下声明：

```
% 中文默认字体：宋体，粗体以黑体代替，斜体以楷书代替
\setCJKmainfont { SimSun } [ BoldFont = SimHei , ItalicFont = KaiTi ]
% 中文无衬线字体：微软雅黑，粗体为对应的微软雅黑粗体
\setCJKsansfont { Microsoft-YaHei } [ BoldFont = *~Bold ]
% 中文等宽字体：仿宋
```

⁵具体参考请见：[LaTeX 中文字体配置基础指南](#)

```

\setCJKmonofont { FangSong }
% 设置中文字族
\setCJKfamilyfont { zhsong } { SimSun }
\setCJKfamilyfont { zhhei } { SimHei }
\setCJKfamilyfont { zhfs } { FangSong }
\setCJKfamilyfont { zhkai } { KaiTi }
\setCJKfamilyfont { zhyahei } { Microsoft~YaHei } [ BoldFont = *-Bold ]
\setCJKfamilyfont { zhli } { LiSu }
\setCJKfamilyfont { zhyou } { YouYuan }
% 字体命令，可用于文档中自由设置字体
\NewDocumentCommand \songti { } { \CJKfamily { zhsong } }
\NewDocumentCommand \heiti { } { \CJKfamily { zhhei } }
\NewDocumentCommand \fangsong { } { \CJKfamily { zhfs } }
\NewDocumentCommand \kaishu { } { \CJKfamily { zhkai } }
\NewDocumentCommand \lishu { } { \CJKfamily { zhli } }
\NewDocumentCommand \youyuan { } { \CJKfamily { zhyou } }
\NewDocumentCommand \yahei { } { \CJKfamily { zhyahei } }

```

上面有一句 `BoldFont = SimHei`, 就表示`\textbf{< 文字 >}` 中的内容是“黑体”。但是在 Linux 下对应的设置为: `BoldFont = FandolSong-Bold`, 这就是真正的 `fandol` 字体中的粗体了。

比如 `ElegantLATEX` 系列中的 `Book` 文档类, 你能在其中看到如下的字体声明命令:

```

% 英文部分
\setmainfont{TeXGyreTermesX}[
  UprightFont = *-Regular ,
  BoldFont = *-Bold ,
  ItalicFont = *-Italic ,
  BoldItalicFont = *-BoldItalic ,
  Extension = .otf ,
  Scale = 1.0
]
\setsansfont{texgyreheros}[
  UprightFont = *-regular ,
  BoldFont = *-bold ,
  ItalicFont = *-italic ,
  BoldItalicFont = *-bolditalic ,
  Extension = .otf ,
  Scale = 0.9
]

% 中文部分
\RequirePackage[UTF8, scheme=plain, fontset=none]{ctex}
\setCJKmainfont[BoldFont={FZHei-B01}, ItalicFont={FZKai-Z03}]{FZShuSong-Z01}

```



```

\setCJKsansfont[BoldFont={FZHei-B01}]{FZKai-Z03}
\setCJKmonofont[BoldFont={FZHei-B01}]{FZFangSong-Z02}
\setCJKfamilyfont{zhSong}{FZShuSong-Z01}
\setCJKfamilyfont{zhHei}{FZHei-B01}
\setCJKfamilyfont{zhKai}[BoldFont={FZHei-B01}]{FZKai-Z03}
\setCJKfamilyfont{zhfs}[BoldFont={FZHei-B01}]{FZFangSong-Z02}
\newcommand*{\songti}{\CJKfamily{zhSong}}
\newcommand*{\heiti}{\CJKfamily{zhHei}}
\newcommand*{\kaishu}{\CJKfamily{zhKai}}
\newcommand*{\fangsong}{\CJKfamily{zhfs}}

```

所以 C_T_EX 还是给我们解决了很多的字体问题的, 如果你的文档中加载了 `ctex` 宏包, 不妨使用其提供的 `<fontset>` 选项进行字体的设置. 实在不行, 再考虑自行配置字体也不迟.

1.7.10 Furthermore

本节并没有介绍在 pdf_T_EX 下的字体配置方法, 因为如果你想在此引擎下使用字体, 那么你可能需要自己去做一些复杂的字体工作, 自己去做好对应的 `tfm`, `map`, `enc`, `vf` 文件, 这并不简单.

还有一些 FallBack 字体 (主字体文件中不存在字符的备用字体) 的问题? 比如在 Windows 下可以使用其自带的一个超大字符集 `SimSun-ExtB`, 设置方法如下:

```

\XeCJKsetup{AutoFallBack=true}
\setCJKmainfont{SimSun}[FallBack=SimSun-ExtB]

```

什么, 你还想了解 PDF 的知识? 或者是虚拟字体相关的工作? 这些都不是我们一般人能够去碰的, 如果你想要了解更多字体相关的知识, 不妨就局限在宏包调用的水平, 可以多看看 `fontspec`, `ctex` 相关的文档, 或者是看看 *The L^AT_EX Companion* 这本书中的内容.

后续可能就是一些 OpenTrueType 字体或者是 Unicode 的坑了, 即使是换了 Lua_T_EX 这些坑你还是得踩的, 还不一定出的来. 关于字体有一些好用的工具: `MetaFont`, `FontForge`, `Glyphs`. 这些工具可以用于查看字体或者设计字体.

1.8 数学环境

1.8.1 常用数学环境

本文档类使用宏包 `amsthm` 定义了主要的定理类环境, 大致分为两类: 定理类环境和证明类环境; 默认情况下定理类环境和证明类环境相同. 具体的环境名称见下方:

- 定理类环境
 - `axiom`
 - `definition`

- theorem
- lemma
- corollary
- proposition
- remark
- 证明类环境
 - proof
 - exercise
 - example
 - solution
 - problem

常用的数学环境本模板基本已经覆盖, 如果有其它需求, 可以自行添加. 但在添加之前请先仔细阅读本文档, 确保没有冲突之处. 后续可能会开发一个对应的接口用于这类数学环境的统一声明和管理.

1.8.2 使用方法

现在介绍怎么使用这些数学环境, 其实上述每一个环境的基本调用格式基本相同, 格式如下:

```
\begin{<theorem like env>}[<theorem name>]
你的定理内容就写在这个环境的内部.

your theorem writing here.
\end{<theorem like env>}
```

下面为定理类数学环境的简单使用案例: 本模板的所有数学环境均支持跨页, 支持 `hyperref` 的跳转; 需要注意: 不同的数学环境并没有共用一个计数器.

想要对定理类环境添加 `label` 的语法如下:

```
\begin{<theorem like env>}[<theorem name>]\label{thm:test}
你的定理内容就写在这个环境的内部.

your theorem writing here.
\end{<theorem like env>}
```

后续引用直接使用命令 `\cref{thm:test}`, 如引用刚才标记的定理, 结果为: [定理 \(1.8.1\)](#), 可以看到, 这个 `hyperref` 的跳转功能正常. 本文档类中 `\cref` 命令会自动根据计数器的类别和文档的语言选项决定具体的引用格式. 针对于图表的引用也是同理的, 你只需要把这一切都交给 `\cref` 即可. 相关的详细信息还请参见本文档后续部分:[节 1.9](#).

1.8.3 定理类环境样式

zL^AT_EX 中的预想定义有 3 中样式: `plain`, `leftbar`, `background`, `fancy`. 其中 `plain` 表示数学环境不加载任何的修饰; `leftbar` 本质上就是 `framed` 宏包提供的 `leftbar` 环境; `background` 表示给对应的段落加上色彩背景; `fancy` 表示数学环境加载 `leftbar` 的同时设置其背景颜色, 为对应色彩的 10%.

只需要用户在加载本文档类时指定 `env-style` 选项即可设置设定对应的定理类环境格式, 比如本手册的数学环境主题为 `leftbar`(默认样式为 `plain`):

```
\documentclass[
  mathSpec={envStyle=leftbar}
]{zlatex}
```

zL^AT_EX 没有加载 `tcolorbox` 宏包用于这部分“盒子”的设计, 仅使用一个简单的 `framed` 宏包. 毕竟对于一篇普通的笔记或者书籍来说, 你并不需要这些可能会分散你亦或者是这个材料潜在读者注意力的彩色高亮“盒子”.

► `plain` 样式

定理 1.8.1 (prime theorem) As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.1)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.2)$$

劳仑衣普桑, 认至将指点效则机, 最你更枝. 想极整月正进好志次回总般, 段然取向使张规军证回, 世市总李率英茄持伴.

定义 1.8.1 (prime definition) As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.3)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.4)$$

劳仑衣普桑, 认至将指点效则机, 最你更枝. 想极整月正进好志次回总般, 段然取向使张规军证回, 世市总李率英茄持伴.

► `leftbar` 样式

引理 1.8.1 (prime lemma) As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.5)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.6)$$

劳伦衣普桑，认至将指点效则机，最你更枝。想极整月正进好志次回总般，段然取向使张规军证回，世市总李率英茄持伴。

注记 1.8.1 (prime remark) As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.7)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.8)$$

劳伦衣普桑，认至将指点效则机，最你更枝。想极整月正进好志次回总般，段然取向使张规军证回，世市总李率英茄持伴。

► background 样式

引理 1.8.2 (prime lemma) As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.9)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.10)$$

劳伦衣普桑，认至将指点效则机，最你更枝。想极整月正进好志次回总般，段然取向使张规军证回，世市总李率英茄持伴。

注记 1.8.2 (prime remark) As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.11)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.12)$$

劳仑衣普桑，认至将指点效则机，最你更枝. 想极整月正进好志次回总般，段然取向使张规军证回，世市总李率英茄持伴.

► fancy 样式

公理 1.8.1 (prime axiom) As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.13)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.14)$$

劳仑衣普桑，认至将指点效则机，最你更枝. 想极整月正进好志次回总般，段然取向使张规军证回，世市总李率英茄持伴.

命题 1.8.1 (prime proposition) As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.15)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.16)$$

劳仑衣普桑，认至将指点效则机，最你更枝. 想极整月正进好志次回总般，段然取向使张规军证回，世市总李率英茄持伴.

1.8.4 证明类环境

证明类环境比较朴素，没有可选的那部分默认参数，使用方法见下：

```
\begin{<proof like env>}
  定理内容.
\end{<proof like env>}
```

证明: As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.17)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.18)$$

劳仑衣普桑，认至将指点效则机，最你更枝. 想极整月正进好志次回总般，段然取向使张规军证回，世市总李率英茄持伴. ■

示例: As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.19)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.20)$$

劳仑衣普桑，认至将指点效则机，最你更枝. 想极整月正进好志次回总般，段然取向使张规军证回，世市总李率英茄持伴.

用户可以自行定制 proof 环境对应的证明结束标志，但是需要注意：对应的标志命令须放入公式环境，如果你的结束标志只能用于公式环境时. 例如，把证明结束符从 ■ 替换为 □:

```
\renewcommand{\qedsymbol}{\ensuremath{\square}}
```

1.8.5 注意事项

默认的数学类环境字体均采用正体 `\upshape`，如果使用者不喜欢前者默认的“正体”字体样式，可以直接在数学类环境开始时使用字体命令 `\itshape` 对原有字体 shape 进行覆盖，示例如下：

```
\begin{theorem}[test theorem]\itshape
  你好，Hello world !
\end{theorem}
```

注记 1.8.3 *As any dedicated reader can clearly see, the Ideal of practical reason is a representation of, as far as I know, the things in themselves;*

$$\mathbf{v} \otimes \mathbf{w} = \mathbf{v} \otimes \mathbf{w} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} u^i v^j \quad (1.8.21)$$

$$= \sum_{i=1}^3 (a_{i1} u^i v^1 + a_{i2} u^i v^2 + a_{i3} u^i v^3) \quad (1.8.22)$$

劳仑衣普桑，认至将指点效则机，最你更枝. 想极整月正进好志次回总般，段然取向使张规军证回，世市总李率英茄持伴.

同时，本文档类中数学类环境和前文的自定义高亮环境 `\zlatexFramed` 均默认首行不缩进，需手动添加缩进.

1.8.6 自定义数学环境

目前还没有开发对应的接口，主要基于目前环境已经基本够用的事实。

1.9 标签与引用

1.9.1 Footnote

可能部分用户并不喜欢默认脚注格式：脚注没有位于页面底部，而是在页脚偏上的位置；脚注符号无法自定义；为此用户可以独立加载宏包 `footmisc` 用于：强制脚注位于页面底部，或者定义脚注符号。但本文档类不打算默认加载此宏包，用户若需要，可自行在导言区添加如下命令：

```
% placement
\usepackage[bottom]{footmisc}
% symbol
\usepackage[symbol]{footmisc}
```

其实也可以使用 \LaTeX 2_ϵ 内置命令 `\fnsymbol` 进行脚注符号的自定义。

1.9.2 cleveref

$\text{z}\text{\LaTeX}$ 文档类使用 `cleveref` 宏包构建标签-引用系统。常规的 `\label{}` 操作并没有什么变化，区别主要在引用标签功能上。对于普通的模板你可能会看到如下的说明：使用 `\eqref` 进行公式标签的索引，使用 `\figref` 进行图片的索引，使用 `\tabref` 进行表格的索引... 但是在 $\text{z}\text{\LaTeX}$ 中，这一切都已经配置好，用户只需使用命令 `\cref` 即可。使用此命令可以避免用户在写作时书写如下格式引用命令：

```
定理:\ref{thm:test}
% or
\newcommand\thmref{定理:\ref{#1}}
```

在 $\text{z}\text{\LaTeX}$ 中，引用格式预设值如下（至于多个标签引用时，只有 `lang=en` 时采用部分变化，对应的前缀变为复数）：

语言	公式	图片	表格	part	chapter	section	subsection
<code>lang=en</code>	equation	figure	table	part	chapter	section	subsection
<code>lang=cn</code>	方程	图	表	部分	章	节	小节

表 1.4: cref 引用格式

对于 `cleveref` 中其它命令，如 `\Cref`, `\crefrange`, `\Crefrange` 等，本文档类未对其进行修改，故以上命令均兼容。但是详细的使用说明请参见 `cleveref` 宏包官方文档。

1.9.3 figure and table

zL^AT_EX 采用 cleveref 提供的 `\cref` 命令进行图片与列表的引用. 本文档类内置的 `\cref` 命令的用法和原宏包中 `\cref` 用法一致, 只是在引用时会根据文档的语言选项 (`<alang>`) 更改对应的 prefix. 比如在 `lang=cn` 时: 默认: “fig 1.1” 会变为中文环境下: “图 1.1”.

1.10 列表环境

zL^AT_EX 对 book 文档类的无编号计数器进行了定制, 有序列表和无序列表现在的具体样式如下:

- | | |
|--------|----------|
| • 一级项目 | 1. 一级项目 |
| ◦ 二级项目 | (a) 二级项目 |
| ◊ 三级项目 | i. 三级项目 |

1.11 文献引用

1.11.1 基本使用

使用 `\cite {<ref>}` 进行参考文献的引用, 然后使用命令 `\printbibliography` 输出参考文献. 使用 `\nocite {*}` 可以输出 .bib 中的所有参考条目. 下面是一个简单例子:

```
% 参考文献: ref.bib
@book{ahlfors1953complex,
  title={Complex Analysis},
  author={Ahlfors},
  year={1953},
  publisher={McGraw-Hill},
  address={New York}
}

% 正文引用
\cite{ahlfors1953complex}
```

默认参考文献页的名称为: Bibliography, 若想要自定义名称, 可以在输出参考文献前使用如下命令:

```
\renewcommand{\bibname}{参考文献}
\printbibliography
```


1.11.2 使用 tip

这里说明部分在 $\text{zL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 中进行参考文献引用时的 Tip. 当你想要把“参考文献”栏目加入目录, 可以使用命令:

```
\addcontentsline{toc}{chapter}{参考文献} % or
\addcontentsline{toc}{chapter}{Bibliography}
```

或者是可以这样设置:

```
\printbibliography[heading=bibnumbered, title={参考文献}]
```

其中 `heading=bibnumbered` 表示参考文献页的标题是编号的, `title` 表示标题的名称. 其实还有一种方法可用, 不过没有前面的两种那么的便捷, 而且也并不是那么的完美:

```
% \usepackage{zhnumber}
\addtocounter{chapter}{1}
\addcontentsline{toc}{chapter}{第\zhnum{chapter}章 ~~ 参考文献}
```

这样做的缺点就是, 在 pdf 书签和目录中均会显示为“第五章 参考文献”, 但我们其实不需要在前者中显示“第五章”这样的 prefix, 所以这种方法并不是合适的.

注记 1.11.1 一般情况下并不需要给“参考文献”这一小节加上对应的编号, 因此部分其实不是正文的一部分. 但是这里为了说明此命令的使用方法, 所以加上了对应的编号.

1.12 索引

1.12.1 使用方法

$\text{zL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 文档类采用 `indextools` 宏包提供的命令进行索引的生成, 并不没有使用传统的 `makeidx` 宏包. 但是其具体使用方法和 `makeidx` 宏包基本一致, 这里给一个简单示例:

```
% 导言区
\makeindex[title=Concept index]
% 添加索引到目录, 生成索引
\addcontentsline{toc}{part}{Index}
\printindex
```

或者, 你在文档导言区声明某种 index 的类型, 比如 `person`, 随后在正文中即可使用命令 `\index [person]{<the person>}` 进行索引. 索引的打印和索引格式的定制见下:

```
% 导言区索引格式定义
\makeindex[name=person, title=Index of names, columns=3]
% 文档末尾
\indexprologue{In this index you' ll find only famous people' s names}
\printindex[person]
```

使用`\index {<index>}` 命令时, 在此命令中 `<index>` 并不会显示在 PDF 文档中的, 所以如果你要添加一个“函数”的 index 项目时, 在你的 L^AT_EX 文档中应该这样写:

```
函数\index{函数}是从集合到 ...
```

若要定制`\printindex` 输出索引条目格式, 可以在导言区进行一定的设置, `indextools` 宏包提供了命令 `\indexsetup {}` 用于索引格式设定, 采用键值对的格式进行指定. 可用键: `<level>`, `<toclevel>`, `<...>`, 详情请参见对应文档. 默认 `level` 为 `\chapter *`(`\section *` 在 `article` 中): 表示此章节为无编号章节. 但为了说明用法, 这里我们设置为有编号章节, `level` 为 `chapter`:

```
\indexsetup[level=\chapter]
```

或者和前面 `\printbibliography` 类似, 可以采用如下一个不那么漂亮的做法:

```
% \usepackage{zhnumber}
\addtocounter{chapter}{1}
\addcontentsline{toc}{chapter}{第\zhnum{chapter}章 ~~ 部分命令/名词索引}
```

为何我一再坚持使用 `indextools` 宏包? 相较于传统的 `makaidx` 宏包需在命令行中先使用 L^AT_EX 引擎编译, 后再使用 `makeindex` 命令编译, 最后再使用 L^AT_EX 引擎编译两遍. 现在的 `indextools` 宏包可以在不超过两次的 L^AT_EX 引擎编译下直接生成对应 index, 方便了许多.

1.12.2 Bug

目前 index 生成工具命令所依赖的 `indextools` 宏包和 `tikz` 的 `external` 库在本文档类下有冲突. 具体表现为: 当 `indextools` 和 `external` 库同时使用时, 在第一次编译含有 `tikzpicture` 的文档时会抛出如下错误信息:

```
==== 'mode=convert with system call': Invoking 'pdflatex -halt-on-error -inter
action=batchmode -jobname "tikzdatamain-figure0" "\def\tikzexternalrealjob{release
}\input{release}"' =====

! Package tikz Error: Sorry, the system call 'pdflatex -halt-on-error -interact
ion=batchmode -jobname "tikzdata/release-figure0" "\def\tikzexternalrealjob{release}\i
nput{release}"' did NOT result in a usable output file 'tikzdatamain-figure0' (exp
ected one of .pdf:.jpg:.jpeg:.png:). Please verify that you have enabled system
```

```
calls. For pdflatex, this is 'pdflatex -shell-escape'. Sometimes it is also named 'write 18' or something like that. Or maybe the command simply failed? Error messages can be found in 'tikzdata/release-figure0.log'. If you continue now, I'll try to typeset the picture.
```

目前的解决方法有两个：

- 取消加载 `indextools` 宏包，改用传统的 `makeidx` 宏包。(需自行去修改 `zlatex.cls` 中的加载项)
- 仍然使用此宏包，但是在第一遍 (tikz 图片还没有缓存时) 取消导言区以及文档末尾的如下命令：

```
% 导言区
\makeindex[title=Test Title, columns=3]
% 文末
\addcontentsline{toc}{chapter}{部分名词索引}
\printindex
```

然后在文档的第二次编译时取消两处命令的注释，以此达到正常编译的目的。

第二部分

索引与参考

部分命令/名词索引

- `\cref` , 31, 37
- `\setmainfont` , 20
- `\setmathfont` , 27
- `\setmonofont` , 20
- `\setsansfont` , 20
- `\zlatexColorSetup` , 12
- `\zlatexFontSetup` , 20
- `\zlatexFramed` , 13
- `\zlatexOptions` , 4
- `\zlatexUpdateCounterAfter` , 15
- alias, 11
- amsthm, 15
- l3draw, 14
- newcomputermodern, 19
- toc, 8
- unicode-math, 27
- basic packages, 4
- Emoji, 27
- font-config, 19
- language packages, 4
- math packages, 5
- optional packages, 5
- slide mode, 6
- 字体族声明, 26
- 字体系列, 22
- 字体配置, 6
- 常见字体问题, 28
- 数学字体, 6, 27
- 数学环境样式, 32
- 文档类选择, 6
- 查看系统字体, 20
- 模板字体预设, 19
- 模板语言, 6
- 边注, 6
- 配置选项, 6
- 页面设置, 6

参考文献

- [1] Ahlfors. *Complex Analysis*. New York: McGraw-Hill, 1953.
- [2] Auslander and MacKenzie. *Introduction to Differentiable Manifolds*. New York: McGraw-Hill, 1963.
- [3] Michel Goossens, Frank Mittelbach, Alexander Samarin, et al. *The LATEX companion*. Vol. 1. Addison-Wesley Reading, 1994.

第三部分

Implement

zL^AT_EX Implement

4.1 文档选项

4.1.1 Key-Value

```

1 \cs_new_protected:Npn \zlatex_define_option:n
2   { \keys_define:nn { zlatex / option } }
3 \cs_new_protected:Npn \zlatex_define:nn #1
4   { \keys_define:nn { zlatex / #1 } }
5 \cs_new_protected:Npn \zlatex_set:nn #1
6   { \keys_set:nn { zlatex / #1 } }
7 \cs_new_protected:Npn \zlatex_load_after:n #1
8   { \AtBeginDocument {#1} }

```

4.1.2 Load Options

```

1 \RequirePackage {l3keys2e}
2 % key-value alias
3 \cs_new_protected:Npn \zlatex_define_option:n
4   { \keys_define:nn { zlatex / option } }
5 \cs_new_protected:Npn \zlatex_define:nn #1
6   { \keys_define:nn { zlatex / #1 } }
7 \cs_new_protected:Npn \zlatex_set:nn #1
8   { \keys_set:nn { zlatex / #1 } }
9 \cs_new_protected:Npn \zlatex_load_after:n #1
10   { \AtBeginDocument {#1} }
11
12 % setup options for class
13 \zlatex_define_option:n {
14   % zlatex language
15   lang          .str_gset:N   = \g__zlatex_lang_str,
16   lang          .initial:n    = { en },
17   hyper         .bool_gset:N  = \g__zlatex_hyperref_bool,
18   hyper         .initial:n    = { false },

```



```

19  % class and options
20  class      .str_gset:N    = \g__zlatex_subclass_type_str,
21  class      .initial:n    = { article },
22  classOption .clist_gset:N = \g__zlatex_subclass_option_clist,
23  classOption .initial:n   = { oneside, 10pt },
24  % zlatex options meta key
25  layout     .meta:nn      = {zlatex / layout}{#1},
26  mathSpec   .meta:nn      = {zlatex / mathSpec}{#1},
27  font       .meta:nn      = {zlatex / font}{#1},
28  bib        .meta:nn      = {zlatex / bib}{#1},
29  % toc setup
30  toc        .multichoice:,
31  toc / 2column .code:n    = { \zlatex_load_after:n {
\RequirePackage[toc]{multitoc} } },
32  toc / redef   .code:n    = {
33      \str_case:VnF \g__zlatex_lang_str {
34          {en}{ \zlatex_load_after:n {\renewcommand{\contentsname}{\hfill\bfseries\huge
Contents \hfill}}} }
35          {cn}{ \zlatex_load_after:n {\renewcommand{\contentsname}{\hfill\bfseries\huge
目录 \hfill}}} }
36      }\msg_error:nn {zlatex}{option-language}}
37  },
38  toc / unknown.code:n    = {
39      \msg_new:nnn {zlatex}{option-toc}{Current~toc~option~is~::~'#1'~,default~toc~settin
gs~substitute.}
40      \msg_warning:nn {zlatex}{option-toc}
41  },
42  }
43
44  % sub-key for each option
45  \zlatex_define:nn {layout}{
46      margin      .bool_gset:N    = \g__zlatex_margin_bool,
47      margin      .initial:n      = { false },
48      slide       .bool_gset:N    = \g__zlatex_slide_bool,
49      slide       .initial:n      = { false },
50      aspect      .tl_gset:N      = \g__zlatex_aspectratio_tl,
51      aspect      .initial:n      = { 12|9 },
52  }
53  \zlatex_define:nn {mathSpec}{
54      alias      .bool_gset:N    = \g__zlatex_math_alias_bool,
55      alias      .initial:n      = { false },
56      envStyle    .tl_gset:N      = \g__zlatex_math_env_style_tl,
57      envStyle    .initial:n      = { plain },
58      font        .choice:,
59      font / newtx .code:n        = { \zlatex_load_after:n {
\RequirePackage{newtxmath} } },

```

```

60 font / mtp2pro2 .code:n = { \zlatex_load_after:n { \RequirePackage[lite,
    subscriptcorrection, slantedGreek, nofontinfo]{mtp2pro2} } },
61 font / euler .code:n = { \zlatex_load_after:n { \RequirePackage[OT1,
    euler-digits]{eulervm} } },
62 font / mathpazo .code:n = { \let\rmbefore\rmdefault\zlatex_load_after:n {
    \RequirePackage{mathpazo} } \let\rmdefault\rmbefore},
63 font / unknown .code:n = {
64     \msg_new:nnn {zlatex}{option-mathfont}{Current-math-font-option-is-~'#1'~,defau
        lt-math-font-substitute.}
65     \msg_warning:nn {zlatex}{option-mathfont}
66 },
67 }
68 \zlatex_define:nn {font}{
69
70 }
71 \zlatex_define:nn {bib}{
72     source .str_gset:N = \g__zlatex_bib_source_str,
73     source .initial:n = { ref.bib },
74     backend .str_gset:N = \g__zlatex_bib_backend_str,
75     backend .initial:n = { biber },
76 }
77
78 % option setup
79 \ProcessKeysOptions {zlatex / option}
80 \NewDocumentCommand{\zlatexSetup}{m}{
81     \zlatex_set:nn {option}{#1}
82 }

```

4.2 class load

```

1 % pass clist options main subclass: 'article', 'book', 'ctexbook'
2 \str_case:VnF \g__zlatex_subclass_type_str {
3     {article}{
4         \PassOptionsToClass{\g__zlatex_subclass_option_clist}{ article }
5         \LoadClass{article}
6     }
7     {book}{
8         \PassOptionsToClass{\g__zlatex_subclass_option_clist}{ book }
9         \LoadClass{book}
10    }
11    {ctexbook}{
12        \str_set:Nn \g__zlatex_lang_str {cn}

```

```

13     \PassOptionsToClass{\g_zlatex_subclass_option_clist}{ctexbook }
14     \PassOptionsToPackage{quiet}{fontspec}
15     \LoadClass{ctexbook}
16   }
17 }{\relax}

```

4.3 compile engine

```

1  % basic document class and packages option
2  \msg_new:nnn
3  {zlatex}{compile-engine-pdfTeX}{Current~compile~engine~is~XETEX,~use~PDFTEX~instead.}
4  \msg_new:nnn {zlatex}{compile-engine-xetex
5  }{Current~compile~engine~is~PDFTEX,~use~XETEX~instead.}
6  \str_case:VnF \g_zlatex_lang_str {
7    {en} {
8      \sys_if_engine_xetex:TF
9        {\msg_warning:nn {zlatex}{compile-engine-pdfTeX}}
10       {\RequirePackage[utf8]{inputenc}}
11       \RequirePackage[T1]{fontenc}
12       \RequirePackage[english]{babel}
13     }
14     {cn} {
15       \sys_if_engine_xetex:TF {}{\msg_error:nn {zlatex}{compile-engine-xetex}}
16       \PassOptionsToPackage{quiet}{fontspec}
17       \PassOptionsToPackage{no-math}{fontspec}
18       \str_if_eq:VnF \g_zlatex_subclass_type_str {ctexbook}{
19         \RequirePackage[UTF8, heading]{ctex}
20         \linespread{1.3}
21       }
22     }
23   }{\msg_error:nn {zlatex}{option-language}}

```

4.4 Layout

4.4.1 geometry

```

1  \RequirePackage{geometry}
2  % page layout
3  \if@twoside
4    \geometry{a4paper, left=3cm, right=5.5cm, bottom=3.5cm, footskip=1.5cm,
5    marginparsep=1em}

```

```

5      \bool_if:NTF \g__zlatex_margin_bool {}{
6          \msg_new:nnn {zlatex}{option-page-margin}{No-margin-option-is-only-accessible-in
              ~oneside~layout,~margin-option-is-now-enabled-by~default.}
7          \msg_warning:nn {zlatex}{option-page-margin}
8      }
9  \else
10     \bool_if:NTF \g__zlatex_margin_bool {
11         \geometry{a4paper, left=3cm, right=5.5cm, bottom=3.5cm, footskip=1.5cm,
              marginparsep=1em}
12         \dim_gset:Nn \marginparwidth{9.25em}
13     }{
14         \geometry{a4paper, left=3cm, right=3cm, bottom=3.5cm, footskip=1.5cm,
              marginparsep=1em}
15         \renewcommand{\marginpar}[1]{\leftbar\noindent#1\endleftbar}
16     }
17 \fi

```

4.4.2 Slide mode

```

% slide mode (only in 'oneside' mode)
\exp_args:NNnx \seq_set_split:Nnn \l_tmpa_seq {}{\g__zlatex_aspectratio_tl}
\bool_if:NTF \g__zlatex_slide_bool {
    \RequirePackage{lastpage}
    \definecolor{slideRed}{HTML}{a30000}
    \definecolor{slideGray}{HTML}{e0e0e0}
    \definecolor{slideWhite}{HTML}{f0f0f0}
    % hook shipout
    \RequirePackage{eso-pic}
    \AddToShipoutPictureBG {
        \AtPageUpperLeft {
            \textcolor{slideRed}{\rule[-1.25em]{.5\paperwidth}{1.25em}}
            \textcolor{slideGray}{\rule[-1.25em]{.5\paperwidth}{1.25em}}
        }
        \AtPageLowerLeft {
            \textcolor{slideRed}{\rule{.35\paperwidth}{1.25em}}
            \textcolor{slideWhite}{\rule{.3\paperwidth}{1.25em}}
            \textcolor{slideGray}{\rule{.35\paperwidth}{1.25em}}
        }
    }
}
% layout and mark
\RequirePackage{titling}
\IfClassLoadedTF{book}{
    \renewcommand\chaptermark[1]{\markboth{#1}{}}
}

```

```

}{\relax}
\renewcommand\sectionmark[1]{\markboth{#1}{}}
\geometry {
  papersize={\seq_item:Nn \l_tmpa_seq {1}cm, \seq_item:Nn \l_tmpa_seq {2}cm},
  hmargin=1cm,
  vmargin=.65cm,
  head=0.52cm,
  foot=12pt,
  headsep=5pt,
  footskip=15pt
}
% init slide mode
\zlatex_load_after:n {\pagestyle{fancy}}
}{\relax}

```

4.4.3 fancyhdr

```

1 % fancy page settings
2 % fancy page settings
3 \cs_if_exist_use:NF \theauthor{
4   \newcommand{\theauthor}{}
5   \newcommand{\thetitle}{}
6   \newcommand{\thedata}{}
7 }
8
9 \fancypagestyle{fancy}{
10   \fancyhf{}
11   \dim_gset:Nn \headheight{15pt}
12   \renewcommand{\headrule}{\hrule width\textwidth}
13   % slide mode
14   \bool_if:NTF \g_zlatex_slide_bool {
15     \def\headrule{}
16     \fancyhead[L]{\hbox to .42\paperwidth{\hss\sffamily\textcolor{white}{Section \
17       \thesection}\;}}
18     \fancyhead[R]{\hbox to
19       .42\paperwidth{\;\sffamily\textcolor{slideRed}{\leftmark}\hss}}
20     \fancyfoot[L]{\sffamily\textcolor{white}{\theauthor}}
21     \fancyfoot[C]{\sffamily\textcolor{slideRed}{\thetitle}}
22     \fancyfoot[R]{\sffamily\textcolor{slideRed}{\hbox to
23       .25\paperwidth{\hfill\thedata\kern2.5em\thepage/\pageref{LastPage}}}}
24   }{
25     % doc mode
26     \if@twoside

```

```

24     \fancyhead[EL]{\leftmark}
25     \fancyhead[ER]{\thepage}
26     \fancyhead[OL]{\thepage}
27     \fancyhead[OR]{\rightmark}
28   \else
29     \IfClassLoadedTF{book}{
30       \fancyhead[L]{\thepage}
31       \fancyhead[R]{\rightmark}
32     }{
33       \fancyhead[L]{\thepage}
34       \fancyhead[R]{\leftmark}
35     }
36   \fi
37 }
38 }
39
40 % front and main matter cmds in book class
41 \IfClassLoadedTF{book}{
42   \renewcommand\frontmatter{
43     \cleardoublepage
44     \pagestyle{plain}
45     \@mainmatterfalse
46     \pagenumbering{Roman}
47   }
48   \renewcommand\mainmatter{
49     \cleardoublepage
50     \pagestyle{fancy}
51     \@mainmattertrue
52     \pagenumbering{arabic}
53   }
54 }{\relax}

```

4.5 math

4.5.1 math env theme

```

1 % color spec for zlatex
2 \zlatex_define:nn {color}{
3   % structure color
4   link          .tl_set:N   = \l__zlatex_link_color_tl,
5   link          .initial:n  = { purple },
6   cite          .tl_set:N   = \l__zlatex_cite_color_tl,

```

```

7      cite          .initial:n    = { teal },
8      url           .tl_set:N     = \l__zlatex_url_color_tl,
9      url           .initial:n    = { RoyalRed },
10     chapter       .tl_set:N     = \l__zlatex_chapter_color_tl,
11     chapter       .initial:n    = { RoyalRed },
12     chapter-rule   .tl_set:N     = \l__zlatex_chapter_rule_color_tl,
13     chapter-rule   .initial:n    = { black },
14     % math envs    color
15     axiom          .tl_set:N     = \l__zlatex_axiom_color_tl,
16     axiom          .initial:n    = { mathaxiomColor },
17     definition     .tl_set:N     = \l__zlatex_definition_color_tl,
18     definition     .initial:n    = { mathdefinitionColor },
19     theorem        .tl_set:N     = \l__zlatex_theorem_color_tl,
20     theorem        .initial:n    = { maththeoremColor },
21     lemma          .tl_set:N     = \l__zlatex_lemma_color_tl,
22     lemma          .initial:n    = { mathlemmaColor },
23     corollary      .tl_set:N     = \l__zlatex_corollary_color_tl,
24     corollary      .initial:n    = { mathcorollaryColor },
25     proposition    .tl_set:N     = \l__zlatex_proposition_color_tl,
26     proposition    .initial:n    = { mathpropositionColor },
27     remark         .tl_set:N     = \l__zlatex_remark_color_tl,
28     remark         .initial:n    = { mathremarkColor },
29 }
30 \NewDocumentCommand{\zlatexColorSetup}{m}{
31   \zlatex_set:nn {color}{#1}
32   \bool_if:NTF \g__zlatex_hyperref_bool {
33     \hypersetup{
34       colorlinks = true,
35       urlcolor   = \tl_use:N \l__zlatex_url_color_tl,
36       linkcolor  = \tl_use:N \l__zlatex_link_color_tl,
37       citecolor  = \tl_use:N \l__zlatex_cite_color_tl,
38     }
39   }{\relax}
40 }
41 \zlatexColorSetup{link=purple, cite=teal, url=RoyalRed}

```

4.5.2 math environment

```

1 % framed env for user interface
2 \cs_new_protected:Npn \zlatexFramed:nn #1#2 {
3   \DeclareDocumentEnvironment{#1}{0{#2}}{
4     \def\FrameCommand{{\color{##1}\vrule width 3pt}\colorbox{##1!10}}
5     \MakeFramed{\advance\hsize-\width\FrameRestore}\noindent

```

```

6      }\endMakeFramed}
7  }
8  \NewDocumentCommand\zlatexFramed{mO{black}}{
9      \zlatexFramed:nn {#1}{#2}
10 }
11
12 % theorem/proof-like envs list
13 \clist_gset:Nn \g__zlatex_theoremlike_envs_clist {
14     axiom, definition, theorem,
15     lemma, corollary, proposition, remark
16 }
17 \clist_gset:Nn \g__zlatex_prooflike_envs_clist {
18     proof, exercise, example,
19     solution, problem,
20 }
21
22 % math envs' name accordingt to language
23 \msg_new:nnn {zlatex}{mathenv-name}{Current-math-env-name-is-~'#1'~,which-is-invalid.}
24 \str_case:VnTF \g__zlatex_lang_str {
25     {en}{
26         \zlatex_define:nn {math-env}{
27             math-env .multichoice:,
28             math-env / axiom .code:n = { \tl_gset:cn {zlatex#1Name}{Axiom} },
29             math-env / definition .code:n = { \tl_gset:cn {zlatex#1Name}{Definition}
30             },
31             math-env / theorem .code:n = { \tl_gset:cn {zlatex#1Name}{Theorem} },
32             math-env / lemma .code:n = { \tl_gset:cn {zlatex#1Name}{Lemma} },
33             math-env / corollary .code:n = { \tl_gset:cn {zlatex#1Name}{Corollary} },
34             math-env / proposition .code:n = { \tl_gset:cn {zlatex#1Name}{Proposition}
35             },
36             math-env / remark .code:n = { \tl_gset:cn {zlatex#1Name}{Remark} },
37             math-env / proof .code:n = { \tl_gset:cn {zlatex#1Name}{Proof} },
38             math-env / exercise .code:n = { \tl_gset:cn {zlatex#1Name}{Exercise} },
39             math-env / example .code:n = { \tl_gset:cn {zlatex#1Name}{Example} },
40             math-env / solution .code:n = { \tl_gset:cn {zlatex#1Name}{Solution} },
41             math-env / problem .code:n = { \tl_gset:cn {zlatex#1Name}{Problem} },
42             math-enc / unknown .code:n = {
43                 \msg_error:nn {zlatex}{mathenv-name}
44             },
45         }
46     }
47     {cn}{
48         \zlatex_define:nn {math-env}{
49             math-env .multichoice:,
50             math-env / axiom .code:n = { \tl_gset:cn {zlatex#1Name}{公理} },

```



```

49      math-env / definition .code:n = { \tl_gset:cn {zlatex#1Name}{定义} },
50      math-env / theorem .code:n = { \tl_gset:cn {zlatex#1Name}{定理} },
51      math-env / lemma .code:n = { \tl_gset:cn {zlatex#1Name}{引理} },
52      math-env / corollary .code:n = { \tl_gset:cn {zlatex#1Name}{推论} },
53      math-env / proposition .code:n = { \tl_gset:cn {zlatex#1Name}{命题} },
54      math-env / remark .code:n = { \tl_gset:cn {zlatex#1Name}{注记} },
55      math-env / proof .code:n = { \tl_gset:cn {zlatex#1Name}{证明} },
56      math-env / exercise .code:n = { \tl_gset:cn {zlatex#1Name}{练习} },
57      math-env / example .code:n = { \tl_gset:cn {zlatex#1Name}{示例} },
58      math-env / solution .code:n = { \tl_gset:cn {zlatex#1Name}{解} },
59      math-env / problem .code:n = { \tl_gset:cn {zlatex#1Name}{问题} },
60      math-enc / unknown .code:n = {
61          \msg_error:nn {zlatex}{mathenv-name}
62      },
63  }
64 }
65 {\zlatex_set:nn {math-env}{math-env={axiom, definition, theorem, lemma, corollary,
66 proposition, remark, proof, exercise, example, solution, problem}}}
67
68 % math env's style
69 \newtheoremstyle{zlatexMathEnv}
70     {2pt}{2pt}{}
71     {0pt}{\bfseries}{}
72     {.25em}{\thmname{#1}~ \thmnumber{#2}~ \thmnote{(#3)}}
73 \theoremstyle{zlatexMathEnv}
74
75 % theorem-like env declaration
76 \msg_new:nnn {zlatex}{mathenv-type}{Current~math~env~is:~'\str_use:N \g__zlatex_math_en
77 v_style_tl'~,only~'plain',~'leftbar',~'background',~'fancy'~types~are~valid.}
78 \DeclareDocumentEnvironment{zlatexTheoremLikeFrame}{0{black}}{
79     \str_case:VnF \g__zlatex_math_env_style_tl {
80         {plain}{
81             \def\FrameCommand{}
82             \MakeFramed {\advance\hsize-\width \FrameRestore}
83         }
84         {leftbar}{
85             \def\FrameCommand{{\color{#1}\vrule width 3pt}\hspace{5pt}}
86             \MakeFramed {\advance\hsize-\width \FrameRestore}
87         }
88         {background}{
89             \def\FrameCommand{\colorbox{#1!10}}
90             \MakeFramed {\advance\hsize-\width \FrameRestore}
91         }
92         {fancy}{

```

```

92     \def\FrameCommand{{\color{#1}\vrule width 3pt}\colorbox{#1!10}}
93     \MakeFramed{\advance\hsize-\width \FrameRestore}
94   }
95   }{\msg_error:nn {zlatex}{mathenv-type}}
96 }{\endMakeFramed}
97
98 % loop to create math env, setup \cref
99 \clist_map_inline:Nn \g__zlatex_theoremlike_envs_clist {
100   % theorem create
101   \newtheorem{zlatex#1}{\tl_use:c {zlatex#1Name}}[section]
102
103   % env create (3 types: 'leftbar', 'none' and 'backgroud')
104   \NewDocumentEnvironment{#1}{0{}}{
105     \begin{zlatexTheoremLikeFrame}[\tl_use:c {l__zlatex_#1_color_tl}]
106     \begin{zlatex#1}[\##1]
107   }{\end{zlatex#1}\end{zlatexTheoremLikeFrame}}
108
109   % cref settings
110   \cs_generate_variant:Nn \exp_args:Nnnx {Nxxx}
111   \str_case:VnF \g__zlatex_lang_str {
112     {en}{
113       \crefname{zlatex#1}{#1}{#1s}
114       \creflabelformat{zlatex#1}{##2(##1)##3}
115     }
116     {cn}{
117       \exp_args:Nxxx \crefname{zlatex#1}{\tl_use:c {zlatex#1Name}}{\tl_use:c
118         {zlatex#1Name}}
119       \creflabelformat{zlatex#1}{##2(##1)##3}
120     }
121   }{\msg_error:nn {zlatex}{option-lang}}
122 }
123
124 % proof-like env decalration
125 \NewDocumentEnvironment{zlatexProofLikeFrame}{0{}}{
126   \def\FrameCommand{}
127   \MakeFramed {\advance\hsize-\width \FrameRestore}
128 }{\endMakeFramed}
129 \renewcommand{\qedsymbol}{\ensuremath{\blacksquare}}
130 \clist_map_inline:Nn \g__zlatex_prooflike_envs_clist{
131   \DeclareDocumentEnvironment{#1}{0{}}{
132     \begin{zlatexProofLikeFrame}[]
133     {\noindent{\bfseries\tl_use:c {zlatex#1Name}:}}
134     \tl_set:Nn \l__arg_one_tl {#1}
135   }{\str_if_eq:VnTF
136     \l__arg_one_tl{proof}{\hfill\qedsymbol\par}{\par}\end{zlatexProofLikeFrame}}

```

135

}

4.5.3 Math alias

```

1  % math related cmds alias
2  \bool_if:NTF \g_zlatex_math_alias_bool{
3      \RequirePackage{amssymb, mathtools}
4      \RequirePackage{bm}
5      % Math Font
6      \newcommand{\dd}{\mathrm{d}}
7      \newcommand{\C}[1]{\ensuremath{\mathcal{#1}}}
8      \let\ss\S
9      \renewcommand{\S}[1]{\ensuremath{\mathscr{#1}}}
10     \newcommand{\B}[1]{\ensuremath{\mathbb{#1}}}
11     \newcommand{\FF}[1]{\ensuremath{\mathbf{#1}}}
12     \newcommand{\F}[1]{\ensuremath{\bm{#1}}}
13     \newcommand{\R}[1]{\ensuremath{\mathrm{#1}}}
14     \newcommand{\K}[1]{\ensuremath{\mathfrak{#1}}}
15     % Math Arrow
16     \newcommand{\lr}{\ensuremath{\longrightarrow}}
17     \let\LL\ll
18     \renewcommand{\ll}{\ensuremath{\longleftarrow}}
19     \newcommand{\equ}{\ensuremath{\Longleftarrow\rightarrow}\,}
20     \newcommand{\sr}{\ensuremath{\longmapsto}}
21     \newcommand{\lrr}[2][\,]{\ensuremath{\xrightarrow{#1}{#2}}}
22     \renewcommand{\lll}[2][\,]{\ensuremath{\xleftarrow{#1}{#2}}}
23     \newcommand{\ns}{\ensuremath{\varnothing}}
24     \newcommand{\A}{\ensuremath{\forall}}
25     % Math Operator
26     \newcommand{\alt}{\ensuremath{\mathrm{Alt}\,;\,}}
27     \newcommand{\sgn}{\ensuremath{\mathrm{sgn}\,;\,}}
28     \newcommand{\curl}{\ensuremath{\mathrm{curl}\,;\,}}
29     \newcommand{\grad}{\ensuremath{\mathrm{grad}\,;\,}}
30     \newcommand{\trace}{\ensuremath{\mathrm{trace}\,;\,}}
31     \renewcommand{\div}{\ensuremath{\mathrm{div}\,;\,}}
32 }{}
```

4.6 structure Style

4.6.1 hyperref/index and bib

```

1 \str_case:VnF \g__zlatex_bib_backend_str {
2     {bibtex}{\RequirePackage[backend=bibtex]{biblatex}}
3     {biber}{\RequirePackage[backend=biber]{biblatex}}
4 }{\relax}
5 \exp_args:Nx \addbibresource{\str_use:N \g__zlatex_bib_source_str}
6 \RequirePackage{indextools}
7 \RequirePackage{hyperref}

```

4.6.2 cref name

```

1 \RequirePackage{indextools}
2 \RequirePackage{hyperref}
3 \RequirePackage[nameinlink]{cleveref}
4
5 % figure and table prefix for \cref
6 \str_case:VnF \g__zlatex_lang_str {
7     {en}{
8         \IfClassLoadedTF{book}{
9             \crefname{part}{part}{parts}
10            \crefname{chapter}{chapter}{chapters}
11        }{\relax}
12        \crefname{section}{section}{sections}
13        \crefname{subsection}{subsection}{subsections}
14        \crefname{figure}{figure}{figures}
15        \crefname{table}{table}{tables}
16        \crefname{equation}{equation}{equations}
17    }
18    {cn}{
19        \IfClassLoadedTF{book}{
20            \crefname{part}{部分}{部分}
21            \crefname{chapter}{章}{章}
22        }{\relax}
23        \crefname{section}{节}{节}
24        \crefname{subsection}{小节}{小节}
25        \crefname{figure}{图}{图}
26        \crefname{table}{表}{表}
27        \crefname{equation}{方程}{方程}
28    }
29 }{\msg_error:nn {zlatex}{option-language}}

```

4.6.3 contents

```

1  % partial ToC
2  \RequirePackage{titletoc}
3  \NewDocumentCommand\stopPartToc{mm}{
4    \int_compare:nNnTF {#2}<{1}
5      {\relax}{\stopcontents[#1]}
6  }
7  \NewDocumentCommand{\partialToc}{0{2}}{
8    % depth setup
9    \setcounter{tocdepth}{#1}
10   \titlecontents{psection}[2em]
11     {} {\contentslabel{2em}} {} {\titlerule*[1pc]{.}\contentspage}
12   \titlecontents{psubsection}[4.5em]
13     {} {\contentslabel{2.5em}} {} {\titlerule*[1pc]{.}\contentspage}
14   \titlecontents{psubsubsection}[5.5em]
15     {} {\contentslabel{0em}} {} {\titlerule*[1pc]{.}\contentspage}
16   % print pToc
17   \IfClassLoadedTF{book}{
18     \startcontents[chapters]
19     \printcontents[chapters]{p}{1}{}
20     \pretocmd{\chapter}{
21       \stopPartToc{chapters}{\thechapter}
22     }{}{}
23   }{
24     \startcontents[sections]
25     \printcontents[sections]{p}{1}{}
26     \pretocmd{\section}{
27       \stopPartToc{sections}{\thesection}
28     }{}{}
29   }
30 }

```

4.6.4 Chapter and section

```

1  % chapter head style
2  \IfClassLoadedTF{book}{
3    \RequirePackage{titlesec}
4    \titleformat{\chapter}[display]
5      {\bfseries\huge\color{black}}

```

```

6      {\flushright\Large\color{\tl_use:N \l__zlatex_chapter_color_tl}
7      \MakeUppercase{\chaptertitlename}\hspace{1ex}
8      {\scalebox{4}{\thechapter}}}}
9      {10pt}
10     {\color{\tl_use:N \l__zlatex_chapter_rule_color_tl}\titlerule\vspace{1ex}}
11     % chapter space
12     \titlespacing{\chapter}{0pt}{-30pt}{40pt}
13     \titleformat{name=\chapter, numberless}
14         {\normalfont\bfseries\Huge}
15         {}{0pt}{-}
16 }{\relax}
17
18 % title page
19 % title page
20 \let\orimaketitle\maketitle
21 \def\Maketitle{
22     \newgeometry{margin=1in}
23     \orimaketitle
24     \restoregeometry
25 }
26 \bool_if:NTF \g__zlatex_slide_bool {
27     \let\orithepage\thepage
28     \renewcommand\maketitle{
29         \Maketitle
30         \def\thepage{}
31         \setcounter{page}{0}
32         \newpage
33         \let\thepage\orithepage
34     }
35 }{
36     \renewcommand{\maketitle}{
37         \bool_if:NTF \g__zlatex_hyperref_bool {\hypersetup{pageanchor=false}}{\relax}
38         \newgeometry{margin=1in}
39         \begin{titlepage}
40             \vfill\vspace*{40pt}
41             \noindent\hspace*{134pt}\rule[-75pt]{6pt}{95pt}
42             {\hspace*{10pt}\leavevmode\parbox[t]{25em}{\fontsize{25}{25}\selectfont\bfseries\@
43             title}}\par
44             \vspace*{-15pt}
45             \noindent\hspace*{150pt}{\leavevmode\parbox[t]{30em}{\Large\bfseries\@author}}\par
46             \vfill
47             \noindent\hspace*{150pt}{\Large\textcolor{gray}{\@date}}
48         \end{titlepage}
49         \restoregeometry
50         \bool_if:NTF \g__zlatex_hyperref_bool {\hypersetup{pageanchor=true}}{\relax}

```

```
50   }  
51 }  
52  
53 % reset counter command  
54 \NewDocumentCommand{\zlatexUpdateCounterAfter}{mm}{\@addtoreset{#1}{#2}}
```