

EASYBOOK 使用手册

v2024eh 2024/10/08

简便地排版中文学位论文或书籍

瞿毅

<https://gitee.com/texno3/easybook>

toquyi@163.com

`easybook` 是基于 `ctexbook` 书籍文档类创建的纯净学术模板，同时兼具 `book` 与 `article` 文档类的功能。结合中国许多高校学位论文的通用框架设计，提供多个命令和接口可以使用户简便地定制论文模板。它的基础宏包 `easybase` 还可以搭配 \LaTeX 和标准文档类使用。

目录

1 准备开始	2 4 环境和命令	23
1.1 前言	4.1 封面和文区	23
1.2 基本用法	4.2 目录命令	24
1.3 编译方式	4.3 双语章节标题和新建标题	26
1.4 依赖宏包	4.4 上标和下标	28
	4.5 公式	29
	4.6 表格	29
2 全局选项	4.7 列表	29
2.1 基本选项	4.8 悬挂缩进	30
2.2 配置字体	4.9 计数器	30
2.3 <code>article</code> 模式		
	5 扩展功能	31
3 模块选项	5.1 盒子	31
3.1 工具	5.2 代码	32
3.2 超链接	5.3 物理量	33
3.3 间距	5.4 标签引用	33
3.4 部件字体	5.5 索引	33
3.5 参考文献	5.6 边注	34
3.6 定理	5.7 公式符号对照	34
3.7 页面尺寸	5.8 插入文档	35
3.8 图表题注		
3.9 页眉页脚	A 更新日志	35
3.10 脚注	B 参考文献	37
3.11 目录	C 索引	38

1 准备开始

您在使用 **easybook** 文档类编写文档前可能需要具备以下的条件：

- 使用 XeLaTeX （推荐）或 LuaLaTeX 两种编译方式之一。
- 由于文档类用到了一些 LaTeX 和依赖宏包较新的功能，需要版本至少在 **2021/11/15** 以后的 LaTeX 内核支持。建议您安装最新的 TeX 发行版。
- 虽然手册对基本用法做了介绍，但这并不是入门教程，所涉及到宏包的用法与命令并未进行详细介绍，需要对它们有所了解。您应当具有一定的 LaTeX 使用基础，可以在 TeX 综合网站 [CTAN](#) 查看相关资源。

1.1 前言

对于排版外文文档已有诸如 KOMA-Script 和 memoir 等应用广泛的文档类，但这些文档类并不是为中文用户设计的，难免有些水土不符，且用户手册繁杂，对于普通用户来说学习成本高，难以快速上手。中文文档的解决方案 CTeX 宏集解决了中文输出和章节标题格式的设置问题，但在编写文档过程中往往需要更多基本的定制功能，例如目录、图表标题、页眉页脚、脚注和页面边距等，这往往需要额外加载不同功能的宏包。**easybook** 文档类在 **ctexbook** 基础上载入了一些使用频率较高且基本稳定的宏包，对它们的功能进行整合并优化联动使用，模块化的改装提供统一的接口兼容 CTeX 文档类，可以满足学术类文档例如中国学位论文模板的基本需求。**easybook** 提供了与 CTeX 相同现代化风格的键值设置，支持不同类型的选项与层次化的选项设置。

1.2 基本用法

以下一份简单的 TeX 文档演示了 **easybook** 的最基本用法，在导言区进行宏包载入与文档设置，一些详细的使用样例会不定期上传在 [LaTeX 工作室](#) 的网站。或者直接使用 **easybase** 宏包，目前适用于 **ctexbook**、**ctexrep**、**ctexart** 以及标准文档类，包含了 [节 3](#) 所示各个模块接口命令的功能。当在**标准文档类**后载入时还可以使用 **ctex** 宏包的选项，将会传递给 **ctex** 宏包。文档类和宏包支持的全局选项见 [节 2](#)。

```
\documentclass[全局选项]{easybook}
% \documentclass[全局选项]{ctexbook}

% \usepackage[全局选项]{easybase}
\ctexset{键值列表}
\usepackage{宏包列表}

\begin{document}
\chapter{hello :)}
\section{Welcome to easybook}
Hello, \LaTeX
\end{document}
```

1.3 编译方式

假设您的 TeX 源文件名为 **easybook-demo.tex**，需要使用参考文献和索引功能，并使用 XeLaTeX 和 BibTeX 引擎编译文档，那么需要在命令行中执行：

```
xelatex -shell-escape easybook-demo
bibtex easybook-demo
makeindex easybook-demo
xelatex -shell-escape easybook-demo
xelatex -shell-escape easybook-demo
```

当然也可以使用 `latexmk` 及一些软件自带的编译工具，这里就不做介绍。

► 需要注意正确编译索引要增加 `-shell-escape` 命令行选项。

1.4 依赖宏包

表 1 为 `easybase` 直接调用的关键宏包清单，通常情况下它们是常用且稳定的。蓝色背景的宏包表示修改了或依赖它们的内部命令。绿色背景的宏包表示并未调用它们，但做了一些预配置，可在导言区手动载入以实现相关的功能。

表 1 依赖的宏包清单

amsthm	bicaption	ctex	caption	chemformula
enumitem	etoolbox	fancyhdr	geometry	hyperref
imakeidx	listings	marginnote	mathtools	multicol
pifont	pdfpages	siunitx	spbookmark	thmtools
tcolorbox	titletoc	tabularray	xcolor	graphicx

2 全局选项

全局选项应当在使用 `easybook` 文档类或 `easybase` 宏包（部分支持）时指定，可见小节 1.2 的例子。选项前带有 © 符号表示只可用于文档类，否则文档类与宏包都适用。文档类选项除了下面额外增加的以外，还支持标准和 \LaTeX 文档类的选项。

2.1 基本选项

draft

开启草稿模式会加快编译速度，表现有显示页面边框、行溢出的地方显示黑色方块、图片变成占位方框、显示当前日期和关闭超链接渲染。不使用绘图宏包及依赖 `tcolorbox` 宏包的盒子环境也会增快编译速度，详情见小节 5.1。

© newline

部分和章标题的样式为另起一行，符合一般英文书籍的习惯。

© scheme = `chinese` | `plain`

文档主题方案选项 `scheme` 扩展了 \LaTeX 文档类中同名选项原有的作用范围，选项值 `plain` 可使各类标题变为英文形式。包括章节标题、图表标题和定理名等。

© enmode

同时开启 `newline` 和 `scheme = plain` 选项以符合英文原生风格。

`paper = a4paper | b5paper | c5paper | letterpaper | other paper`

文档页面尺寸选项将参数交给 `geometry` 宏包处理，可使用它支持的所有标准纸张规格。

☉ `configs =` (配置文件列表)

载入用户配置的文件名列表，文件列表以花括号包裹并以英文逗号分隔。文件名应包含如 `tex`、`def` 或 `cfg` 等扩展名。若配置文件不在当前文件夹则文件名需要包含路径，即 `path/file.cfg`。

`floatpage = true | false`

开启这个选项后浮动环境 `table` 和 `figure` 的比例参数会使用默认值。关闭它则尽可能避免出现浮动页，使图表环境与正文段落更加紧凑，减少多余的空白版面。

`theorem = thmtools | keytheorems | false`

是否提供定理功能支持，定理模块的介绍可见 [小节 3.6](#)。选项 `thmtools` 会载入 `amsthm` 和 `thmtools` 宏包，选项 `keytheorems` 会载入 `keytheorems` 宏包，选项 `false` 不载入定理功能宏包。

`chem = chemformula | mhchem | false`

是否提供化学式功能支持。选项 `chemformula` 会载入 `chemformula` 宏包，选项 `mhchem` 会载入 `mhchem` 宏包，选项 `false` 不载入化学式功能宏包。`chemformula` 的工作原理与 `mhchem` 非常相似，但对化合物、化学计量数和箭头的输入方式控制更为严格。

`\ch[⟨键值列表⟩]{⟨化学式⟩}`

`chemformula` 的无机化学式书写命令。化合物表达式中的原子数不需要加下标符号，化学计量数与表达式间隔一个空格。

例 1

```
\ch{[Cu(NH3)4]^2+}
\ch{Cu2S + 2 O2 - 2 e^- == 2 Cu^2+ + SO4^2-}
```

2.2 配置字体

☉ `cjkfont = adobe | fandol | founder | mac | macnew | macold | ubuntu | windows | none`

中文字体采用 `CTEX` 宏集包含的字库并与其一致，具体配置方案可以查看 `CTEX` 使用手册。它与 `fontset` 选项是等效的，如果没有指定 `cjkfont` 的值，将会自动检测操作系统并选择相应的字体。

如果需要使用其它中文字体，可以在启用选项 `cjkfont = none` 后，再使用 `xeCJK` 宏包的相关命令设置字体以及声明字体族：

例 2

```
\setCJKmainfont{SimSun}[BoldFont = SimHei,ItalicFont = KaiTi]
\setCJKsansfont{SimHei}
\setCJKmonofont{FangSong}
\newCJKfontfamily[song]\songti{SimSun}
```

如果用 `Overleaf` 或 `TeXPage` 等在线平台编译文档，则只有 `fandol` 字体可用。使用其它字体可将字体文件上传至项目文件夹，例如上传至 `font` 文件夹设置正文和无衬线字体：

例 3

```
\setCJKmainfont{simsun.ttc}[Path = ./font/]
\setCJKsansfont{simhei.ttf}[Path = ./font/]
```

☉ `font = noto | times | times* | ebgarmond | libertine | palatino | auto`

预置的英文字体相关配置方案如 [表 2](#) 所示。这些字体已经包含在了 `TeX Live` 发行版中，所以通常

不需要再专门安装。如果您想使用默认的 Latin Modern 或其它英文字体，可以使用 `auto` 选项配合 `fontspec` 宏包的相关命令进行设置。

表 2 英文字体预设方案

	正文字体	无衬线字体	等宽字体
<code>noto</code>	Noto Serif	Noto Sans	Noto Sans Mono
<code>times</code>	XITS	TG Heros	TG Cursor
<code>times*</code>	Times New Roman	Arial	Courier New
<code>ebgaramond</code>	EB Garamond	Libertinus Sans	LM Mono
<code>palatino</code>	TG Pagella	Libertinus Sans	LM Mono
<code>libertine</code>	Libertinus Serif	Libertinus Sans	LM Mono

► `times*` 方案为 windows 系统的预置字体。

☺ `mathfont = [times] | noto | notosans | nc | ncf | stix2 | charter | ebgaramond | libertine
minion | garamondx | baskervaldx | baskerville | none`

选择合适的数学字体，将会加载 `newtxmath` 和 `bm` 宏包。使用选项 `none` 时可以使用 `unicode-math` 等其它数学字体宏包定义字体。

2.3 article 模式

`class = [book] | article`

使用 `easybook` 文档类可以有两种模式，默认为 `book` 模式。选用 `article` 可用于短篇幅的文章，文档会变为 `oneside` 单面排版，同时标题编号和封面等与标准 `article` 文档类相似，若使用 `\chapter` 命令会报错。如果使用 `ctexart` 配合 `easybase` 宏包，则选项值 `book` 会失效。

`book`

开启 `book` 书籍模式，与 `class = book` 等效。

`article`

开启 `article` 文章模式，与 `class = article` 等效。

3 模块选项

`\ctexset[⟨模块名⟩]{⟨键值列表⟩}`

`easybook` 扩展了 \LaTeX 宏集 \LaTeX 3 风格键值的通用控制命令 `\ctexset`，模块选项同样可用它设置，原有的 `(键值列表)` 仍然可用。本文档类目前有 10 个模块用于实现常规的排版功能，如表 3 所示。

表 3 模块功能和名称清单

工具	部件字体	间距	超链接	参考文献
<code>tools</code>	<code>font</code>	<code>space</code>	<code>ref</code>	<code>bib</code>
定理	页面尺寸	页眉页脚	目录	图表题注
<code>thmt</code>	<code>geo</code>	<code>hdr</code>	<code>toc</code>	<code>cap</code>

例 4 使用 [小节 3.1](#) 介绍的工具模块 `tools` 举例三种键值列表设置方式：1. 各 `<模块名>` 是统一接口 `\ctexset` 的键路径。2. 不同模块还具有单独的接口命令 `\eb<模块名>set`。3. `<模块名>` 也可以使用 `\ctexset` 命令的可选参数设置。路径名与所属下级选项之间可以 `/` 或 `=` 号分开，斜线 `/` 后不能有空格或者换行。

例 4

```
\ctexset
{
  tools/color      = seaside,
  tools            =
  {
    fntnumwith     = page,
    uppercase      = first
  }
}
\ctexset[tools]{spcmd = \color{red}}
\ebtoolsset{hyphen-figure = {-}}
```

`easybook` 文档类有四种选项：1. 布尔型 `key = true | false` 选项的值为真 `true` 或假 `false`，如果不指定默认为真。2. 键值型 `key = <value>` 需要为选项指定值，后面 `=` 号不能省略。3. 键型 `key` 不需要为选项指定值。4. 选择型 `key = one | two | three` 根据提供的可选值进行设定。

tools ⇒ 3.1 工具

工具模块 `tools` 主要用于综合设置各类杂项的排版元素。

`color = none | seaside | energy | cyberpunk`

`easybook` 预置了四种颜色主题，默认主题 `none` 为黑白页面，颜色主题不包括超链接。

`\DeclareThemeColor{<风格名称>,<颜色1>,<颜色2>,<颜色3>}`

声明新的颜色主题命令。它的参数以英文逗号分隔，第一部分为颜色风格名称，后面三部分为 HTML 颜色代码，构成主题颜色列表。其中 `<颜色1>` 为标题标签，`<颜色2>` 为列表编号、定理标签和公式编号括号，`<颜色3>` 为代码抄录。类似地更改超链接颜色见 [小节 3.2](#)。

例 5

```
\DeclareThemeColor{fancy,2acb25,60aacc,ebc01c}
\ctexset{tools/color = fancy}
```

`color-every = <HTML颜色代码>`

统一设置主题的各部分颜色。参数是 6 个十六进制数字组成的 HTML 颜色代码，任意颜色的代码可以在常见的图片处理工具拾色器中查看。

`multoc = <整数>`

Default: 2

设置目录的排版栏数，与 [小节 4.2](#) 中目录命令的同名选项有相同效果。

`withpart = true | false`

使章标题跟随部分计数器编号，仅在导言区有效。当开启新的部分标题时章编号从 1 开始。

`fntnumwith = part | chapter | page`

设置脚注重置编号的方式，仅在导言区有效。默认跟随章计数器，即开启新的章标题时脚注编号从 1 开始。选项 `page` 代表的是换页而不是页计数器，这是由于在一些排版情况下换页后页计数器仍为

上页的值。

`uppercase = all | first | none | none*`

使用 `\rightmark` 和 `\leftmark` 等标题标记命令获取标题内容用于页眉页脚时其中英文的大小写类型。`all` 为标题标记中的英文字母全部大写。`first` 为标记以英文起始时仅第一个字母大写。`none` 不改变标记中英文字母的大小写形式, 变体 `none*` 则强制标记的字母均为小写形式。

`tagform = {⟨左括号⟩,⟨右括号⟩}`

Default: `{(,)}`

设置行间公式标签的括号及格式, 以英文逗号分隔, 分别代表左右括号。⟨左括号⟩的末尾可以是带有一个参数的命令, 用于接受公式编号。将标签括号设置为方括号, 并给编号使用无衬线字体:

例 6

```
\newcommand{\bracketsf}[1]{\textsf{#1}}
\ebtoolsset{tagform = {\bracketsf,}}
```

`figurepath = {⟨路径名1⟩}{⟨路径名2⟩}...{⟨路径名n⟩}`

设置插入图片的文件夹路径名。每个路径需包含在花括号内, 只有一个路径时括号可以省略。

例 7

```
\ctexset{tools/figurepath = path1}
\ctexset{tools/figurepath = {path2}{path3}}
```

以下几个选项用于设置题注编号的连字符, `chapter` 计数器的值大于 0 时连字符才有意义。若需更改题注编号的输出格式可在正文中重新定义 `\the⟨计数器⟩` 命令。

`hyphen-figure = ⟨图片连字符⟩`

Default: `{.}`

设置图片题注编号的连字符。

`hyphen-table = ⟨表格连字符⟩`

Default: `{.}`

设置表格题注编号的连字符。

`hyphen-lstlisting = ⟨代码连字符⟩`

Default: `{.}`

设置代码题注编号的连字符。`listings` 宏包的 `numberbychapter` 选项失效。

`hyphen-equation = ⟨公式连字符⟩`

Default: `{.}`

设置公式题注编号的连字符。

`hyphen-every = ⟨连字符⟩`

Default: `{.}`

统一设置上述所有编号的连字符。

ref ⇒ 3.2 超链接

模块 `ref` 将超链接选项交给 `hyperref` 宏包处理。除了一些超链接颜色的选项失效, 其它由宏包全局和接口命令 `\hypersetup` 支持的多数选项可用。

`linktype = edge | colors | none`

超链接类型。`edge` 使超链接形式为边框, `colors` 使超链接字体为彩色, `none` 关闭超链接边框与颜色, 此时 `linkcolor` 选项无效。

`linkcolor = fresh | cutepink | navyblue | crimson`

当超链接类型为 `colors` 时文本的颜色样式。预置了四种超链接颜色风格, 默认为海军蓝 `navyblue`。

`\DeclareLinkColor{⟨风格名称⟩,⟨颜色1⟩,⟨颜色2⟩,⟨颜色3⟩}`

声明新的超链接颜色样式命令。它的参数以英文逗号分隔, 第一部分为颜色风格名称, 后面三部分

为 HTML 颜色代码，构成超链接颜色列表。其中〈颜色1〉为普通链接，〈颜色2〉为网址，〈颜色3〉为参考文献引用。

`linkcolor-every` = 〈HTML颜色代码〉

统一设置超链接的各部分颜色。参数是 6 个十六进制数字组成的 HTML 颜色代码，任意颜色的代码可以在常见的图片处理工具拾色器中查看。

`linktoc` = none|section|page|all

将目录中的超链接取消、置于标题上、置于页码上或置于标题和页码上。

space ⇒ 3.3 间距

间距模块 `space` 用于调整段落中的间距。带有 `spread` 的选项可以设置行距倍数，〈数值〉是不带有长度单位的浮点数。各类行距既可以在导言区全局设置，也可以在正文中设置对后面生效。

`linespread` = 〈数值〉 Default: 1.354

正文行距因子。中文主题下行距因子初始值为 1.354（相当于 Word 的 1.25 倍行距），如果设置了文类或 `ctex` 宏包的全局行距选项 `linespread` 则初始值设为它的值。

`capfig-spread` = 〈数值〉 Default: 1.2

图片题注的行距因子。

`captab-spread` = 〈数值〉 Default: 1.2

表格题注的行距因子。

`caplst-spread` = 〈数值〉 Default: 1.2

代码题注的行距因子。

`caption-spread` = 〈数值〉 Default: 1.2

同时设置图片、表格和代码题注的行距因子。

`fntspread` = 〈数值〉 Default: 1.2

脚注内容整体的行距因子。可以使用 `hdr` 模块 `fnpara-format` 选项的代码单独覆盖每个脚注的行距，从而使脚注间和内容文本自身的行距不同。

例 8

```
\spaceset{fntspread = 1.3}
\ebhdrset{fnpara-format = \usestretch{1.2}}
```

`everyspread` = 〈数值〉

统一设置上述所有的行距因子。

`abovedisplayskip` = 〈弹性长度〉 Default: 0.5ex plus .1ex minus .1ex

行间公式与上面正文间的额外距离。

`belowdisplayskip` = 〈弹性长度〉 Default: 0.5ex plus .1ex minus .1ex

行间公式与下面正文间的额外距离。

`\usestretch{〈数值〉}`

设置之后文本的行距。可在任意地方使用，作用范围为局部。

`\begin{usespace}{〈数值〉}`

设置环境内文本的行距。可包含行间公式和表格等环境。

New
2024dt

New
2024dt

例 9

```
\begin{usespace}{1.5}\nointerlineskip
\ebspaceset{belowdisplayskip = 6pt}
\begin{align}
\cos 2x &= \cos^2 x - \sin^2 x \\
&= 2\cos^2 x - 1
\end{align}
\end{usespace}
```

font ⇒ 3.4 部件字体

部件字体模块 **font** 用于设置除了正文外其它排版元素的字体。

caption-table = \langle 字体命令 \rangle Default: `\sffamily\small`

设置表格题注的字体，包括字体族、字体系列、字体尺寸和字体形状命令。

caption-figure = \langle 字体命令 \rangle Default: `\sffamily\small`

设置图片题注的字体，包括字体族、字体系列、字体尺寸和字体形状命令。

caption-lstlisting = \langle 字体命令 \rangle Default: `\sffamily\small`

设置代码题注的字体，包括字体族、字体系列、字体尺寸和字体形状命令。

caption-every = \langle 字体命令 \rangle Default: `\sffamily\small`

同时设置图片、表格和代码题注的字体命令。

footnote = \langle 字体命令 \rangle

设置脚注标记与内容的字体，包括字体族、字体系列和字体形状命令。

marginpar = \langle 字体命令 \rangle Default: `\footnotesize`

设置边注的字体，包括字体族、字体系列、字体尺寸和字体形状命令。

\ebaddtofont{ \langle 字体类型 \rangle }{ \langle 字体命令 \rangle }

用于在上述各种字体原有的格式上附加代码。其中 \langle 字体命令 \rangle 也可以包括行距命令且相对于 **space** 模块具有优先性，不同题注的 \langle 字体类型 \rangle 则为相应选项的后缀名。

例 10

```
\ebaddtofont{every}{\usestretch{1.5}}
\ebaddtofont{footnote}{\sffamily}
```

bib ⇒ 3.5 参考文献

backend = `bibtex` | `biblatex`

参考文献处理后端。选择 **bibtex** 时使用传统的 **B_{ib}T_EX** 引擎处理参考文献，加载 **natbib** 宏包。选择 **biblatex** 时使用 **biber** 引擎，加载 **biblatex** 宏包。若自动载入参考文献宏包则位于导言区末尾，但在 **hyperref** 宏包之前。

bibstyle = `numerical` | `authoryear` | `other sty`

参考文献格式使用中国参考文献推荐标准 GB/T 7714-2015，对应顺序编码制和著者—出版年制两种风格。若使用其它格式，使用 **B_{ib}T_EX** 处理时需确保正确的 **bst** 文件，使用 **biber** 处理可使用 **biblatex** 宏包支持的参考文献风格名称。

citestyle = \langle 标注风格 \rangle

标注参考文献的风格。后端为 **B_{ib}T_EX** 时可使用 **super**（数字上标标注，默认）或 **numbers**（数字标

注) 等 `natbib` 宏包带有的标注风格, 后端为 `biber` 时可使用 `biblatex` 宏包支持的标注风格。

`datafile` = <数据库列表>

参考文献数据库的文件名列表, 文件列表以花括号包裹并以英文逗号分隔。使用 `biber` 处理时文件名要带 `bib` 扩展名, 若参考文献文件不在当前文件夹则文件名需要包含路径, 即 `path/file.bib`。不设置参考文献数据库名时 `bib` 模块中的选项会失效, 即不自动载入 `natbib` 或 `biblatex` 宏包, 此时需要手动载入相关宏包配置参考文献。

`\printbibliography` [<键值列表>]

两种方式均使用此命令打印参考文献, 不同处理后端此命令的 <键值列表> 不同。使用 `BibTeX` 处理时选项与 [小节 4.2](#) 目录命令的选项相同, 使用 `biber` 处理时常用的选项如下:

`heading` = `bibliography` | `subbibliography` | `bibintoc` | `subbibintoc`

参考文献通常有一个章标题或者节标题, 该选项选择由 `\defbibheading` 定义的标题名。

`title` = <参考文献标题名>

如果标题定义支持的话, 该选项覆盖由 `heading` 选项提供的缺省标题名。

`prenote` = <参考文献前注>

该选项选择由 `\defbibnote` 所定义的前注, 缺省状态下不打印任何前注。

`section` = <整数>

Default: 0

只打印第 <整数> 文节中引用的条目。该参考文献节从 1 开始编号, 所有在 `refsection` 环境外给出的引用标记为第零节。

`type` = <条目类型>

只打印名为 <条目类型> 的文献条目。

`filter` = <过滤器名>

使用由 `\defbibfilter` 定义的 `filter` 来过滤条目。

可能在一些情况下您希望更加灵活地设置参考文献或有更多需求, 下面举一个在导言区手动配置 `biblatex` 宏包并使一个条目输出双语对照格式文献的例子:

例 11

```
\usepackage[style = gb7714-2015]{biblatex}
\addbibresource{reference.bib}
\defbibheading{bibliography}[参考文献] % 定义参考文献标题
{\chapter*{#1}\markboth{#1}{#1}}
\defbibentryset{Zhang2022}{张三2022,ZhangSan2022}
\cite{Zhang2022} % 在正文中引用
```

thmt ⇒ 3.6 定理

定理模块 `thmt` 根据全局选项 `theorem` 的设置可以由 `thmtools` 或 `keytheorems` 宏包提供支持。`keytheorems` 宏包使用 `LATEX3` 编写, 它的定理功能更完善并且提供了 `thmtools` 宏包大部分的功能, 但发布日期较新 (2024 年 9 月) 还未被广泛使用, 下面主要介绍传统的 `thmtools` 宏包的功能。`thmtools` 提供了定理环境和样式声明两个命令, 文档类未预声明任何定理环境, 如果声明定理环境时不指定样式, 则使用默认定理样式 `default`, 可以对它重新定义。

`\declaretheorem` [<键值列表1>] {<环境名列表>} [<键值列表2>]

`thmtools` 的定理环境声明命令。`thmtools` 宏包中此命令实际相当于 `\newtheorem`, 无法强制创建定理环境, 这里修复了这个问题。可选参数的 <键值列表1> 与 <键值列表2> 等效, 常用的选项如下:

`name` = \langle 定理名 \rangle

定理环境的标题名称，默认为英文首字母大写的环境名。

`numbered` = `no` | `yes` | `unless unique`

定理可以被编号、不被编号或者只有在文档中多次出现时才被编号。

`within` = `chapter` | `section` | `empty` | `counter`

定理编号跟随的计数器。当计数器值递增时定理编号会从零开始，通常这是一个章节计数器。

`numberlike` = \langle 计数器 \rangle

定理将使用此 \langle 计数器 \rangle 进行编号，通常这是另一个定理环境的名称。

`qed` = \langle 证毕符号 \rangle

证毕符号。位于定理段落末尾行的版心右边，可以设置成`\qedsymbol`默认是空心方块。

`style` = \langle 定理样式 \rangle

Default: `default`

选项的值为使用`\declaretheoremstyle`命令新定义的定理样式。

使用`\declaretheorem`命令以例 13 中的定理样式 `fancy` 新建一个定理环境 `lemma`，然后用一个基于 `tcolorbox` 宏包的预置彩色盒子 `easyboxi` 增加背景：

例 12

```
\declaretheorem[style = fancy,name = 引理]{lemma}
\addtotcbstyle{tc-easyboxi}{colback = GreenYellow}
\tcolorboxenvironment{lemma}{tc-easyboxi}
```

`\declaretheoremstyle` [\langle 键值列表1 \rangle] [\langle 定理样式 \rangle] [\langle 键值列表2 \rangle]

`thmtools` 的定理样式声明命令，键值列表支持自身以及定理声明命令`\declaretheorem`的所有选项。为命令增加了可选参数 \langle 键值列表2 \rangle ，与 \langle 键值列表1 \rangle 等效，常用的选项如下：

`headfont` = \langle 字体命令 \rangle

Default: `\sffamily`

定理标题部分（包括题注）的字体，包括字体族、字体系列、字体尺寸和字体形状命令。

`notefont` = \langle 字体命令 \rangle

Default: `\sffamily`

定理题注部分的字体，包括字体族、字体系列、字体尺寸和字体形状命令。

`bodyfont` = \langle 字体命令 \rangle

Default: `\normalfont`

定理内容部分的字体，包括字体族、字体系列、字体尺寸和字体形状命令。

`headpunct` = \langle 代码 \rangle

紧随定理标题末尾的标记代码，位于定理第一段内容的间距`postheadspace`之前，可以是字符或插入换行符`\newline`等命令。

`headindent` = \langle 刚性长度 \rangle

Default: `0pt`

定理标题的缩进。默认无缩进，可用`\parindent`命令设置为段落缩进。

`postheadspace` = \langle 刚性长度 \rangle

Default: `1em`

定理标题后与定理第一段内容的间距。

`spaceabove` = \langle 弹性长度 \rangle

Default: `0ex plus .1ex`

定理环境前的间距，使用 `tcolorbox` 的盒子环境包装后被覆盖。

`spacebelow` = \langle 弹性长度 \rangle

Default: `0ex plus .1ex`

定理环境后的间距，使用 `tcolorbox` 的盒子环境包装后被覆盖。

`notebraces` = $\{\langle$ 左括号 $\rangle\}\{\langle$ 右括号 $\rangle\}$

Default: `\{()\}`

设置定理题注两边的括号，两个必选参数分别代表左括号和右括号。

`headformat` = \langle 标题排列 \rangle

定理标题各组成元素的位置排列和格式。定理标题由定理名、编号和题注组成， \langle 标题排列 \rangle 中命令 \backslash NAME、 \backslash NUMBER 和 \backslash NOTE 分别代表定理名、编号和题注。也可以设置值为 `margin` 或 `swapnumber`，它们的编号均在定理名前方，其中 `margin` 的编号在左页边距中。

下面定义一个名为 `fancy` 的定理样式用于 \backslash declaretheorem 的 `style` 选项：

例 13

```
\declaretheoremstyle{fancy}[
  within      = chapter,
  spaceabove  = 6pt,
  spacebelow  = 6pt,
  headfont    = \normalfont\bfseries,
  notefont    = \sffamily,
  notebraces  = {[}{]},
  bodyfont    = \itshape
]
```

\backslash listoftheorems* \langle 键值列表 \rangle

`listtheoremname` = \langle 定理目录标题名 \rangle

\backslash settheoremtocline* \langle 环境名 \rangle { \langle 标题定义 \rangle }

对 `thmttools` 宏包的定理环境目录命令 \backslash listoftheorems 进行了适配调整¹，小节 4.2 和小节 3.11 中的选项机制均可用，但标题定义选项 `tocline` 较为特殊，需用 \backslash settheoremtocline 命令单独设置。在 \langle 标题定义 \rangle 中有三个命令 \backslash thmtname、 \backslash thmtenvname 和 \backslash thmtnote 分别代表定理名、定理环境名和题注用于排版目录条目。

`thmt/loeskip` = \langle 弹性长度 \rangle

Default: 10pt

定理目录中每章第一个条目上方增加的垂直距离。与 \TeX 的表格和图片目录中每章间的条目距离选项 `lotskip` 和 `lofskip` 类似，在 `article` 模式中此选项失效。

\backslash thmtifnote{ \langle 存在题注代码 \rangle }{ \langle 不存在题注代码 \rangle }

如果标题没题注即定理环境不用可选参数，则标题不加入定理目录，这是默认且合理的行为。若需要将无题注的标题加入目录可将 \langle 环境名 \rangle 传给目录风格的 `show` 选项或使用 \backslash settheoremtocline* 命令。是否存在题注需要在 \langle 标题定义 \rangle 中使用 \backslash thmtifnote 命令判定，从而输出不同的内容。

\backslash thmtifname{ \langle 存在编号代码 \rangle }{ \langle 不存在编号代码 \rangle }

在 \langle 标题定义 \rangle 中测试当前定理是否有编号从而输出不同的内容，定理风格选项 `numbered = no` 设置当前定理环境没有编号。

例 14

```
\contentsuse{lemma}{loe}
\RegisterTocName{lemma}
\setlisttheoremstyle{ignoreall,show = lemma}
\settheoremtocline{lemma}
{
  \listnumberline{lemma}
  \thmtname\thmtifnote{[\thmtnote]}{}
}
```

如果使用 `keytheorems` 宏包创建定理环境，它的目录样式同样可以用 `toc` 模块的命令注册后定

1. 也对 `keytheorems` 宏包作为兼容提供的这个命令进行了适配。

制，其中 `tocline` 可用目录命令自带的 `format-code` 选项设置：

例 15

```
\newkeytheorem{corollary}[name = 推论]
\contentsuse{corollary}{thlist}
\RegisterTocName{corollary}
\begin{document}
\keytheoremset{format-code = \numberline{#2}#1#3}
\listofkeytheorems % 在正文中输出定理目录
```

geo ⇒ 3.7 页面尺寸

页面尺寸模块 `geo` 用于设置纸张大小和各种页面边距，它将选项交给页面尺寸宏包 `geometry` 处理，支持通用接口 `\geometry` 的键值列表，预置的页面尺寸与 Word 的默认值接近。下面列举几种 `geometry` 不同键值输入方式的选项，前面有 ☉ 符号的选项为额外附加，其它选项可查阅宏包帮助文档。当用此模块设置纸张规格时，会覆盖全局选项 `paper` 的设置。模块 `geo` 的通用接口命令只能在导言区中使用。

☉ `headruleskip` = 〈页眉线间距〉 Default: 0pt
 ☉ `footruleskip` = 〈页脚线间距〉 Default: 3.5pt
`hdivide` = {〈左边距〉,〈版心宽度〉,〈右边距〉}
`hmargin` = {〈左边距〉,〈右边距〉}
`textheight` = 〈版心高度〉
`nofoot` = true | ☐ false

例 16

```
\ctexset[geo]
{
  headruleskip = 3pt,
  vscale       = 0.9,
  hdivide      = {2cm,15cm,*},
  vcentering
}
```

`geometry` 宏包支持在文档中部改变页面尺寸，提供了新建和保存页面布局的命令。需要注意的是，`fancyhdr` 宏包配置页面标记是以文档开始的页面尺寸为参考，若中途更改页面布局后需用标记偏移命令 `\fancyhfoffset` 重置参考的尺寸，以确保标记位置正确。

`\newgeometry`{〈键值列表〉}

在正文中更改页面布局，〈键值列表〉为 `geometry` 宏包通用接口的各种页面尺寸选项。

`\restoregeometry`

恢复正文开始时的页面布局。

`\savegeometry`{〈布局名〉}

保存当前页面布局名为〈布局名〉。

`\loadgeometry`{〈布局名〉}

载入保存的名为〈布局名〉的页面布局。

例 17

L1

```

\newgeometry{left = 3cm,right = 1cm,bottom = 0.1cm}
\savegeometry{L2}
  L2 (new, saved)
\restoregeometry
  L1 (restored)
\newgeometry{margin = 1cm,includefoot}
  L3 (new)
\loadgeometry{L2}
\fancyhfoffset{0pt}
  L2 (loaded)

```

cap ⇒ 3.8 图表题注

模块 **cap** 用于设置图表等浮动体的题注格式。题注主要由三部分组成：标题标签即图表名与题注编号的组合，标题文本本身通常是内容的简短描述，以及将文本与标签分开的题注分隔符。可以在导言区设置整体格式，也可以在正文使用局部更改格式。模块 **cap** 由 **caption** 宏包提供支持，接口选项与 **caption** 宏包完全一致，但写法为 \langle 条目名 \rangle/\langle 选项 \rangle 。**caption** 宏包提供了多种方法来定制浮动环境（例如 **figure** 和 **table**）中的题注，并可与许多其它宏包配合使用。功能包括旋转题注、侧向题注和连续题注等。下面列举几个题注选项的用法，例 18 中两种设置题注的方法是等效的，更多选项和创建方法可查看宏包文档：

table/labelsep = \langle 标签分隔符 \rangle

figure/format = \langle 题注格式 \rangle

lstlisting/font = \langle 题注字体 \rangle

例 18

```

\ctexset[cap]
{
  table/labelsep = space,
  figure/format  = plain
}
\captionsetup[table]{labelsep = space}
\captionsetup[figure]{format = plain}

```

\RegisterCapOptions{ \langle 条目名列表 \rangle }

给模块 **cap** 注册题注类型为 \langle 条目名 \rangle 的 **caption** 接口选项。已预注册 **figure**、**table** 和 **lstlisting** 三种类型的题注选项。可以搭配 **newfloat** 宏包创建更多类型的浮动环境并设置相应题注。

\caption*{ \langle 目录标题 \rangle }{ \langle 标题 \rangle }

\captionof*{ \langle 条目名 \rangle }{ \langle 目录标题 \rangle }{ \langle 标题 \rangle }

\bicapition*{ \langle 目录中文标题 \rangle }{ \langle 中文标题 \rangle }{ \langle 目录英文标题 \rangle }{ \langle 英文标题 \rangle }

题注命令 **\caption** 用于在图表上下方插入题注，需要在浮动环境 **figure** 和 **table** 中使用。在浮动环境外可以使用 **\captionof** 命令，第一个参数指定条目名。**\bicaption** 命令可以实现中英文双语题注，需要在导言区载入 **bicaption** 宏包，具体用法见宏包文档。

hdr ⇒ 3.9 页眉页脚

模块 **hdr** 设置的命令由 **fancyhdr** 宏包提供支持。标题标记由标签和标题组成，选项中参数 #1 代表当前章、节或小节的标题。在 **book** 模式时使用标题标记选项 **chapter-mark** 和 **section-mark**，

在 `article` 模式时使用 `section-mark` 和 `subsection-mark` 选项，这与标准文档类的行为一致。重新设置标题标记选项时标记大小写选项 `uppercase` 会失效。

`chapter-mark` = \langle 标题标记 \rangle

`chapter` 级别标题标记的内容，包含了标签选项 `chapter-label` 和章标题 #1。

`section-mark` = \langle 标题标记 \rangle

`section` 级别标题标记的内容，包含了标签选项 `section-label` 和节标题 #1。

`subsection-mark` = \langle 标题标记 \rangle

`subsection` 级别标题标记的内容，包含了标签选项 `subsection-label` 和小节标题 #1。

`chapter-label` = $[\langle$ 编号输出命令 $\rangle][\langle$ 距离命令 $\rangle]$ Default: `\hspace{1em}`

`chapter` 级别标题标记的标签（标题有编号时）。比标题标记选项更常用，可选参数是对应章编号计数器值的输出命令，必选参数是标题标签与内容之间的距离命令：

例 19

```
\ebhdrset
{
  chapter-label = [\thechapter]{\hspace{1em}},
  section-mark  = \CTEXifname{\CTEXthesection---}{\}#1
}
```

`section-label` = $[\langle$ 编号输出命令 $\rangle][\langle$ 距离命令 $\rangle]$ Default: `\hspace{0.5em}`

`section` 级别标题标记的标签，用法与 `chapter-label` 类似。

`subsection-label` = $[\langle$ 编号输出命令 $\rangle][\langle$ 距离命令 $\rangle]$ Default: `\hspace{0.5em}`

`subsection` 级别标题标记的标签，用法与 `chapter-label` 类似。

`headrule` = \langle 横线命令 \rangle

绘制页眉横线的选项，可以使用 `\hrule` 或例 22 所述 `\markrule` 等画线命令来绘制横线。

`footrule` = \langle 横线命令 \rangle

绘制页脚横线的选项，可以使用 `\hrule` 或例 22 所述 `\markrule` 等画线命令来绘制横线。

`footnoterule` = \langle 横线命令 \rangle

绘制脚注横线的选项，可以使用 `\hrule` 或例 22 所述 `\markrule` 等画线命令来绘制横线。

`headrulewd` = \langle 刚性长度 \rangle Default: 0.5pt

页眉横线默认的粗细，使用 `headrule` 选项绘制横线后失效。

`footnoterulewd` = \langle 刚性长度 \rangle Default: 0.5pt

脚注横线默认的粗细，使用 `footnoterule` 选项绘制横线后失效。

`\fancypagestyle` $\{\langle$ 风格名 $\rangle\}[\langle$ 基础风格 $\rangle][\langle$ 风格定义 $\rangle]$

定义名为 \langle 风格名 \rangle 的页面风格。 \langle 风格定义 \rangle 中可包含设置页眉页脚不同位置格式的命令，如果为空则等于 `fancy` 风格。如果存在 \langle 基础风格 \rangle ，则在它的基础上进行定义。

例 20

```
\fancypagestyle{plain}
{
  \fancyhf{}
  \fancyfoot[C]{\textbf{---\thepage---}}
  \ebhdrset{headrule = {}}
}
```


`\pagestyle{<风格名>}`

Default: fancy

设置 <风格名> 为当前及之后页的页面风格。

`\fancyhead[<位置>]{<内容>}`

`\fancyfoot[<位置>]{<内容>}`

为了确保页面显示正确，使用 `\fancyhead` 和 `\fancyfoot` 等命令设置页眉页脚时需要在页边距的命令之后。默认设置的是一个名为 fancy 的页面风格。fancyhdr 将页眉页脚分成了左中右和奇数页 (odd)、偶数页 (even) 共 12 个部分，可选参数中 E 和 O 代表偶数和奇数，L、C 和 R 代表左、中 and 右部分。单面打印时，含有偶数位置的命令无效。

`\markboth{<左页标题标记>}{<右页标题标记>}`

`\markright{<右页标题标记>}`

`\markdouble{<双页标题标记>}`

这三个命令用于手动设置当前页标题标记，特殊情况下可以使用它们定义与标题不同的页眉页脚标记。其中 `\markdouble` 根据是否是双面打印设置相应标记。如果是单面打印，则 <左页标题标记> 无效。标题标记选项 <级别>-mark 定义的是它们的参数，左右页标记规则已经进行了优化调整，若需更改可以重定义 `\<级别>mark` 命令。

例 21

```
\makeatletter
\renewcommand{\sectionmark}[1]{\markright{\eb@mark@section@whole{#1}}}
\makeatother
```

`\markrule[<长度>][<颜色名>]{<线宽>}[<上间距>|<下间距>]`

实线命令 `\markrule` 既可在 <横线命令> 中用于绘制页眉页脚横线，也可以在正文中使用。<长度> 默认为版心的宽度，上下间距为横线与上下方部件的间距，带有长度单位，上下间距之间用符号 | 分开，若只有一部分则表示 <上间距>。下面举一个设置页眉内容的例子：

例 22

```
\ebhdrset{headrule = \markrule(DarkOrchid){0.5pt}[2pt|2pt]}
\fancyhead[ER]{\FirstMark{2e-left}} % 来自ltmarks的标记机制
\fancyhead[OL]{\rightmark}
\fancyhead[EL,OR]{\textbf{\thepage}}
```

hdr ⇒ 3.10 脚注

模块 `hdr` 还可以灵活调整脚注编号和内容的格式。格式主要包括脚注标记的编号类型、字体、附加符号和对齐方式等，以及脚注内容的首行缩进、行距和字体等。基本思想是提供常规和自定义标记 (fncustom) 两种脚注，常规脚注的标记分为正文 (fnmark) 和内容 (fntext) 标记两种，且两者编号类型分离，自定义脚注的标记与其共享格式。目前脚注模块基本兼容 `footmisc` 宏包。

`footnotetype = plain|hang|flush|bottom`

脚注的类型。悬挂缩进 `hang` 是默认值，`plain` 是原始的类型，`flush` 使脚注内容标记位于脚注首行边缘外侧并右对齐。当存在浮动环境排版时，脚注可能位于页面中间，`bottom` 使脚注始终位于页面底端。其中 `bottom` 是复合选项，可以与前三种类型重复或复合使用：

例 23

```
\ebhdrset{footnotetype = plain,footnotetype = bottom}
\ebhdrset{footnotetype = {hang,bottom}}
```

New
2024cx**\PushFntBottom**

默认情况下当具有 **b** 参数的底部浮动体与脚注同时存在时，脚注将位于浮动体上部。此命令可使脚注始终位于页面底部，与 **footnotetype = bottom** 作用相同，只能在导言区全局使用。

fnfirstindent = 〈刚性长度〉 Default: 1em

每个脚注首段的首行缩进（悬挂缩进 **hang** 为整段缩进），等于长度 **\fnfirstindent** 的值。

fnafterindent = 〈刚性长度〉 Default: 2em

每个脚注第二段及之后段落的首行缩进，等于长度 **\fnafterindent** 的值。

fnparskip = 〈弹性长度〉 Default: 0ex plus .1ex

不同脚注自身的段落间距，脚注内容文本分段时生效，等于长度 **\fnparskip** 的值。

footnotesep = 〈刚性长度〉

不同脚注之间的距离，默认情况下它自适应 **space** 模块脚注行距选项 **fnspread** 的值。

fn-text-pos = **normal** | **super**

设置脚注内容标记的位置，正常文本或是上标形式。

fnmark-num = **plain** | **pifont** | **pifont*** | **pisans** | **pisans*** | **short num**

脚注正文标记编号的类型。默认为原生样式 **plain**，还可以改为 **pifont** 的各个版本，从左往右依次为普通版、阴文衬线版、阳文无衬线版和阴文无衬线版，缺点是它们的编号仅包括 1-10。

短编号的参数与列表环境类似，有 1、A、a、I 和 i，除此之外还增加了中文数字 **c** 和符号 **s** 格式。原生样式 **plain** 等同于单独使用数字 1 参数。这个选项可以用在正文中改变其后脚注编号类型。

\footref{〈脚注标签名〉}

在当前位置对自动编号的脚注进行引用，与被引用的正文标记共享一个编号与格式。

例 24

```
\ebhdrset{fnmark-num = 1} % 等于编号类型plain
\ebhdrset{fnmark-num = (a)}
```

\SetFntSymsList{〈符号表名〉}{〈符号数量〉}{〈符号列表〉}**\UseFntSymsList**{〈符号表名〉}

这两个命令用于在脚注短编号标记为 **s** 自定义符号型时设置符号列表。**\SetFntSymsList** 命令用于定义名为〈符号表名〉的符号列表，**\UseFntSymsList** 命令用于启用已定义的符号列表。默认脚注符号列表是空的，注意使用符号型标记的脚注时计数器 **footnote** 的值仍然递增，以及定义新列表时〈符号列表〉的写法和命令顺序：

例 25

```
\SetFntSymsList{fancy}{2}{\{1\}{*}{2}{\P}}
\UseFntSymsList{fancy}
\ebhdrset{fnmark-num = s}
```

fn-text-num = **plain** | **pifont** | **pifont*** | **pisans** | **pisans*** | **short num**

脚注内容标记编号的类型。用法与 **fnmark-num** 一样，所以脚注内容与正文标记编号类型可以不同。

fnmarktext-num = **plain** | **pifont** | **pifont*** | **pisans** | **pisans*** | **short num**

同时设定脚注正文标记编号 **fnmark-num** 与脚注内容标记编号 **fn-text-num** 的类型。目前脚注编号类型选项对于迷你页环境 **minipage** 无效，其中的脚注保持默认英文字母的编号类型。

fnmark-vmove = **fixed length** | **match** Default: 0pt

脚注正文标记的垂直偏移。可以设为刚性长度，或用 `match` 匹配 `spvmove` 全局的值。

`fnmark-hmove = fixed length|match` Default: 0pt

脚注正文标记的水平偏移。可以设为刚性长度，或用 `match` 匹配 `sphmove` 全局的值。

`fntext-vmove = fixed length|match` Default: 0pt

脚注内容标记的垂直偏移。可以设为刚性长度，或用 `match` 匹配 `spvmove` 全局的值。

`fnmarktext-format = format code|match`

默认脚注标记的格式，选项 `match` 可以匹配 `spcmd` 全局的值。一般配合 `fnmarktext-num = plain` 原生编号形式使用，例如用 `circledtext` 宏包的带圈字符命令 `\circledtext` 给脚注标记添加圆圈：

例 26

```
\usepackage{circledtext}
\circledtextset{resize = real,width = 0.5em}
\ebhdrset
{
  fntext-pos      = normal,
  fnmarktext-format = \circledtext
}
```

`\defupfntmark{<正文标记格式>}`

定义脚注正文标记的格式，与正文标记编号命令 `\theupfootnote` 配合使用。

`\defdownfntmark{<内容标记格式>}`

定义脚注内容标记的格式，与内容标记编号命令 `\thedownfootnote` 配合使用。

`\deffntpara{<段落文本格式>}`

定义脚注段落文本的格式的指令。脚注段落文本由脚注内容标记和文本两部分组成，段落格式定义中参数 #1 代表内容文本。

`\notminipage{<普通页内容>}{<迷你页内容>}`

用于在定义脚注标记的命令中判断当前脚注是否处于迷你页而输出不同的内容。

`\theupfootnote`

保存正文标记编号类型选项 `fnmark-num` 指定的正文标记。

`\thedownfootnote`

保存内容标记编号类型选项 `fntext-num` 指定的内容标记。

例 27

```
\defupfntmark{\super[vmove = 2pt]{\theupfootnote}}
\defdownfntmark{\notminipage{\textcolor{blue}{\thedownfootnote}}{}}
\deffntpara{\makebox[2em][r]{\thefootnote}#1}
```

`fncustom-format = format code|match`

单独设定自定义脚注标记的格式，选项 `match` 可以匹配 `spcmd` 全局的值。自定义脚注内容和正文标记的命令与用法可以见下方和例 28 的介绍。

`\Footnote[<内容标记>]{<正文标记>}{<脚注文本>}`

`\Footnotetext{<内容标记>}{<脚注文本>}`

`\Footnotemark{<正文标记>}`

这些是自定义标记且不影响正常编号的脚注命令，用法与它们对应首字母小写的命令类似。`<内容标记>`

默认与〈正文标记〉一致，这也是最常见的情况，也可以定义为不同符号或格式。如果当前标记为空，那么将会使用上一个脚注标记。

例 28

```
\Footnote{*}{这是可以自定义标记的脚注。}
```

```
fnboth-format = format code|match
```

同时设定默认脚注标记 `fnmarktext-format` 和自定义脚注标记 `fncustom-format` 的格式。两种脚注标记匹配上标格式 `spcmd` 的值是独立的，使用 `match` 则可以同时匹配。会覆盖 `font` 模块脚注字体选项 `footnote` 相同类型的命令，最后一个格式命令可以带有一个参数接受脚注标记。

```
fnpara-format = 〈格式代码〉
```

脚注内容文本的格式。会覆盖 `font` 模块脚注字体选项 `footnote` 相同类型的命令，最后一个格式命令可以带有一个参数接受脚注内容文本。

```
fntext-code = {〈前代码〉,〈后代码〉}
```

可以用〈前代码〉与〈后代码〉额外增加脚注内容标记前后的代码。也可以只有前部分，〈前代码〉中最后一个格式命令可以带有一个参数接受脚注内容标记。由于脚注标记处于水平模式，可以配合盒子命令设置脚注内容标记的对齐方式，或在标记与脚注首行间插入一点距离。

例 29

```
\ebhdrset
{
  footnotetype = hang,
  fntext-code  = {\hspace{1em}\llap,\enskip}
}
```

toc ⇒ 3.11 目录

模块 `toc` 将使用 `titletoc` 宏包处理目录样式。默认包括 `part`→`subparagraph`、`figure`、`table` 和 `lstlisting` 条目名的键路径。选项前带有 ☉ 符号表示支持所有相应条目名的路径，其它选项则处于 `toc` 键路径下。目录选项均支持标题编号判定命令 `\CTEXifname`²。目前支持多数学术论文目录样式，更复杂的样式可以查看 `titletoc` 的相关用法。

☉ `tocformat` = 〈条目格式〉

目录中各类型条目的格式。可以设置条目字体和加入垂直间距等，垂直间距最好使用防止垂直间距重合的 `\addvspace` 命令。如果是段落型目录，格式代码仅在段落中当前级别条目前执行一次。也可以用 `\CTEXifname` 命令或 `tocformat*` 选项给有无编号的标题分别设置格式。

☉ `tocformat+` = 〈条目格式〉

与选项 `tocformat` 用法一样，只是在已有的格式命令后附加代码。

☉ `tocformat*` = 〈条目格式〉

没有编号/名字标题的附加格式代码。目录中判断标题是否有编号/名字的依据为：如果在 `tocline` 选项中使用了 `\numberline` 命令输出标签，则判定为有编号/名字。格式最后的命令可带有一个参数，用于接受标题内容。

☉ `tocindent` = 〈刚性长度〉

目录条目左边的缩进距离，通常标题级别越低缩进越大。如果是段落型目录，仅在当前条目级别为段落中最高时有效。

2. 此功能需要 2023/06/01 以后的 L^AT_EX 内核支持。

☉ **tocrule** = <引导线命令>

目录条目的引导线和页码。引导线可以使用基于 `\xleaders` 命令的 `\tocrule` 绘制（同时输出页码），如果是一般不需要引导线的段落型目录，可以使用 `\thecontentspage` 命令输出页码。

`\tocrule*-[<引导点间距>](<放大因子>){<引导点>}[<页码格式>]`

生成目录引导线的命令，需要在引导线选项 `tocrule` 中使用。带有星号的命令与 `dotalign = false` 效果相同。命令带 `-` 号参数不输出目录页码，此时无格式的页码被保存在 `\thecontentspage` 命令中，可以配合盒子命令设置不同的页码左右间距和对齐方式。`<放大因子>` 为 `<引导点>` 的放大倍数。`<页码格式>` 中最后一个命令可以带有一个参数用于接受页码。

☉ **toclabel** = <标题标签定义>

目录中各类型条目标题标签的定义，可与 `tocline` 选项联用设置标题。其中参数 `#1` 代表标签的编号 `\thecontentslabel`，参数 `#2` 代表<条目名>。定义最后的命令可带有一个参数，用于接受标题内容，一些特殊情况例如需要将标题内容放在标签前，可以通过定义命令实现：

例 30

```
\newcommand{\seqentry}[2]{#2-#1} % #2为标题内容
\ctexset{chapter/toclabel = \seqentry{#1}}
```

`\hytoclink{<目录条目元素>}`

给目录条目元素增加超链接，主要用于 `toclabel` 选项中的标签。使用 `titletoc` 宏包时条目标题的标签超链接会缺失，这个命令可修复此问题。

☉ **tocafter** = <距离命令>

Default: `\hspace{1em}`

目录条目标签与标题间的代码，默认为插入 `1em` 横向距离。

☉ **tocbelow** = <距离命令>

在目录条目下方执行的代码，可以插入垂直距离等命令。

☉ **tocline** = <标题定义>

这个选项在章节标题基础上扩展了浮动环境的题注标题在目录文件中的定义。章节和题注标题选项中参数 `#1` 代表标题编号 `\the<计数器>` 对应的计数器名称，参数 `#2` 则代表相应标题内容。

例 31

```
\ctexset
{
  chapter
  {
    tocformat      = \CTEXifname{\addvspace{8pt}}{\large}{},
    tocindent      = \CTEXifname{2em}{1.5em},
    tocrule        = \tocrule[0.6pc](1.1){.}{\zihao{5}},
    tocline        = \CTEXnumberline{#1}#2
  },
  section/toclabel = \contentspush{\hytoclink{#1}\hspace{0.5em}},
  section/tocrule  = \tocrule-{}\makebox[1cm]{\rlap{\thecontentspage}},
  toc/pagenumwd    = 0pt
}
```

`\RegisterTocName*{<条目名列表>}`

给<条目名列表>中的每个<条目名>标题注册用于设置目录样式的选项，即选项前有符号 ☉ 表示需要注册后使用。前面提到的条目类型均已注册，新类型的条目注册后默认格式与图表条目相同。这个

New
2024as

New
2024eb

Changed
2024bo

命令可以重复使用，会重置目录条目为默认格式。带有星号参数的命令使注册目录的条目形成一个段落，此时多出下面三个选项可用：

☉ **tocbegin** = 〈开始代码〉

New 2024as 当前级别目录条目第一个标题前的代码，仅在存在更高级别的条目时有效。

☉ **tocsep** = 〈中间代码〉

Default: { . — }

New 2024as 当前级别目录条目每个标题之间的代码。

☉ **tocend** = 〈结束代码〉

Default: { . }

New 2024as 当前级别目录条目最后一个标题后的代码。

\contentsuse[〈目录层次〉]{〈条目名〉}{〈扩展名〉}

Changed 2024ce 激活条目类型为〈条目名〉的目录定制功能，目录文件后缀为〈扩展名〉，若〈扩展名〉为 **toc** 则条目会自动激活。可选参数〈目录层次〉为条目在目录中的层次深度，默认值为 -1000，当它大于目录深度计数器 **tocdepth** 时不显示。图表等标题的目录条目无深度因此为默认值，章节标题 **part** → **subparagraph** 的深度则从 -1 到 5 依次递增。需要在激活条目后再使用命令 **\RegisterTocName** 注册选项。

\DeclareFloatList{〈条目名〉}{〈扩展名〉}

声明一种条目类型的目录命令 **\listof**〈条目名〉**s**，并且支持 [小节 4.2](#) 中图表目录命令的机制。使用这个命令要确保将条目加入目录的机制已经存在，例如搭配 **newfloat** 宏包创建新的浮动环境。选项前有实心符号 ●（仍包含空心符号 ☉ 选项的性质）表示在声明后可用。下面举一个声明算法环境和对应题注目录的例子：

例 32

```
\usepackage{newfloat}
\DeclareFloatingEnvironment[
  fileext    = loa,
  listname   = 算法清单,
  name       = 算法
]{algorithm}
\DeclareFloatList{algorithm}{loa}
\begin{document}
\listofalgorithms[multoc] % 在正文中输出双栏算法目录
```

\AssignCaptionExt{〈条目名〉}{〈扩展名〉}

给条目类型名称为〈条目名〉的标题分配目录文件〈扩展名〉，注意〈扩展名〉需要被定义在 **\ext@**〈条目名〉命令中才有效（**caption** 宏包的默认定义形式）。可以使不同的〈条目名〉共享一个目录：

例 33

```
\AssignCaptionExt{table}{lof}
\newcommand{\listoffiguretables}{\listoffigures} % 同时输出图表目录的命令
```

\AssignPartialExt{〈扩展名1〉}{〈扩展名2〉}

New 2024bk 给目录文件扩展名为〈扩展名1〉的目录分配扩展名为〈扩展名2〉的局部目录³，预注册的条目均无需再分配。用于搭配题注标题 **\caption** 命令和 **titletoc** 宏包的局部题注目录命令。

例 34

```
\AssignPartialExt{loa}{pla}
\startlist[fancy]{loa}
```

3. 此功能需要 2023/06/01 以后的 L^AT_EX 内核支持。

```
\captionof{algorithm}{算法标题}
\stoplist[fancy]{loa}
\printlist[fancy]{loa}{}{}
```

\SetTocStyle{<条目名>}{<风格名>}{<键值列表>}

定义条目类型名称为<风格名>的目录风格。使用 **\ctexset** 命令设置目录样式时，除了条目标题定义选项 **tocline**，其它选项只有在目录输出命令前设置才有效，定义完目录风格后用 **\UseTocStyle** 命令可以在正文中任意位置生效。如果是没有⊙符号的选项，<条目名>设置为 **toc**。

\UseTocStyle{<条目名>}{<风格名>}{<扩展名>}

使用对应条目类型名称为<风格名>的目录风格，随后标题在目录中的格式会改变。本命令导言区和正文区均可使用，第三个必选参数还需要指定条目标题加入目录文件的<扩展名>。

例 35

```
\SetTocStyle{chapter}{fancy}{tocformat = \color{red}\zihao{3}}
\UseTocStyle{chapter}{fancy}{toc}
```

\listnumberline{<计数器名>}

在图表等浮动环境的题注目录中插入条目标签，默认标签的名字可以由格式命令 **\labelformat** 设置。其中<计数器名>一般与<条目名>相同，可以用参数 **#1** 指定或用对应计数器名称显式指定。

例 36

```
\ctexset
{
  lstlisting/tocline = \listnumberline{#1}#2,
  figure/tocline     = \protect\numberline{图\thefigure\hspace{1em}}#2
}
```

\contentsmargin[<修正距离>]{<右边距>}

设置目录中引导线到版心右边的距离，用于条目页码的精细调整。当设计条目页码对齐方式或数字较大时，可以设置适当的<右边距>。<修正距离>是在过长而换行的标题右侧<右边距>基础上附加的距离。此命令可在导言区全局使用或在 **tocformat** 选项给特定条目类型单独使用。

hang = ☐true | false

悬挂缩进选项 **hang** 使目录条目为悬挂缩进形式。缩进距离为条目标签及其与标题内容间距的长度和。通常情况下会统一设置条目标题是否悬挂缩进，因此这个选项被设计为对所有类型条目生效。

lolskip = <弹性长度>

Default: 10pt

代码目录中每章第一个条目上方增加的垂直距离。与 \CTEX 的表格和图片目录中每章间的条目距离选项 **lotskip** 和 **lofskip** 类似，在 **article** 模式中此选项失效。

⊙ **between** = <格式代码>

目录标题与条目内容间的局部钩子（作用于标题下方间距之后），如果是 **book** 模式则默认插入 -10pt 垂直距离抵消首行目录的额外垂直间距，这是由于图表和由 **newfloat** 宏包所创建环境的目录默认每使用一次章标题在题注目录中插入 10pt 距离。除了条目名的键路径外，此选项也可用于 **toc** 路径中，表示章节标题目录标题代码钩子，如果是 **book** 模式则默认插入 -1pc 垂直距离。

bibetween = <格式代码>

与选项 **between** 类似，是英文章节标题目录（如果启用）标题代码钩子。

dotalign = ☐true | false

将目录页码装进一个右对齐的水平盒子里。这样可以使目录引导线右端对齐并与页码保持一定间距，否则会跟页码相连。

`pagenumwd` = \langle 刚性长度 \rangle Default: 1.55em

设置开启 `dotalign` 选项时页码水平盒子的宽度。

`tocafter-every` = \langle 距离命令 \rangle Default: `\hspace{1em}`

统一设置所有类型条目编号与标题间的代码。

`tocindent-every` = \langle 刚性长度 \rangle

统一设置所有类型条目的左边缩进。

`tocrule-every` = \langle 引导线命令 \rangle

统一设置所有类型的目录引导线样式。

`\ExcludeEveryTocName`{ \langle 条目名列表 \rangle }

用于指定带有 `every` 后缀选项排除的条目类型。

New
2024bn

4 环境和命令

4.1 封面和文区

`\maketitle`[\langle 键值列表 \rangle]

重新改写了默认的封面标题命令，在保证原生风格的前提下使用 `xtemplate` 宏包将封面抽象为由实例组成的模板。并为命令增加了可选参数，选项的含义与标准文档类一致：

`title` = \langle 文档标题 \rangle

设置文档的标题，可以包含格式命令。如果没有设置标题而输出封面会报错。

`author` = \langle 文档作者 \rangle

设置文档的作者，可以包含格式命令。多位作者之间可以用 `\and` 命令分隔，每个作者后可用 `\thanks` 命令添加致谢脚注。

`date` = \langle 文档日期 \rangle Default: `\today`

设置文档的日期，可以包含格式命令。默认的文档日期为当前日期 `\today`。

`\DeclareInstance`{easybook}{ \langle 位置 \rangle }{cover}{ \langle 键值列表 \rangle }

封面模板可用 `xtemplate` 提供的实例声明命令进行自由定制，例 37 举例说明了定制页面中部的内容。模板由页面上、中和下三个位置的实例组成，它们的 \langle 位置 \rangle 名字分别为 `top`、`middle` 和 `bottom`。每个实例的属性可用 \langle 键值列表 \rangle 修改，包含的选项如下：

`content` = \langle 实例内容 \rangle

设置实例内容，可以包含任何记号列表。通常为格式和排版内容的命令，作用范围为局部。

`beforecode` = \langle 实例前代码 \rangle

设置实例内容前的代码，可以包含任何记号列表。通常为换页和垂直距离的命令。

`aftercode` = \langle 实例后代码 \rangle

与 `before` 选项类似，设置实例内容后的代码。

`align` = l| \boxed{c} |r|n

设置实例内容的对齐方式，分别为左、中、右和自然对齐，自然对齐即正常的段落文本。

例 37

```
\DeclareInstance{easybook}{middle}{cover}
```

New
2024dh

```

{
  content      =
  {
    \thispagestyle{empty}
    \begin{tblr}
    {
      columns      = c,
      hline{2-4}   = {2}{0.5pt,solid},
      column{1}    = {rightsep = 0pt}
    }
    标题: & 一个文档标题 \\
    作者: & 张三 \\
    日期: & \today
    \end{tblr}
  },
  beforecode = \null\vspace{3cm},
  aftercode  = \clearpage
}

```

\frontmatter* [⟨编号格式⟩]

Default: Roman

前言区命令。使部分和章标题不编号，带有星号的变体命令不使用 `\cleardoublepage` 命令。页码 ⟨编号格式⟩ 可设置为 arabic、Alph、alph、Roman、roman 和 chinese 形式。

\mainmatter*

主文区命令。章标题正常编号且页码为从 1 开始计数的阿拉伯数字，带有星号的变体命令不使用 `\cleardoublepage` 清双页命令。

\backmatter

后记区命令。页码格式与编号保持不变，章标题不编号。

\cleardoublepage [⟨页面风格⟩]

Default: empty

清双页命令可以确保下一页文本总在奇数页开始，在奇数页使用它会产生空白偶数页。文区命令 `\frontmatter` 和 `\mainmatter` 会用到清双页命令，在启用文类选项 `openright` 时标题命令 `\part` 和 `\chapter` 也会默认用到清双页命令。

\blankpagestyle {⟨页面风格⟩}

多出空白页的 ⟨页面风格⟩ 用空白页面风格命令 `\blankpagestyle` 全局指定，或使用清双页命令的可选参数局部指定。空白页默认的风格为 `empty`，即空的页眉页脚。

4.2 目录命令

`easybook` 改进了标准文档类中定制性较低的章节标题和图表标题目录命令。现在目录命令带有可选参数用于设置标题格式和栏数等，并且与标准文档类不同的是目录命令可在正文中重复使用⁴。更多目录命令的创建和样式设置可参考小节 3.11。

\tableofcontents* [⟨键值列表⟩] (⟨代码钩子⟩)

章节标题目录命令，支持中英文双语标题目录输出。带有星号参数不输出目录条目上方的标题。⟨代码钩子⟩ 是当输出方案 `plan = both` 时中文与英文目录之间执行的代码，可以插入设置目录样式和换页等命令。

4. 此功能需要 2023/06/01 以后的 L^AT_EX 内核支持。

`plan = [cn|bi|both]`

章节标题目录的输出方案。`cn` 只输出中文目录，`bi` 只输出英文目录。`both` 同时连续输出中文和英文目录，需要与双语目录标题命令 `\bichapter` 等或双语标题选项 `title` 配合使用。

`bicontentsname = <英文目录标题>`

Default: Contents

英文目录的标题，为通用接口命令 `\ctexset` 的选项。

`\listoffigures*{<键值列表>}`

输出图片目录命令。带有星号参数的命令不输出目录标题。

`\listoftables*{<键值列表>}`

输出表格目录命令。带有星号参数的命令不输出目录标题。

`\listoflstlistings*{<键值列表>}`

输出代码目录命令。带有星号参数的命令不输出目录标题。

如果需要局部而不是全文范围的目录或其它类型的目录，可以查看[小节 3.11](#) 和 `titletoc` 宏包的相关示例。如果需要双语图表目录，可以查看 `bicaption` 宏包的相关示例。上面目录命令的可选参数改进了原本单一的功能，也提供了配置命令 `\SetTocOuter` 在目录命令外设置选项默认值。

`\SetTocOuter{<键值列表>}`

统一设置上述所有目录可选参数的默认值⁵，其中 `<键值列表>` 的选项如下：

`level = <标题命令>`

设定输出目录标题的命令。如果使用 `title` 选项设置了双语标题则需要配合双语标题命令。

例 38

```
\tableofcontents[level = \chapter*]
\listoffigures[level = \bichapter,title = {图片索引,List of Figures}]
```

`chapter`

设置目录标题级别为 `chapter`，与 `level = \bichapter*` 等效。

`section`

设置目录标题级别为 `section`，与 `level = \bisection*` 等效。

`none`

不输出目录标题，与带星号参数的目录命令等效。

`multoc = <整数>`

Default: 2

设置目录的排版栏数。`<整数>` 代表目录栏数，不指定栏数的值则默认为双栏。

`title = {<中文目录标题>,<英文目录标题>}`

用于更改目录的双语标题。标题之间用一个英文逗号分开，对主目录来说前部分为中文目录标题，后部分为英文目录标题，对图表和代码目录来说后部分则是加入英文目录的条目。也可以只有一部分，表示中文目录的标题。

`bilist = mix|new`

确定章节标题目录中双语标题条目的输出排列方式。选项 `mix` 使每个英文标题条目在中文目录中跟随对应的中文标题输出，`new` 则在新建的英文目录中输出英文标题，此新目录的扩展名为 `tec`。这是个全局选项，在所有目录的可选参数中使用均可生效，但推荐在章节标题目录中使用。

`\ebifbitoc{<英文目录代码>}{<中文目录代码>}`

启用混合双语目录条目即 `bilist = mix` 时，在设置目录样式时可以配合 `\ebifbitoc` 命令判断当前

5. 由于目录命令通常只在文中使用一次，因此不设计为可以分别指定不同目录的参数。

条目是否为英文，从而输出不同的代码。

例 39

```
\SetTocStyle{section}{fancy}
{
  tocformat = \ebifbitoc{\color{red}}{\small},
  tocindent = \ebifbitoc{1em}{2em}
}
\UseTocStyle{section}{fancy}{toc}
\tableofcontents[bilist = mix]
```

除了上述选项还支持 C_T_EX 键值列表所有路径的章节标题样式选项，下面仅列举几个选项例子。需要注意的是标题样式选项根据 `level` 选项中的最后一个标题命令判断键路径，所以这个标题命令名必须存在于 C_T_EX 键值列表路径中，或带有 `bi` 前缀表示双语标题。

`format` = 〈目录标题格式〉

设置目录标题的格式。

例 40

```
\tableofcontents[
  level   = \clearpage\bisection*,
  format  = \centering\sffamily\Large
]
```

`aftername` = 〈标签后代码〉

目录标题的标签与内容之间插入的代码。

`aftertitle` = 〈标题后代码〉

目录标题内容之后插入的代码。

`break` = 〈格式命令〉

控制章节标题与之前正文的分隔关系。一般用于设置是否在标题之前分页或者设置行间罚点。

`pagestyle` = 〈页面风格〉

Default: empty

目录标题所在页的页面格式，仅当目录以 `chapter` 级别输出时有效。

4.3 双语章节标题和新建标题

双语目录标题命令将中文标题正常输出，并使中英文标题加入双语目录。目前设置了常用的三个级别，这个功能是为国内一些高校的学位论文模板的要求设计。类似于标准文档类，这三个标题命令分别具有独立的计数器⁶，并对应 `\the`〈计数器〉的值，可以用于 `number` 选项更改英文目录中标题的编号格式。

`\bichapter*`〔〈目录中文标题〉〕{〈中文标题〉}{〈目录英文标题〉}

`chapter` 章级别的双语目录标题命令，具有独立的 `bichapter` 计数器。

`\bisection*`〔〈目录中文标题〉〕{〈中文标题〉}{〈目录英文标题〉}

`section` 节级别的双语目录标题命令，具有独立的 `bisection` 计数器。

`\bisubsection*`〔〈目录中文标题〉〕{〈中文标题〉}{〈目录英文标题〉}

`subsection` 小节级别的双语目录标题命令，具有独立的 `bisubsection` 计数器。

6. 默认情况下与普通标题命令共享使用计数器。

命令的星号参数使正文中的标题不编号，但是仍然加入双语目录，若英文标题为空那么仅加入中文目录，这一特点与 `\chapter*` 等命令的行为不太一样，在一些情况下会用到。可选参数是目录中的中文条目，默认与中文标题一致，正文标题过长时可用它设置在目录中显示的标题。默认情况下双语目录条目的格式是一致的，共享使用 `toc` 模块的设置。提供了两个选项 `biname` 和 `binumber` 修改英文目录章标题条目的名字及编号：

`chapter/biname` = {<英文目录前名字>,<英文目录后名字>} Default: `Chapter\space`

更改英文目录 `chapter` 级别的标题条目编号前后名字，类似于 \TeX 的 `name` 选项。

`chapter/binumber` = <英文目录编号输出命令> Default: `\arabic{chapter}`

更改英文目录 `chapter` 级别的标题条目编号输出格式，类似于 \TeX 的 `number` 选项。下面两种设置英文目录中章标题编号的方式等效：

例 41

```
\ctexset{chapter/binumber = \arabic{chapter}}
\BeforeAddtoBitoc{\ctexset{chapter/nmuber = \arabic{chapter}}}
```

`\appendix` [<前名字>,<后名字>] (<编号输出命令>)

附录起始命令，其后的章节标题变为附录形式。<前名字> 为英文目录中附录标题的前名字，同理 <后名字> 为英文目录中附录标题的后名字。前后名字之间用一个半角逗号分开，也可以只有一部分，表示英文目录中附录标题前名字。<编号输出命令> 为附录标题计数器在英文目录中的编号命令。注意这两个参数仅对 `chapter` 级别的标题生效。

`\BeforeAddtoBitoc` *{<格式代码>}

英文目录中标题条目的格式钩子，<格式代码> 在英文标题加入目录前执行，作用范围为局部。可以插入条目标题定义等命令临时改变英文目录中条目的格式，带有星号参数的命令会重置之前的代码。这个命令设置英文目录条目的标题格式具有最高优先级，放置在导言区或 `\bichapter` 等双语标题命令前生效。举一个在英文目录节标题条目编号前添加 `Section` 字符串的例子：

例 42

```
\BeforeAddtoBitoc{\ctexset{section/name = Section\space}}
\bisection{中文标题}{English title}
```

`\NewTitleCommand` [<基础标题>] {<条目名>} {<章节层次>}

New
2024aw

创建一个命令名为 <条目名> 新章节标题，并且支持 \TeX 中的标题样式选项机制。指定可选参数后标题格式除了编号/名字会与 <基础标题> 相同，否则默认为空。同时创建一个名为 <条目名> 的计数器和标记命令 `\<条目名>mark`。同时还需指定标题的 <章节层次>，例如标准文档类中 `chapter` 级别为 0。

`\NewTitleEntryCommand` [<扩展名>] {<条目名>}

New
2024aw

创建一个命令 `\<条目名>entry` {<标题>}，仅将 <标题> 加入文件扩展名为 <扩展名> 的目录中，而不在正文中输出标题，默认扩展名为 `toc`。

例 43

```
\NewTitleCommand[chapter]{lchapter}{0}
\NewTitleEntryCommand{lchapter}
\contentsuse[0]{lchapter}{toc}
\RegisterTocName{lchapter}
\ctexset
{
  lchapter/runin = true,
```

```

\lchapter/number = \thelchapter
}
\lchapter{正常标题}
\lchapterentry{加入目录的标题}

```

`\CopyTitleFormat{<条目名1>}{<条目名2>}`

将名为<条目名2>的章节标题命令除了编号/名字外的格式复制给名为<条目名1>的标题命令。

例 44

```

\ctexset{section/format = \sffamily\color{blue}}
\CopyTitleFormat{subsection}{section}

```

4.4 上标和下标

上标和下标的功能来自 `spbmark` 宏包，可以使用键值列表的形式对上下标格式进行局部设置或用 `\spbset` 命令全局设置。所有选项也支持用 `tools` 模块设置，更多选项可见宏包文档。

`\super*[(键值列表1)]{<内容>}[(键值列表2)]`

输出上标的命令，可用于文本和数学模式，在数学模式中可代替 \wedge 符号和 `\sp` 命令。命令中两个可选参数的键值列表等效。

`\sub*[(键值列表1)]{<内容>}[(键值列表2)]`

输出下标的命令，可用于文本和数学模式，在数学模式中可代替 $_$ 符号和 `\sb` 命令。命令中两个可选参数的键值列表等效。

`\supersub*[(键值列表1)]{<上标内容>}{<下标内容>}[(键值列表2)]`

同时输出上标和下标的命令，与简短命令 `\spb` 等效。命令中两个可选参数的键值列表等效。

`\spbifmath{<数学模式代码>}{<文本模式代码>}`

根据上下标处于数学还是文本模式分别设置不同的值，应当在格式命令或偏移距离选项中使用。

上下标命令键值列表中共同的选项如下所示。当下面除 `mode` 以外的选项用于 `style` 模块全局设置时，它们将根据前面最近一个命令的类型设定相应值。

`vmove` = <刚性长度>

Default: 0pt

上标或下标的垂直偏移。如果是上下标命令使用则代表上标和上标间的额外垂直距离 `vsep`。

`hmove` = <刚性长度>

Default: 0pt

上标或下标的水平偏移。如果是上下标命令使用则代表上标和上标的共同偏移。

`cmd` = <格式命令>

上标或下标的格式命令选项。最后一个命令可以带有一个参数，这个参数用于接受上标或下标。下面两种设置方式是等效的：

例 45

```

\spbset{spcmd = \color{red}}
\ctexset{tools}{spcmd = \color{red}}

```

`height` = <刚性长度>

段落中当前行上标或下标的基线与前一行的间距。

`depth` = <刚性长度>

段落中当前行上标或下标的基线与下一行的间距。

`mode = text|math|match`

上标或下标的输出格式。可以为文本 `text` 或数学 `math` 模式。当使用选项 `match` 时可以根据当前段落所处的模式自动匹配。

4.5 公式

`\begin{equation}`

`\begin{multline}`

`\begin{gather}`

`\begin{align}`

`amsmath` 宏包提供了一系列行间数学环境，它们可以排布各种对齐或不对齐的公式组。`multline` 环境用于一行无法放下的长公式，`gather` 环境用于每行居中的公式组，`align` 环境用于多列对齐的公式组。还有例如 `mathtools` 等对数学功能进行扩展的宏包，提供了更全面的数学命令与环境。

4.6 表格

`\begin{tabular}{<列格式>}`

`\begin{tblr}{<键值列表>}`

`tabularray` 是一个强大且全面的表格宏包，不依赖其它表格宏包所以同时使用不会产生冲突。不仅提供了各种传统表格宏包的功能，还实现了用键值列表方式简便地设置表格样式。具体用法可见宏包的说明文档，下面演示一个 `tblr` 表格的用法：

例 46

```
\begin{tblr}
{
  colspec      = {X[2,l]XX[LimeGreen]X},
  rowspec      = {Q[t]Q[m]Q[b,Magenta]Q[t]Q[m]Q[b]},
  hline{1,6}   = {1pt,solid},
  hline{3-5}   = {LightBlue,dashed},
  vline{1,5}   = {3-4}{dotted},
  rows         = {1cm,abovesep = 2pt},
  column{1}    = {font = \sffamily,mode = math}
}
Alpha   & Beta   & Gamma   & Delta \\
Epsilon & Zeta    & Eta     & Theta \\
Iota    & Kappa   & \SetCell[r = 2]{h,c,Cyan}Lambda & Mu \\
Nu      & Xi      & Omicron & Pi \\
Rho     & Sigma   & Tau     & Upsilon
\end{tblr}
```

4.7 列表

列表宏包 `enumitem` 为系统预置的三个列表环境提供了灵活的标签以及间距控制。

`\setlist*{<列表类型>,<级别>}{<键值列表>}`

设置列表环境的通用接口命令。带有星号参数的命令在原有格式上补充。`<列表类型>` 是排序列表 `enumerate`、常规列表 `itemize` 和主题列表 `description` 三个基本类型，也可以是 `\newlist` 新建的定理环境，若不指定则设置所有列表类型的格式。若不指定 `<级别>` 则设置所有列表级别的格式。

`\newlist{⟨列表名⟩}{⟨列表类型⟩}{⟨最大级别⟩}`

克隆基本的⟨列表类型⟩在其基础上新建一个名为⟨列表名⟩的列表环境。不要将⟨最大级别⟩设置过大以避免创建过多的计数器和相关宏。

例 47

```
\setlist{} % 清除所有列表共享的格式
\setlist[1]{labelindent = \parindent}
\setlist[itemize,1]{label = $\triangleleft$}
\setlist[enumerate]{labelsep = *,leftmargin = 1.5pc}
\newlist{steps}{enumerate}{2}
\setlist[steps,1,2]{label = (\alph*)}
```

`\begin{enumerate}[⟨键值列表⟩]`

`\begin{itemize}[⟨键值列表⟩]`

`\begin{description}[⟨键值列表⟩]`

已为三种列表各自预置了三个级别。同种列表或不同种列表间的不同级别可相互嵌套，可以使用短编号的形式，带有星号的环境为行内列表。默认列表标签与标准文档类一致，下面举几个更改列表标签以及其它格式的例子：

例 48

```
\setlist*{wide = 0pt} % 取消所有列表的左边距
\begin{enumerate}[label = (\arabic*),leftmargin = 1cm,resume]
\begin{enumerate*}[(1)] % 短编号形式
\begin{itemize}[label = \textbullet,labelsep = *]
\begin{description}[font = \sffamily\bfseries,style = nextline]
```

4.8 悬挂缩进

`\hangpara{⟨缩进长度⟩}{⟨缩进开始行数⟩}`

悬挂缩进命令 `\hangpara` 使其后的一个段落保持悬挂缩进。参数与 `hangparas` 环境相同，不过只有一个段落。⟨缩进长度⟩可以是带单位的数值或由文本长度测定命令 `\widthof` 提供。

`\hangfrom*{⟨缩进长度⟩}{⟨悬挂内容⟩}[⟨缩进开始行数⟩]`

指定首行的⟨悬挂内容⟩，其后一个段落保持悬挂缩进。⟨缩进长度⟩默认为⟨悬挂内容⟩文本长度 `\hangwd` 的值，支持数学表达式。⟨缩进开始行数⟩默认为 1。如果是带有星号参数的命令，则首行不输出⟨悬挂内容⟩，以等于⟨缩进长度⟩长度的空白代替。

例 49

```
\hangfrom[\hangwd + 1em]{悬挂标签\hspace{0.5em}}[1]
\hangfrom*[3em]{}
```

`\begin{hangparas}{⟨缩进长度⟩}{⟨缩进开始行数⟩}`

悬挂缩进环境 `hangparas` 有两个必选参数，使其中的段落均保持悬挂缩进。每个段落的缩进将以⟨缩进长度⟩从⟨缩进开始行数⟩开始。

4.9 计数器

`\newcounter{⟨计数器⟩}[⟨父计数器⟩]`

创建一个初始值为 0 的计数器。若⟨计数器⟩已经存在则报错，若指定⟨父计数器⟩则将它加入⟨父计数器⟩

的递增归零清单。

`\declarecounter{<计数器>}[<父计数器>]`

创建一个初始值为 0 的计数器，与 `\newcounter` 的区别是若 `<计数器>` 已经存在仍会强制创建。

`\setcounter{<计数器>}{<整数>}`

将计数器的值设定为指定值。

`\stepcounter{<计数器>}`

使计数器的值递增 1，同时它的递增归零清单中的子计数器值归零。

`\counterwithin*{<格式>}{<计数器>}[<连字符>]{<父计数器>}`

对计数器进行绑定的命令。将 `<计数器>` 绑定于 `<父计数器>`，即 `<父计数器>` 的值递增时 `<计数器>` 的值会归零。输出格式 `\the<计数器>` 被定义为 `\the<父计数器>[<连字符>]{<格式>}{<计数器>}`。`<格式>` 一般为更改计数器编号形式的命令，默认为 `\arabic`。带有星号参数的命令仅进行绑定不更改输出格式。

`\counterwithout*{<格式>}{<计数器>}{<父计数器>}`

对计数器关系进行解绑的命令。将 `<计数器>` 解绑于 `<父计数器>`，同时输出格式被改为 `<格式>{<计数器>}`。带有星号参数的命令仅进行解绑不更改输出格式。

例 50

```
\counterwithin[\Roman]{child}[-]{parent}
\counterwithout*{child}{parent}
```

5 扩展功能

5.1 盒子

文档类预置了两个行间盒子环境和两个行内盒子命令，需要在导言区**手动载入** `tcolorbox` 宏包提供支持。它们（配置选项）的键值列表可见宏包文档的 **Option Keys** 一节，可以进行边距、字体及背景颜色等各种个性化设置。如果 `<配置选项>` 较长且希望在原有基础上改为全局固定的样式，可以使用 `\addtobcstyle` 命令设置，第一个参数为 `tc-<环境/命令名>`。

`\deftcstyle{<盒子风格>}{<键值列表>}`

新建或重置一个名称为 `<盒子风格>` 的 `tcolorbox` 宏包盒子风格，可以与 `\addtobcstyle` 命令联用。

`\addtobcstyle{<盒子风格>}{<键值列表>}`

修改已存在的 `tcolorbox` 宏包盒子风格。这个命令可以调整以下即将介绍的预置盒子风格，对于预置盒子来说参数 `<盒子风格>` 即为它们对应的环境或命令名。

例 51

```
\deftcstyle{fancy} % 新建一个盒子风格fancy
{
  colback = Snow,colframe = LimeGreen,coltext = Coral,
  fontupper = \bfseries,arc = 6pt,
  boxrule = 5pt,boxsep = 5pt,
  borderline = {2pt}{2pt}{white}
}
\newtcolorbox{mytcbox}[1][fancy]{enhanced,fancy,title = #1}
\begin{tcolorbox}[fancy,enhanced]
  盒子内容
\end{tcolorbox}
\begin{mytcbox}[盒子标题]
```

```

    盒子内容
\end{mytcbox}

```

`\begin{easyboxi}*[⟨颜色名⟩](⟨段落标题⟩)[⟨配置选项⟩]`

行间环境 `easyboxi` 创建一个带有三个可选参数的扁平盒子。使用星号参数时给盒子添加边框, `⟨颜色名⟩` 选项设置背景颜色, `⟨段落标题⟩` 选项可以设置标题, 默认标题为空。

例 52

```

\addtotcbstyle{tc-easyboxi}{arc = 6pt}
\begin{easyboxi}*[DarkCyan](段落标题)
    上半盒子内容
    \tcblower % 盒子上下部分分界线
    下半盒子内容
\end{easyboxi}

```

`\begin{easyboxii}*[⟨颜色名⟩](⟨段落标题⟩)[⟨配置选项⟩]`

环境 `easyboxii` 用法与 `easyboxi` 类似, 同样带有三个可选参数。盒子的左边带有一条粗竖线, 背景颜色为竖线颜色的浅色。 `⟨颜色名⟩` 选项可以设置竖线的颜色。使用星号参数则盒子无背景颜色。

`\eboxi[⟨颜色名⟩]{⟨内容⟩}[⟨配置选项⟩]`

行内盒子命令 `\eboxi` 具有模糊边缘类似高亮的效果。

`\eboxii[⟨颜色名⟩]{⟨内容⟩}[⟨配置选项⟩]`

行内盒子命令 `\eboxii` 则有直角边缘和四周框线, 效果类似带背景色的 `\fbox` 盒子。

5.2 代码

代码抄录宏包 `listings` 提供了完善的代码排版功能, 并且不需要前端处理器。

`\lstdefinestyle{⟨代码风格⟩}{⟨键值列表⟩}`

代码风格定义命令, 提供大量选项对代码样式进行高度定制。可以根据需要的代码语言定义多种风格, 默认使用的 `⟨代码风格⟩` 为适用于 `TEX` 语言的 `lst-latex`。

`\lstinline[⟨键值列表⟩]{⟨行内代码⟩}`

行内代码抄录命令。较原生的 `\verb` 命令有较大改进, 具有代码语言和样式支持。

`\begin{lstlisting}[⟨键值列表⟩]`

行间代码抄录环境。支持多种语言库, 默认使用 `LaTeX` 代码语言, 可使用 `⟨键值列表⟩` 局部修改代码样式。另外还提供了 `\lstnewenvironment` 命令包装自定义名称的代码环境。

例 53

```

\lstdefinestyle{lst-pascal}
{
    language      = Pascal,
    numbers       = left,
    firstnumber    = last,
    frame         = shadowbox,
    backgroundcolor = \color{MistyRose},
    rulesepcolor  = \color{DarkGray}
}
\lstnewenvironment{mypascal}[1] [] {\lstset{style = lst-pascal,#1}}{}

```

`listlstlistingname` = \langle 代码目录标题名 \rangle

`lstlistingname` = \langle 代码题注标题名 \rangle

代码目录名与代码环境题注名选项 `listlstlistingname` 和 `lstlistingname` 用法与 $\text{CT}_{\text{E}}\text{X}$ 使用手册标题汉一节类似，可以看作对这节的扩展。

5.3 物理量

使用 `siunitx` 宏包提供国际单位制的物理量支持。`siunitx` 宏包旨在为 $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ 用户提供一种统一的方法，以便正确、轻松地排版数字和单位。`siunitx` 的设计理念是默认遵循约定的规则，但允许通过选项设置进行更改。

`\num` [\langle 键值列表 \rangle] { \langle 数字 \rangle }

`\num` 命令可以输出以科学计数法表示的数值。

`\unit` [\langle 键值列表 \rangle] { \langle 单位 \rangle }

单位格式化命令 `\unit` 包含文字元素（例如字母或数字）时，使用 `{.}` 和 `~` 放置在单元乘积中间，并使用 `_` 和 `^` 正确放置指定的下标和上标。格式化方法可在数学和文本模式下使用。

`\qty` [\langle 键值列表 \rangle] { \langle 数字 \rangle } [\langle 前缀 \rangle] { \langle 单位 \rangle }

命令 `\qty` 可用来输出带有单位的量。

`\qtyrange` [\langle 键值列表 \rangle] { \langle 起始数字 \rangle } { \langle 结束数字 \rangle } { \langle 单位 \rangle }

命令 `\qtyrange` 可以输出带有范围和单位的量。

例 54

```
\numproduct{-21x3e5}
\unit{mL~min^{-1}}
\qty{5}{\mole}
\qtyrange[unit-color = green]{1.5}{4}{kg}
```

5.4 标签引用

`\labelformat` { \langle 计数器 \rangle } { \langle 标签引用格式 \rangle }

标签格式命令 `\labelformat` 可以设置标签引用格式。 \langle 标签引用格式 \rangle 中 `#1` 代表 `\the` {计数器}，可以设置引用序号的前缀和后缀。也可以使用 `cleveref` 宏包提供更强大的标签功能。

`\AtEndPreamble` { \langle 格式代码 \rangle }

导言区末尾的钩子命令。为了保证兼容性 `hyperref` 宏包被设计在导言区末尾载入，一些情况下需要钩子命令在 `hyperref` 宏包后方执行 \langle 格式代码 \rangle ，例如载入 `cleveref` 宏包。

例 55

```
\labelformat{chapter}{第#1章}
\labelformat{chapter}{第\thechapter 章}
\AtEndPreamble{\usepackage{cleveref}}
```

5.5 索引

`imakeidx` 宏包增强了索引的功能，可使用户在制作和排版文档的同时制作一个或多个索引。

`\indexsetup` { \langle 键值列表 \rangle }

设置索引表整体格式的通用接口命令，它的 `level` 和 `headers` 选项失效。

`\index[⟨索引名⟩]{⟨标签⟩}`

索引标签命令的可选参数可以用不同的⟨索引名⟩给多个索引表设置标签。

`\makeindex[⟨键值列表⟩]`

用于导言区开启索引收集的命令。这个命令可以重复使用收集多个索引表，以不同的⟨索引名⟩作为区分，如果仅打印一个索引表没必要重复使用。

`\printindex*[⟨键值列表⟩]`

打印不同索引表的命令。这个命令被重新定义，⟨键值列表⟩中的选项与[小节 4.2](#)其它目录命令相同，并多出 `name` 选项用于指定打印索引的名字。

`name = ⟨索引名⟩`

指定输出名为⟨索引名⟩的索引表，如果只使用一个索引表则默认为项目文件名不需要指定。

5.6 边注

`\marginnote[⟨左文本⟩]{⟨右文本⟩}[⟨垂直偏移⟩]`

边注命令用于在两侧页边空白添加标注。如果只给定了⟨右文本⟩，那么边注在奇偶数页文字相同。如果同时给定了⟨左文本⟩，则偶数页使用⟨左文本⟩的文字。

`\reversemarginpar`

边注位置设置命令，用于将边注排版到与默认边空位置相反的边空中。

5.7 公式符号对照

`\begin{eqcomp}[⟨键值列表1⟩](⟨键值列表2⟩)`

一些公式之后可以使用基于列表环境的 `eqcomp` 进行符号对照。对照列表的⟨键值列表2⟩与 `enumitem` 的选项相同，一般较少用到，可以用于设置垂直间距等。

例 56

```
\seteqcomp{intro = 上式中}
\begin{eqcomp}[width = 1.5em]
  \symb{Ab}[$\to$] 符号说明
  \symb{C}[--] 符号说明
\end{eqcomp}
```

`\symb*{⟨符号⟩}[⟨引导符⟩]`

符号条目命令 `\symb` 类似 `\item`，需要在环境内使用，带有星号的命令可在文本段落单独使用。

`\seteqcomplist{⟨键值列表2⟩}`

对照列表条目格式的⟨键值列表2⟩可以使用 `\seteqcomplist` 全局设置。

`\seteqcomp{⟨键值列表1⟩}`

对照环境的⟨键值列表1⟩可以使用 `\seteqcomp` 全局设置，它的选项如下：

`width = ⟨符号宽度⟩`

Default: 10pt

设置适当的符号宽度可以防止文字重叠和边界溢出等问题。

`delim = ⟨引导符⟩`

Default: ——

全局的符号对照引导符。`\symb` 命令局部⟨引导符⟩的对齐方式是基于这个符号的宽度。

`space = ⟨符号间隙⟩`

Default: 0.25em

符号间隙为引导符两侧与符号和说明文字间的距离。

`item-align = l|c|r`

符号的对齐方式，分为左 `l`、中 `c` 和右 `r` 三种对齐方式。

`delim-align = l|c|r`

引导符的对齐方式，分为左 `l`、中 `c` 和右 `r` 三种对齐方式。

`enumitem = <键值列表2>`

可以使用选项 `enumitem` 或命令 `\seteqcomplist` 调整对照列表的水平和垂直间距。

`intro = <首行文字>`

符号对照的说明文字。应当为简短的几个字，只显示在第一行，其后为悬挂缩进的对照列表。

`font = <符号字体>`

符号的字体。数学和文本符号模式应采用不同的字体命令，最后一个命令可以接受一个参数，这个参数为符号。

`mode = math|text`

符号字体模式，可以为数学字体或者文本字体。

5.8 插入文档

`\includepdf[<键值列表>]{<文档名>}`

该命令由文档宏包 `pdfpages` 提供。可以将其它 pdf 文档插入当前文档，比如插入其它软件制作的封面。可选参数可以指定插入页数、逻辑页和横向页面等，详情查看宏包文档。

A 更新日志

1.45 (2022/01/08) — 经过前期各模块的功能调整和错误修复，稳定版本正式发布。

1.46 (2022/01/12)

Added

- 增加定义脚注正文和内容标记格式的命令 `\defupfntmark` 和 `\defdownfntmark`。
- 新增章节标题目录中双语标题条目的输出排列方式选项 `bilist`。

1.55 (2022/01/28)

Added

- 新增文类选项 `theorem` 用于提供定理功能支持。

Deprecated

- 由于命令存在一些无法解决的问题弃用定理环境定义命令 `\deftheorem`。
- 文档类移除预置的定理环境，改为由用户自行定义环境。
- 考虑到功能性和实用性不强弃用摘要环境 `outline` 和习题环境 `exercise`。

1.62 (2022/02/16)

Added

- 增加目录条目下方执行的代码选项 `tocbelow`。

Changed

- 目录改用标题级别选项 `level` 设定标题命令，不输出标题则使用星号参数。

1.636 (2022/03/03)

Added

- `\tableofcontents` 命令增加输出方案选项 `plan`，不再使用 `+` 参数输出英文目录。
- 增加英文主文档目录钩子选项 `bibetween`。

1.638 (2022/03/04) — 增加目录条目样式选项注册命令 `\RegisterTocName`。

1.644 (2022/03/10) — 增加浮动环境目录声明命令 `\DeclareFloatList` 配合 `caption` 宏包。

1.682 (2022/05/01) — 将迷你页判断命令 `\notminipage` 改为被保护命令兼容 `circledtext` 宏包。

2024ap (2024/03/15)

Fixed

- 修复图表等浮动环境题注目录中条目标签编号的计数器值显示错误问题。
- 修复附录命令 `\appendix` 的可选参数设置英文目录中条目名字和编号失效的问题。
- 修复 `biblatex` 宏包对脚注打补丁失效的问题。

Added

- 新增目录风格定义命令 `\SetTocStyle` 和使用命令 `\UseTocStyle`。
- 新增目录配置命令 `\SetTocOuter` 和双语目录判断命令 `\ebifbitoc`。
- 新增脚注段落文本格式定义命令 `\defntpara` 和脚注间距离选项 `footnotesep`。
- 新增悬挂缩进命令 `\hangfrom`。
- 新增行间公式与正文上下间距选项 `abovedisplayskip` 和 `belowdisplayskip`。
- 新增目录文件扩展名分配命令 `\AssignCaptionExt`。

2024aq (2024/03/31) — 修复命令 `\ebifbitoc` 用于条目缩进选项 `tocindent` 失效的问题。

2024as (2024/04/19)

Changed

- 目录条目样式注册命令 `\RegisterTocName` 增加星号参数，支持设置段落型目录。

Added

- 新增目录条目标题标签定义的选项 `toclabel`。
- 新增段落型目录条目附加代码的选项 `tocbegin`、`tocsep` 和 `tocend`。

2024aw (2024/04/22) — 新增新建章节标题命令 `\NewTitleCommand` 和 `\NewTitleEntryCommand`。

2024bb (2024/04/26) — 增加对 `thmtools` 宏包定理环境目录 `\listoftheorems` 的支持。

2024bk (2024/05/07) — 新增局部题注目录分配命令 `\AssignPartialExt`。

2024bn (2024/05/10)

Fixed

- 修复 `toc` 模块中带有 `every` 后缀的选项不对所有已注册条目有效的问题。

Added

- 新增统一设置所有目录格式时的条目类型排除命令 `\ExcludeEveryTocName`。

2024cd (2024/06/14) — 修复全局选项 `class` 无法正确切换单双面排版的问题。

2024cf (2024/06/16)

Fixed

- 修复和改进新建章节/图表标题时目录层次深度的问题。

Changed

- `tools` 模块的图片路径选项 `figurepath` 可在正文中使用。

2024ci (2024/06/22) — 修复使用 `flush` 类型脚注报错。

2024co (2024/07/06) — 目录命令的可选参数支持 \TeX 键值列表所有路径的章节标题样式选项。

2024cq (2024/07/12) — 新增章节标题格式复制命令 `\CopyTitleFormat`。

2024cu (2024/07/20) — 修复无编号的定理环境标签命令 `\p@⟨定理名⟩` 缺失导致报错的问题。

2024cx (2024/07/28) — 新增将脚注置于页面底部的命令 `\PushFntBottom`。

2024dh (2024/08/06)

Changed

- 取消默认载入 `ulem` 下划线宏包。

Added

- 使用 `xtemplate` 宏包将封面标题命令 `\maketitle` 抽象为由实例组成的模板。
- 新增图表题注模块 `cap`。

2024dq (2024/09/16) — 修复脚注重置编号的方式为 `page` 时在一些换页情况下失败的问题。

2024dt (2024/09/22) — 新增用于设置行距的命令 `\usestretch` 和环境 `usespace`。

2024ea (2024/09/28) — 定理模块 `thmt` 支持使用 `keytheorems` 宏包。

B 参考文献

- [CMS23] Yukai Chou, Frank Mittelbach, and Ulrich M. Schwarz.
thmtools. version 0.76, May 4, 2023 (or newer). URL: <https://ctan.org/pkg/thmtools>.
- [CTE22] CTEX.ORG. CTEX Macros Manual. version 2.5.10, July 14, 2022 (or newer).
URL: <https://ctan.org/pkg/ctex>.
- [Hu17] Wei Hu. L^AT_EX 2_ε Document Class and Macro Package Study Manual.
Beijing: Tsinghua University Press, 2017.
- [Liu13] Haiyang Liu. Getting Started with L^AT_EX. Beijing: Electronic Industry Press, 2013.
- [Lóp19] Javier Bezos López. enumitem. version 3.9, June 20, 2019 (or newer).
URL: <https://ctan.org/pkg/enumitem>.
- [Lóp23] Javier Bezos López. titletoc. version 2.15, Oct. 17, 2023 (or newer).
URL: <https://ctan.org/pkg/titletoc>.
- [MF23] Frank Mittelbach and Robin Fairbairns. footmisc. version 6.0f, July 5, 2023 (or newer).
URL: <https://ctan.org/pkg/footmisc>.
- [Mor22] Konstantin Morenko. eqexpl. version 1.1.1, June 18, 2022 (or newer).
URL: <https://ctan.org/pkg/eqexpl>.
- [Som23] Axel Sommerfeldt. caption. version 3.6o, Sept. 8, 2023 (or newer).
URL: <https://ctan.org/pkg/caption>.
- [Stu23] Thomas F. Sturm. tcolorbox. version 6.1.0, Sept. 26, 2023 (or newer).
URL: <https://ctan.org/pkg/tcolorbox>.
- [Tea23a] The L^AT_EX Project Team.
The L^AT_EX3 Interfaces. version 2023-12-11, Dec. 11, 2023 (or newer).
URL: <https://ctan.org/pkg/l3kernel>.
- [Tea23b] The L^AT_EX Project Team. The L^AT_EX 2_ε Sources. version 2023-11-01, Nov. 1, 2023 (or newer).
URL: <https://ctan.org/pkg/source2e>.

- [TF22] Geoffrey Tobin and Robin Fairbairns. **setspace**. version 6.7b, Dec. 4, 2022 (or newer).
URL: <https://ctan.org/pkg/setspace>.
- [TOR23] The L^AT_EX Project Team, Heiko Oberdiek, and Sebastian Rahtz.
hyperref. version 7.01g, Nov. 26, 2023 (or newer).
URL: <https://ctan.org/pkg/hyperref>.
- [WHU21] WHUTUG.
L^AT_EX thesis template for Wuhan University. version 0.6d, May 20, 2021 (or newer).
URL: <https://github.com/whutug/whu-thesis>.
- [XZ24] Yu Xiong and Yichen Zhao.
L^AT_EX thesis template for Nanjing University. version 1.4.1, Apr. 22, 2024 (or newer).
URL: <https://ctan.org/pkg/njuthesis>.
- [Zen23] Xiangdong Zeng.
L^AT_EX thesis template for Fudan University. version 0.9a, May 27, 2023 (or newer).
URL: <https://ctan.org/pkg/fduthesis>.

C 索引

A	bodyfont	11
<code>abovedisplayskip</code>	book	5
<code>\addtotcbstyle</code>	break	26
<code>aftercode</code>	C	
<code>aftername</code>	<code>capfig-spread</code>	8
<code>aftertitle</code>	<code>caplst-spread</code>	8
<code>align (environment)</code>	<code>captab-spread</code>	8
<code>align</code>	<code>\caption</code>	14, 21
<code>\appendix</code>	<code>caption-every</code>	9
<code>article</code>	<code>caption-figure</code>	9
<code>\AssignCaptionExt</code>	<code>caption-lstlisting</code>	9
<code>\AssignPartialExt</code>	<code>caption-spread</code>	8
<code>\AtEndPreamble</code>	<code>caption-table</code>	9
<code>author</code>	<code>\captionof</code>	14
B	<code>\ch</code>	4
<code>backend</code>	<code>\chapter</code>	5
<code>\backmatter</code>	<code>chapter</code>	25
<code>before</code>	<code>chapter-label</code>	15
<code>\BeforeAddtoBitoc</code>	<code>chapter-mark</code>	14 f.
<code>beforecode</code>	<code>chem</code>	4
<code>belowdisplayskip</code>	<code>\circledtext</code>	18
<code>between</code>	<code>citestyle</code>	9
<code>bibbetween</code>	<code>cjkfont</code>	4
<code>bibstyle</code>	<code>class</code>	5, 36
<code>\bicaption</code>	<code>\cleardoublepage</code>	24
<code>\bichapter</code>	<code>cmd</code>	28
<code>bicontentsname</code>	<code>color</code>	6
<code>bilist</code>	<code>color-every</code>	6
<code>biname</code>	<code>configs</code>	4
<code>binumber</code>	<code>content</code>	23
<code>\bisection</code>	<code>\contentsmargin</code>	22
<code>\bisubsection</code>	<code>\contentsuse</code>	21
<code>\blankpagestyle</code>	<code>\CopyTitleFormat</code>	28, 37

<code>\counterwithin</code>	31	<code>fnmark-hmove</code>	18
<code>\counterwithout</code>	31	<code>fnmark-num</code>	17 f.
<code>\CTEXifname</code>	19	<code>fnmark-vmove</code>	17
<code>\ctexset</code>	5 f., 22, 25	<code>fnmarktext-format</code>	18 f.
D		<code>fnmarktext-num</code>	17 f.
<code>datafile</code>	10	<code>fnpara-format</code>	8, 19
<code>date</code>	23	<code>\fnparskip</code>	17
<code>\declarecounter</code>	31	<code>fnparskip</code>	17
<code>\DeclareFloatList</code>	21, 36	<code>fntext-code</code>	19
<code>\DeclareInstance</code>	23	<code>fntext-num</code>	17 f.
<code>\DeclareLinkColor</code>	7	<code>fntext-pos</code>	17
<code>\DeclareThemeColor</code>	6	<code>fntext-vmove</code>	18
<code>\declaretheorem</code>	10 ff.	<code>fntnumwith</code>	6
<code>\declaretheoremstyle</code>	11	<code>fntspread</code>	8, 17
<code>\defbibfilter</code>	10	<code>font</code>	4, 35
<code>\defbibheading</code>	10	<code>\Footnote</code>	18
<code>\defbibnote</code>	10	<code>footnote</code>	9, 19
<code>\defdownfntmark</code>	18, 35	<code>\Footnotemark</code>	18
<code>\deffntpara</code>	18, 36	<code>footnoterule</code>	15
<code>\deftcbstyle</code>	31	<code>footnoterulewd</code>	15
<code>\defupfntmark</code>	18, 35	<code>footnotesep</code>	17, 36
<code>delim</code>	34	<code>\Footnotetext</code>	18
<code>delim-align</code>	35	<code>footnotetype</code>	16 f.
<code>depth</code>	28	<code>\footref</code>	17
<code>description (environment)</code>	30	<code>footrule</code>	15
<code>dotalign</code>	20, 22 f.	<code>footruleskip</code>	13
<code>draft</code>	3	<code>format</code>	26
E		<code>format-code</code>	13
<code>easyboxi (environment)</code>	11, 32	<code>\frontmatter</code>	24
<code>easyboxii (environment)</code>	32	G	
<code>\ebaddtofont</code>	9	<code>gather (environment)</code>	29
<code>\ebifbitoc</code>	25, 36	<code>\geometry</code>	13
<code>\eboxi</code>	32	H	
<code>\eboxii</code>	32	<code>hang</code>	22
<code>enmode</code>	3	<code>\hangfrom</code>	30, 36
<code>enumerate (environment)</code>	30	<code>\hangpara</code>	30
<code>enumitem</code>	35	<code>hangparas (environment)</code>	30
<code>eqcomp (environment)</code>	34	<code>\hangwd</code>	30
<code>equation (environment)</code>	29	<code>hdivide</code>	13
<code>everyspread</code>	8	<code>headers</code>	33
<code>\ExcludeEveryTocName</code>	23, 36	<code>headfont</code>	11
F		<code>headformat</code>	12
<code>\fancyfoot</code>	16	<code>headindent</code>	11
<code>\fancyhead</code>	16	<code>headpunct</code>	11
<code>\fancyhfoffset</code>	13	<code>headrule</code>	15
<code>\fancypagestyle</code>	15	<code>headruleskip</code>	13
<code>figure (environment)</code>	4, 14	<code>headrulewd</code>	15
<code>figure/format</code>	14	<code>height</code>	28
<code>figurepath</code>	7, 36	<code>hmargin</code>	13
<code>filter</code>	10	<code>hmove</code>	28
<code>floatpage</code>	4	<code>hyphen-equation</code>	7
<code>\fnafterindent</code>	17	<code>hyphen-every</code>	7
<code>fnafterindent</code>	17	<code>hyphen-figure</code>	7
<code>fnboth-format</code>	19	<code>hyphen-lstlisting</code>	7
<code>fncustom-format</code>	18 f.	<code>hyphen-table</code>	7
<code>\fnfirstindent</code>	17	<code>\hytoclink</code>	20
<code>fnfirstindent</code>	17		

C 索引

I

<code>\includepdf</code>	35
<code>\index</code>	34
<code>\indexsetup</code>	33
<code>intro</code>	35
<code>item-align</code>	35
<code>itemize (environment)</code>	30

L

<code>\labelformat</code>	22, 33
<code>\leftmark</code>	7
<code>level</code>	25 f., 33, 35
<code>linespread</code>	8
<code>linkcolor</code>	7
<code>linkcolor-every</code>	8
<code>linktoc</code>	8
<code>linktype</code>	7
<code>listlstlistingname</code>	33
<code>\listnumberline</code>	22
<code>\listoffigures</code>	25
<code>\listoflstlistings</code>	25
<code>\listoftables</code>	25
<code>\listoftheorems</code>	12, 36
<code>listtheoremname</code>	12
<code>\loadgeometry</code>	13
<code>loeskip</code>	12
<code>lolskip</code>	22
<code>\lstdefinestyle</code>	32
<code>\lstinline</code>	32
<code>lstlisting (environment)</code>	32
<code>lstlisting/font</code>	14
<code>lstlistingname</code>	33
<code>\lstnewenvironment</code>	32

M

<code>\mainmatter</code>	24
<code>\makeindex</code>	34
<code>\maketitle</code>	23, 37
<code>\marginnote</code>	34
<code>marginpar</code>	9
<code>\markboth</code>	16
<code>\markdouble</code>	16
<code>\markright</code>	16
<code>\markrule</code>	15 f.
<code>mathfont</code>	5
<code>mode</code>	28 f., 35
<code>multline (environment)</code>	29
<code>multoc</code>	6, 25

N

<code>name</code>	11, 34
<code>\newcounter</code>	30 f.
<code>\newgeometry</code>	13
<code>newline</code>	3
<code>\newlist</code>	29 f.
<code>\newtheorem</code>	10
<code>\NewTitleCommand</code>	27, 36
<code>\NewTitleEntryCommand</code>	27, 36
<code>nofoot</code>	13
<code>none</code>	25

<code>notebraces</code>	11
<code>notefont</code>	11
<code>\notminipage</code>	18, 36
<code>\num</code>	33
<code>number</code>	26
<code>numberbychapter</code>	7
<code>numbered</code>	11 f.
<code>numberlike</code>	11
<code>\numberline</code>	19
<code>numbers</code>	9

P

<code>pagenumwd</code>	23
<code>\pagestyle</code>	16
<code>pagestyle</code>	26
<code>paper</code>	4, 13
<code>plan</code>	24 f., 36
<code>postheadsapce</code>	11
<code>prenote</code>	10
<code>\printbibliography</code>	10
<code>\printindex</code>	34
<code>\PushFntBottom</code>	17, 37

Q

<code>qed</code>	11
<code>\qedsymbol</code>	11
<code>\qty</code>	33
<code>\qtyrange</code>	33

R

<code>\RegisterCapOptions</code>	14
<code>\RegisterTocName</code>	20 f., 36
<code>\restoregeometry</code>	13
<code>\reversemarginpar</code>	34
<code>\rightmark</code>	7

S

<code>\savegeometry</code>	13
<code>scheme</code>	3
<code>section</code>	10, 25
<code>section-label</code>	15
<code>section-mark</code>	14 f.
<code>\setcounter</code>	31
<code>\seteqcomp</code>	34
<code>\seteqcomplist</code>	34 f.
<code>\SetFntSymsList</code>	17
<code>\setlist</code>	29
<code>\settheoremtocline</code>	12
<code>\SetTocOuter</code>	25, 36
<code>\SetTocStyle</code>	22, 36
<code>show</code>	12
<code>space</code>	34
<code>spaceabove</code>	11
<code>spacebelow</code>	11
<code>\spb</code>	28
<code>\spbifmath</code>	28
<code>\spbset</code>	28
<code>spcmd</code>	18 f.
<code>sphmove</code>	18
<code>spvmove</code>	18

C 索引

<code>\stepcounter</code>	31	<code>tocbelow</code>	20, 35
<code>style</code>	11	<code>tocend</code>	21, 36
<code>\sub</code>	28	<code>tocformat</code>	19, 22
<code>subsection-label</code>	15	<code>tocformat*</code>	19
<code>subsection-mark</code>	15	<code>tocformat+</code>	19
<code>\super</code>	28	<code>tocindent</code>	19, 36
<code>super</code>	9	<code>tocindent-every</code>	23
<code>\supersub</code>	28	<code>toclabel</code>	20, 36
<code>\symb</code>	34	<code>tocline</code>	12 f., 19 f., 22
T		<code>\tocrule</code>	20
<code>table (environment)</code>	4, 14	<code>tocrule</code>	20
<code>table/labelsep</code>	14	<code>tocrule-every</code>	23
<code>\tableofcontents</code>	24, 36	<code>tocsep</code>	21, 36
<code>tabular (environment)</code>	29	<code>tools</code>	28
<code>tagform</code>	7	<code>type</code>	10
<code>tblr (environment)</code>	29	U	
<code>textheight</code>	13	<code>\unit</code>	33
<code>\thecontentslabel</code>	20	<code>uppercase</code>	7, 15
<code>\thecontentspage</code>	20	<code>\UseFntSymsList</code>	17
<code>\thedownfootnote</code>	18	<code>usespace (environment)</code>	8, 37
<code>theorem</code>	4, 10, 35	<code>\usestretch</code>	8, 37
<code>\theupfootnote</code>	18	<code>\UseTocStyle</code>	22, 36
<code>\thmtenvname</code>	12	V	
<code>\thmtifname</code>	12	<code>vmove</code>	28
<code>\thmtifnote</code>	12	<code>vsep</code>	28
<code>\thmtname</code>	12	W	
<code>\thmtnote</code>	12	<code>width</code>	34
<code>title</code>	10, 23, 25	<code>\widthof</code>	30
<code>toafter</code>	20	<code>within</code>	11
<code>toafter-every</code>	23	<code>withpart</code>	6
<code>tocbegin</code>	21, 36		