IMEX 新手上路指南

@AlphaZTX

2022年1月

MEX 5 TEX

MEX = TEX

• LATEX

- LATEX
 - "a software system for document preparation" (Wikipedia)

- LATEX
 - "a software system for document preparation" (Wikipedia)
 - 写论文/提交作业要用的

- LATEX
 - "a software system for document preparation" (Wikipedia)
 - 写论文/提交作业要用的
 - 排版公式很厉害

- LATEX
 - "a software system for document preparation" (Wikipedia)
 - 写论文/提交作业要用的
 - 排版公式很厉害
 - · 基于 TeX

斯EX 与 TEX

- LATEX
 - "a software system for document preparation" (Wikipedia)
 - 写论文/提交作业要用的
 - 排版公式很厉害
 - 基于 T_EX

 $\hbox{\LaTeX} \approx \hbox{T-X} + \hbox{\LaTeX} kernel + document \ classes + packages$

斯EX 与 TEX

- MEX
 - "a software system for document preparation" (Wikipedia)
 - 写论文/提交作业要用的
 - 排版公式很厉害
 - 基于 TeX MeX ≈ TeX + MeX kernel + document classes + packages
- T_EX

- MEX
 - "a software system for document preparation" (Wikipedia)
 - 写论文/提交作业要用的
 - 排版公式很厉害
 - 基于 T_EX MFX ≈ T_EX + MF_EX kernel + document classes + packages
- TEX
 - "a typesetting system which was designed and written by Donald Knuth and first released in 1978" (Wikipedia)

MEX

- "a software system for document preparation" (Wikipedia)
- 写论文/提交作业要用的
- 排版公式很厉害
- 基于 T_EX MEX ≈ T_EX + MEX kernel + document classes + packages

TEX

- "a typesetting system which was designed and written by Donald Knuth and first released in 1978" (Wikipedia)
- 真正负责排版的东西

LATEX

- "a software system for document preparation" (Wikipedia)
- 写论文/提交作业要用的
- 排版公式很厉害
- 基于 T_EX MFX ≈ T_EX + MF_EX kernel + document classes + packages

TEX

- "a typesetting system which was designed and written by Donald Knuth and first released in 1978" (Wikipedia)
- 真正负责排版的东西——"排版引擎"

MEX

- "a software system for document preparation" (Wikipedia)
- 写论文/提交作业要用的
- 排版公式很厉害
- 基于 TeX ヒffeX ≈ TeX + ŁffeX kernel + document classes + packages

TEX

- "a typesetting system which was designed and written by Donald Knuth and first released in 1978" (Wikipedia)
- 真正负责排版的东西——"排版引擎"
- 不只是 TeX:pdfTeX, ε-TeX, XfTeX, LuaTeX, pTeX, ...

MTEX

- "a software system for document preparation" (Wikipedia)
- 写论文/提交作业要用的
- 排版公式很厉害
- 基于 T_EX MEX ≈ T_EX + MEX kernel + document classes + packages

TEX

- "a typesetting system which was designed and written by Donald Knuth and first released in 1978" (Wikipedia)
- 真正负责排版的东西——"排版引擎"
- 不只是 TeX:pdfTeX, ε-TeX, XfTeX, LuaTeX, pTeX, ...
- 目前最流行的中文支持: X_{TF}X(马上就会说到)

下载和安装 TEX Live

解构:

解构:

· 你要下载的并不是一个叫「LATEX」的软件

解构:

- · 你要下载的并不是一个叫「LATEX」的软件
- · 你要下载的东西其实叫做「TEX 发行版」

解构:

- · 你要下载的并不是一个叫「LATEX」的软件
- · 你要下载的东西其实叫做「TEX 发行版」

什么是「TeX 发行版」?

解构:

- · 你要下载的并不是一个叫「LATFX」的软件
- · 你要下载的东西其实叫做「TEX 发行版」

什么是「TeX 发行版」?

• TEX distribution

解构:

- · 你要下载的并不是一个叫「LATEX」的软件
- · 你要下载的东西其实叫做「TEX 发行版」

什么是「TeX 发行版」?

• TEX distribution—套装

解构:

- · 你要下载的并不是一个叫「LATEX」的软件
- · 你要下载的东西其实叫做「TEX 发行版」

什么是「TeX 发行版」?

- TEX distribution—套装
- 引擎、宏包、文档类的"全家桶"

解构:

- · 你要下载的并不是一个叫「LATEX」的软件
- · 你要下载的东西其实叫做「TEX 发行版」

什么是「TeX 发行版」?

- TEX distribution—套装
- 引擎、宏包、文档类的"全家桶"
- · 支持不同的格式,例如:plain TeX, LATeX, ConTeXt

解构:

- · 你要下载的并不是一个叫「LATEX」的软件
- · 你要下载的东西其实叫做「TeX 发行版」

什么是「TeX 发行版」?

- T_EX distribution——套装
- 引擎、宏包、文档类的"全家桶"
- 支持不同的格式,例如:plain TFX, LATFX, ConTFXt

应该安装哪一个 TeX 发行版?

解构:

- · 你要下载的并不是一个叫「LATEX」的软件
- · 你要下载的东西其实叫做「TeX 发行版」

什么是「TeX 发行版」?

- TFX distribution——套装
- 引擎、宏包、文档类的"全家桶"
- 支持不同的格式,例如:plain TeX, LATeX, ConTeXt

应该安装哪一个 TEX 发行版?

· 推荐 TeX Live(跨平台, 每年稳定更新)

解构:

- · 你要下载的并不是一个叫「LATEX」的软件
- · 你要下载的东西其实叫做「TeX 发行版」

什么是「TeX 发行版」?

- TFX distribution——套装
- 引擎、宏包、文档类的"全家桶"
- 支持不同的格式,例如:plain TeX, LATeX, ConTeXt

应该安装哪一个 TEX 发行版?

- 推荐 TeX Live(跨平台, 每年稳定更新)
- macOS 用户也可以考虑 MacT_EX **ℱ**

解构:

- · 你要下载的并不是一个叫「LATEX」的软件
- · 你要下载的东西其实叫做「TeX 发行版」

什么是「TeX 发行版」?

- TFX distribution——套装
- 引擎、宏包、文档类的"全家桶"
- 支持不同的格式,例如:plain TeX, LATeX, ConTeXt

应该安装哪一个 TEX 发行版?

- 推荐 TEX Live(跨平台, 每年稳定更新)
- macOS 用户也可以考虑 MacTeX 🔗
- · 坚决不推荐 CTrX 套装!!!

• 通过网络安装:♂

安装 T_EX Live

- · 通过网络安装: 🔗
 - Windows:install-tl-windows.exe
 - *nix(Unix, Linux, macOS):install-tl-unx.tar.gz

- · 通过网络安装: 🔗
 - Windows:install-tl-windows.exe
 - *nix(Unix, Linux, macOS):install-tl-unx.tar.gz

速度太慢,不推荐

- · 通过网络安装: Ø
 - Windows:install-tl-windows.exe
 - *nix(Unix, Linux, macOS):install-tl-unx.tar.gz

速度太慢,不推荐

· 通过 ISO 镜像安装:

清华 tuna Ø, 中国科大镜像站 Ø, 其他 CTAN 镜像 Ø

- 通过网络安装:●
 - Windows:install-tl-windows.exe
 - *nix(Unix, Linux, macOS):install-tl-unx.tar.gz

速度太慢,不推荐

• 通过 ISO 镜像安装:

清华 tuna **多**, 中国科大镜像站 **多**, 其他 CTAN 镜像 **多** 下载 texlive(年份).iso, 例如 texlive2022.iso

- · 通过网络安装: 🔊
 - Windows:install-tl-windows.exe
 - *nix(Unix, Linux, macOS):install-tl-unx.tar.gz

速度太慢,不推荐

- · 通过 ISO 镜像安装:
 - 清华 tuna **多**, 中国科大镜像站 **多**, 其他 CTAN 镜像 **多** 下载 texlive(年份).iso, 例如 texlive2022.iso
 - Windows:双击 install-tl-windows.bat
 - *nix:终端输入 ./install-tl --gui

安装 TEX Live

- · 通过网络安装: 🔊
 - Windows:install-tl-windows.exe
 - *nix(Unix, Linux, macOS):install-tl-unx.tar.gz

速度太慢,不推荐

- · 通过 ISO 镜像安装:
 - 清华 tuna **𝚱**, 中国科大镜像站 **𝚱**, 其他 CTAN 镜像 **𝚱** 下载 texlive⟨年份⟩.iso, 例如 texlive2022.iso
 - Windows:双击 install-tl-windows.bat
 - *nix:终端输入 ./install-tl --gui

速度较快

安装 TEX Live

- · 通过网络安装: 🔊
 - Windows:install-tl-windows.exe
 - *nix(Unix, Linux, macOS):install-tl-unx.tar.gz

速度太慢,不推荐

- · 通过 ISO 镜像安装:
 - 清华 tuna **𝚱**, 中国科大镜像站 **𝚱**, 其他 CTAN 镜像 **𝚱** 下载 texlive⟨年份⟩.iso, 例如 texlive2022.iso
 - Windows:双击 install-tl-windows.bat
 - *nix:终端输入 ./install-tl --gui

速度较快

可以在安装界面配置环境变量 🔗

能

pause

在线 TEX:Overleaf **多**, TEXPage **多**

能

pause

在线 TEX: Overleaf &, TEXPage &

· Overleaf:模板丰富,多人协作支持很好

4

能

pause

在线 TEX: Overleaf &, TEXPage &

• Overleaf:模板丰富,多人协作支持很好,中文字体支持较少, 可能有网络问题

4

能

pause

在线 TFX: Overleaf **多**, TFXPage **多**

- Overleaf:模板丰富,多人协作支持很好,中文字体支持较少, 可能有网络问题
- · TEXPage:支持更多中文字体, 网络限制少

能

pause

在线 TEX: Overleaf &, TEXPage &

- Overleaf:模板丰富,多人协作支持很好,中文字体支持较少, 可能有网络问题
- T_EXPage:支持更多中文字体, 网络限制少, 免费账号项目数量 受限制

能

pause

在线 TFX: Overleaf **多**, TFXPage **多**

- Overleaf:模板丰富,多人协作支持很好,中文字体支持较少, 可能有网络问题
- T_EXPage:支持更多中文字体, 网络限制少, 免费账号项目数量 受限制

不用费时间下载和安装 TeX Live

能

pause

在线 TFX: Overleaf **多**, TFXPage **多**

- Overleaf:模板丰富,多人协作支持很好,中文字体支持较少, 可能有网络问题
- T_EXPage:支持更多中文字体, 网络限制少, 免费账号项目数量 受限制

不用费时间下载和安装 TFX Live, 但是也少了 texdoc

能

pause

在线 TEX: Overleaf &, TEXPage &

- Overleaf:模板丰富,多人协作支持很好,中文字体支持较少,可能有网络问题
- T_EXPage:支持更多中文字体, 网络限制少, 免费账号项目数量 受限制

不用费时间下载和安装 T_EX Live, <mark>但是也少了 texdoc</mark> 没关系, 我们还有在线的 texdoc:

texdoc

texdoc 是 TeX Live 提供的一个命令行程序

· 需要看什么, 就直接用 texdoc 查什么

- · 需要看什么,就直接用 texdoc 查什么
- · 打开命令行/终端,输入 texdoc〈文件名〉 即可查看对应的手册或文档

- · 需要看什么,就直接用 texdoc 查什么
- · 打开命令行/终端,输入 texdoc〈文件名〉 即可查看对应的手册或文档,支持模糊查找

- · 需要看什么,就直接用 texdoc 查什么
- 打开命令行/终端,输入 texdoc〈文件名〉即可查看对应的手册或文档,支持模糊查找
- texdoc lshort-zh-cn



- · 需要看什么, 就直接用 texdoc 查什么
- 打开命令行/终端,输入 texdoc〈文件名〉即可查看对应的手册或文档,支持模糊查找
- texdoc lshort-zh-cn 得到《一份(不太)简短的 $\mathrm{MEX}\,2_{\varepsilon}\,$ 介绍》



- · 需要看什么,就直接用 texdoc 查什么
- 打开命令行/终端,输入 texdoc〈文件名〉即可查看对应的手册或文档,支持模糊查找
- texdoc lshort-zh-cn
 得到《一份(不太)简短的 ΔΤ_EX 2_ε 介绍》——看完它!!!



准备文档

1. 在你需要的路径下新建一个 .tex 文件;

- 1. 在你需要的路径下新建一个 .tex 文件;
- 2. 用文本编辑器打开它;

- 1. 在你需要的路径下新建一个 .tex 文件;
- 2. 用文本编辑器打开它; (TFXworks, TFXstudio, VS Code, ...)

- 1. 在你需要的路径下新建一个 .tex 文件;
- 2. 用文本编辑器打开它; (TFXworks, TFXstudio, VS Code, ...)
- 3. 开始编辑;

- 1. 在你需要的路径下新建一个 .tex 文件;
- 2. 用文本编辑器打开它; (TEXworks, TEXstudio, VS Code, ...)
- 3. 开始编辑;
- 4. 编辑好了,就可以编译了。

- 1. 在你需要的路径下新建一个 .tex 文件;
- 2. 用文本编辑器打开它; (TFXworks, TFXstudio, VS Code, ...)
- 3. 开始编辑;
- 4. 编辑好了,就可以编译了。

展开讲讲

```
\documentclass{ctexart}
\usepackage{amsmath}
\title{假装是一个标题}
\author{张三}
\date{\today}
\begin{document}
\maketitle
极其敷衍的正文。
[E = m c^2.]
\end{document}
```

```
\documentclass{ctexart}
\usepackage{amsmath}
\title{假装是一个标题}
\author{张三}
\date{\today}
\begin{document}
\maketitle
极其敷衍的正文。
[E = m c^2.]
\end{document}
```

\documentclass{ctexart} 指定文档类为 ctexart;

```
\documentclass{ctexart}
\usepackage{amsmath}
```

\title{假装是一个标题} \author{张三} \date{\today}

\begin{document}

\maketitle

极其敷衍的正文。 \[E = m c^2. \]

\end{document}

\documentclass{ctexart} 指定文档类为 ctexart;

\usepackage{amsmath} 使用 amsmath 包,这是一个数学公式相关的使用频率很高的宏包;

```
\documentclass{ctexart}
\usepackage{amsmath}
```

\title{假装是一个标题} \author{张三} \date{\today}

\begin{document}

\maketitle

极其敷衍的正文。 \[E = m c^2.\]

\end{document}

\documentclass{ctexart} 指定文档类为 ctexart;

\usepackage{amsmath} 使用 amsmath 包,这是一个数学公式相关的使用频率很高的宏包;

\title、\author、\date 分别指定标题、作者、日期。其中,\today 为文档编译时所在的日期。如果不想标注日期的话,可以使用 \date{};

```
\documentclass{ctexart}
\usepackage{amsmath}
```

\title{假装是一个标题} \author{张三} \date{\today}

\begin{document}

\maketitle

极其敷衍的正文。 \[E = m c^2.\]

\end{document}

\documentclass{ctexart} 指定文档类为 ctexart;

\usepackage{amsmath} 使用 amsmath 包,这是一个数学公式相关的使用频率很高的宏包;

\title、\author、\date 分别指定标题、作者、日期。其中,\today 为文档编译时所在的日期。如果不想标注日期的话,可以使用 \date{};

在 \begin{document} 和 \end{document} 之间的称为正文,\begin{document} 之前的称为导言区。导言区禁止写入任何正文中的语句。

```
\documentclass{ctexart}
\usepackage{amsmath}
```

\title{假装是一个标题} \author{张三} \date{\today}

\begin{document}

\maketitle

极其敷衍的正文。 \[E = m c^2. \]

\end{document}

\documentclass{ctexart} 指定文档类为 ctexart;

\usepackage{amsmath} 使用 amsmath 包,这是一个数学公式相关的使用频率很高的宏包;

\title、\author、\date 分别指定标题、作者、日期。其中,\today 为文档编译时所在的日期。如果不想标注日期的话,可以使用 \date{};

在 \begin{document} 和 \end{document} 之间的称为正文,\begin{document} 之前的称为导言区。导言区禁止写入任何正文中的语句。

\maketitle 也就是在当前位置生成带有格式的 标题。

```
\documentclass{ctexart}
\usepackage{amsmath}
```

\title{假装是一个标题} \author{张三} \date{\today}

\begin{document}

\maketitle

极其敷衍的正文。 \[E = m c^2. \]

\end{document}

\documentclass{ctexart} 指定文档类为 ctexart;

\usepackage{amsmath} 使用 amsmath 包,这是一个数学公式相关的使用频率很高的宏包;

\title、\author、\date 分别指定标题、作者、日期。其中,\today 为文档编译时所在的日期。如果不想标注日期的话,可以使用 \date{};

在 \begin{document} 和 \end{document} 之间的称为正文, \begin{document} 之前的称为导言区。导言区禁止写入任何正文中的语句。

\maketitle 也就是在当前位置生成带有格式的 标题。

看完 lshort-zh-cn!!!

把 .tex 变成 .pdf:编译

• 编辑器提供的编译按钮

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置
 - 功能可能会比较单一

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置
 - 功能可能会比较单一
- 命令行编译

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置
 - 功能可能会比较单一
- 命令行编译
 - 编辑器的按钮本质上是使用命令行编译

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置
 - 功能可能会比较单一
- 命令行编译
 - 编辑器的按钮本质上是使用命令行编译
 - 支持更多编译模式

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置
 - 功能可能会比较单一
- 命令行编译
 - 编辑器的按钮本质上是使用命令行编译
 - 支持更多编译模式
 - More debug-friendly

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置
 - 功能可能会比较单一
- 命令行编译
 - 编辑器的按钮本质上是使用命令行编译
 - 支持更多编译模式
 - More debug-friendly

使用命令行编译

8

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置
 - 功能可能会比较单一
- 命令行编译
 - 编辑器的按钮本质上是使用命令行编译
 - 支持更多编译模式
 - More debug-friendly

使用命令行编译

1. 在文件所在的路径下打开命令行;

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置
 - 功能可能会比较单一
- 命令行编译
 - 编辑器的按钮本质上是使用命令行编译
 - 支持更多编译模式
 - More debug-friendly

使用命令行编译

- 1. 在文件所在的路径下打开命令行;
- 输入 〈编译命令〉〈文件名〉

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置
 - 功能可能会比较单一
- 命令行编译
 - 编辑器的按钮本质上是使用命令行编译
 - 支持更多编译模式
 - More debug-friendly

使用命令行编译

- 1. 在文件所在的路径下打开命令行;
- 2. 输入

〈编译命令〉〈文件名〉

编译命令:tex, latex, pdftex, pdflatex, xetex, xelatex, luatex, lualatex, ...

- 编辑器提供的编译按钮
 - 傻瓜式,简单易上手
 - 通常需要自己预先配置
 - 功能可能会比较单一
- 命令行编译
 - 编辑器的按钮本质上是使用命令行编译
 - 支持更多编译模式
 - More debug-friendly

使用命令行编译

- 1. 在文件所在的路径下打开命令行;
- 2. 输入

〈编译命令〉〈文件名〉

编译命令:tex, latex, pdftex, pdflatex, xetex, xelatex, luatex, lualatex, ...

举个例子:xelatex document.tex

• 字符集超大

· 字符集超大——原生支持 Unicode 的引擎更好

• 字符集超大——原生支持 Unicode 的引擎更好: XHATEX

9

- 字符集超大——原生支持 Unicode 的引擎更好:XzlATzX
- 需要对于汉字输出、标点压缩等方面的支持

- 字符集超大——原生支持 Unicode 的引擎更好:XnMTeX
- · 需要对于汉字输出、标点压缩等方面的支持:ctex 宏集

- 字符集超大——原生支持 Unicode 的引擎更好:XnMTeX
- 需要对于汉字输出、标点压缩等方面的支持:ctex 宏集

推荐的解决方案:

- 字符集超大——原生支持 Unicode 的引擎更好:XnMTeX
- 需要对于汉字输出、标点压缩等方面的支持:ctex 宏集

推荐的解决方案:

• 使用 ctex 宏集 + X_HAT_EX 编译

- · 字符集超大——原生支持 Unicode 的引擎更好:X-MITEX
- 需要对于汉字输出、标点压缩等方面的支持:ctex 宏集

推荐的解决方案:

• 使用 ctex 宏集 + X_HAT_EX 编译(不是 CT_EX 套装)

- 字符集超大——原生支持 Unicode 的引擎更好:X-MITEX
- 需要对于汉字输出、标点压缩等方面的支持:ctex 宏集

推荐的解决方案:

• 使用 ctex 宏集 + XFIATFX 编译(不是 CTFX 套装)

不推荐的解决方案:

- 字符集超大——原生支持 Unicode 的引擎更好:X-MITEX
- 需要对于汉字输出、标点压缩等方面的支持:ctex 宏集

推荐的解决方案:

• 使用 ctex 宏集 + XFIATFX 编译(不是 CTFX 套装)

不推荐的解决方案:

• ctex 宏集 + 其他引擎编译

- 字符集超大——原生支持 Unicode 的引擎更好:X-MITEX
- 需要对于汉字输出、标点压缩等方面的支持:ctex 宏集

推荐的解决方案:

• 使用 ctex 宏集 + XHATEX 编译(不是 CTEX 套装)

不推荐的解决方案:

- ctex 宏集 + 其他引擎编译
- · 裸用 xeCJK 宏包

- 字符集超大——原生支持 Unicode 的引擎更好:X_HXT_EX
- · 需要对于汉字输出、标点压缩等方面的支持:ctex 宏集

推荐的解决方案:

• 使用 ctex 宏集 + XFIATFX 编译(不是 CTFX 套装)

不推荐的解决方案:

- ctex 宏集 + 其他引擎编译
- 裸用 xeCJK 宏包

过时的解决方案:

- 字符集超大——原生支持 Unicode 的引擎更好:X-MITEX
- 需要对于汉字输出、标点压缩等方面的支持:ctex 宏集

推荐的解决方案:

• 使用 ctex 宏集 + XHATEX 编译(不是 CTEX 套装)

不推荐的解决方案:

- ctex 宏集 + 其他引擎编译
- 裸用 xeCJK 宏包

过时的解决方案:

• CJK 宏包(别用!)

9

- 字符集超大——原生支持 Unicode 的引擎更好:X-MITEX
- 需要对于汉字输出、标点压缩等方面的支持:ctex 宏集

推荐的解决方案:

• 使用 ctex 宏集 + XHATEX 编译(不是 CTEX 套装)

不推荐的解决方案:

- ctex 宏集 + 其他引擎编译
- 裸用 xeCJK 宏包

过时的解决方案:

• CJK 宏包(别用!)

9

• 公式都是在数学环境中完成的

- 公式都是在数学环境中完成的
- 数学环境不受外界字体命令控制

- 公式都是在数学环境中完成的
- 数学环境不受外界字体命令控制
- 行内公式

- 公式都是在数学环境中完成的
- 数学环境不受外界字体命令控制
- 行内公式
 - 用两个美元符号括起来: \$…\$

- 公式都是在数学环境中完成的
- 数学环境不受外界字体命令控制
- 行内公式
 - 用两个美元符号括起来:\$…\$
- 独显公式

- 公式都是在数学环境中完成的
- 数学环境不受外界字体命令控制
- 行内公式
 - 用两个美元符号括起来:\$…\$
- 独显公式
 - 不编号:\[…\]

- 公式都是在数学环境中完成的
- 数学环境不受外界字体命令控制
- 行内公式
 - 用两个美元符号括起来:\$…\$
- 独显公式
 - 不编号:\[…\]
 - · 编号: equation 环境, 可以加标签

编辑公式

- 公式都是在数学环境中完成的
- 数学环境不受外界字体命令控制
- 行内公式
 - 用两个美元符号括起来:\$…\$
- 独显公式
 - 不编号:\[…\]
 - 编号: equation 环境, 可以加标签
- amsmath 宏包几乎每次都要使用

编辑公式

- 公式都是在数学环境中完成的
- 数学环境不受外界字体命令控制
- 行内公式
 - 用两个美元符号括起来:\$…\$
- 独显公式
 - 不编号:\[…\]
 - · 编号:equation 环境,可以加标签
- amsmath 宏包几乎每次都要使用
- 看 lshort-zh-cn 第四章!

遭遇问题

1. 看完 lshort-zh-cn;

- 1. 看完 lshort-zh-cn;
- 2. 如果你做到了第 1 点,且对常用的宏包有了基本的认识,则往下看;否则回到第 1 点;

- 1. 看完 lshort-zh-cn;
- 2. 如果你做到了第 1 点,且对常用的宏包有了基本的认识,则往下看;否则回到第 1 点;
- 3. 如果 lshort-zh-cn 里面提供的宏包可以解决你的问题,就直接 texdoc 它;否则继续往下看;
- 4. 向互联网寻求答案

- 1. 看完 lshort-zh-cn;
- 2. 如果你做到了第 1 点,且对常用的宏包有了基本的认识,则往下看;否则回到第 1 点;
- 3. 如果 lshort-zh-cn 里面提供的宏包可以解决你的问题,就直接 texdoc 它;否则继续往下看;
- 4. 向互联网寻求答案:
 - 你的问题很有可能是别人以前就遇到过的

- 1. 看完 lshort-zh-cn;
- 2. 如果你做到了第 1 点,且对常用的宏包有了基本的认识,则往下看;否则回到第 1 点;
- 3. 如果 lshort-zh-cn 里面提供的宏包可以解决你的问题,就直接 texdoc 它;否则继续往下看;
- 4. 向互联网寻求答案:
 - 你的问题很有可能是别人以前就遇到过的——善用搜索

- 1. 看完 lshort-zh-cn;
- 2. 如果你做到了第 1 点,且对常用的宏包有了基本的认识,则往下看;否则回到第 1 点;
- 3. 如果 lshort-zh-cn 里面提供的宏包可以解决你的问题,就直接 texdoc 它;否则继续往下看;
- 4. 向互联网寻求答案:
 - 你的问题很有可能是别人以前就遇到过的——善用搜索
 - 彻底搜索过了,还是没有答案

- 1. 看完 lshort-zh-cn;
- 2. 如果你做到了第 1 点,且对常用的宏包有了基本的认识,则往下看;否则回到第 1 点;
- 3. 如果 lshort-zh-cn 里面提供的宏包可以解决你的问题,就直接 texdoc 它;否则继续往下看;
- 4. 向互联网寻求答案:
 - 你的问题很有可能是别人以前就遇到过的——善用搜索
 - ・彻底搜索过了,还是没有答案——在社区提问 ののの

- 1. 看完 lshort-zh-cn;
- 2. 如果你做到了第 1 点,且对常用的宏包有了基本的认识,则往下看;否则回到第 1 点;
- 3. 如果 lshort-zh-cn 里面提供的宏包可以解决你的问题,就直接 texdoc 它;否则继续往下看;
- 4. 向互联网寻求答案:
 - 你的问题很有可能是别人以前就遇到过的——善用搜索
 - 彻底搜索过了,还是没有答案——在社区提问 ��� 不要匿名提问! 截图而不是拍照! 如果是遇到问题的情况,给 出可以复现问题的最小工作样例(MWE)�

- 1. 看完 lshort-zh-cn;
- 2. 如果你做到了第 1 点,且对常用的宏包有了基本的认识,则往下看;否则回到第 1 点;
- 3. 如果 lshort-zh-cn 里面提供的宏包可以解决你的问题,就直接 texdoc 它;否则继续往下看;
- 4. 向互联网寻求答案:
 - 你的问题很有可能是别人以前就遇到过的——善用搜索
 - 彻底搜索过了,还是没有答案——在社区提问 ��� 不要匿名提问! 截图而不是拍照!如果是遇到问题的情况,给 出可以复现问题的最小工作样例(MWE)�
- 5. 有些问题是无解的,或者问题本身就是错误的。

- 1. 看完 lshort-zh-cn;
- 2. 如果你做到了第 1 点,且对常用的宏包有了基本的认识,则往下看;否则回到第 1 点;
- 3. 如果 lshort-zh-cn 里面提供的宏包可以解决你的问题,就直接 texdoc 它;否则继续往下看;
- 4. 向互联网寻求答案:
 - 你的问题很有可能是别人以前就遇到过的——善用搜索
 - 彻底搜索过了,还是没有答案——在社区提问 ��� 不要匿名提问! 截图而不是拍照!如果是遇到问题的情况,给 出可以复现问题的最小工作样例(MWE)�
- 5. 有些问题是无解的,或者问题本身就是错误的。 例如:如何在数学环境中使用 Times New Roman?如何让浮 动体在指定的位置固定不动?

"常规"的符号, 比如: \mathcal{L} , \mathbb{R} , $\overline{\lim}$, \ll , \doteq , ...

"常规"的符号, 比如: \mathcal{L} , \mathbb{R} , $\overline{\lim}$, \ll , \doteq , ...

• 看 lshort-zh-cn!!!

xx 符号怎么输入?

"常规"的符号, 比如: \mathcal{L} , \mathbb{R} , $\overline{\lim}$, \ll , \doteq , ...

• 看 lshort-zh-cn!!!

lshort 里面也没有, 比如:∯, ∳, ...

"常规"的符号, 比如: \mathcal{L} , \mathbb{R} , $\overline{\lim}$, \ll , \doteq , ...

• 看 lshort-zh-cn!!!

lshort 里面也没有, 比如: ∯, ∳, ...

• 先不要去互联网提问

xx 符号怎么输入?

"常规"的符号, 比如: \mathcal{L} , \mathbb{R} , $\overline{\lim}$, \ll , \doteq , ...

• 看 lshort-zh-cn!!!

lshort 里面也没有, 比如: ∯, ∳, ...

- 先不要去互联网提问
- texdoc comprehensive

"常规"的符号, 比如: \mathcal{L} , \mathbb{R} , $\overline{\lim}$, \ll , \doteq , ...

• 看 lshort-zh-cn!!!

lshort 里面也没有, 比如: ∯, ∳, ...

- 先不要去互联网提问
- texdoc comprehensive

texdoc comprehensive 要是还没有……

"常规"的符号, 比如: \mathcal{L} , \mathbb{R} , $\overline{\lim}$, \ll , $\dot{=}$, ...

• 看 lshort-zh-cn!!!

lshort 里面也没有, 比如: ∯, ∳, ...

- 先不要去互联网提问
- texdoc comprehensive

texdoc comprehensive 要是还没有……

• 恭喜你,可以去互联网提问了!

"常规"的符号, 比如: \mathcal{L} , \mathbb{R} , $\overline{\lim}$, \ll , \doteq , ...

• 看 lshort-zh-cn!!!

lshort 里面也没有, 比如: ∯, ∳, ...

- 先不要去互联网提问
- texdoc comprehensive

texdoc comprehensive 要是还没有……

- 恭喜你,可以去互联网提问了!
- 使用一些「dirty tricks」拼接符号 🔗

"常规"的符号, 比如: \mathcal{L} , \mathbb{R} , $\overline{\lim}$, \ll , \doteq , ...

• 看 lshort-zh-cn!!!

lshort 里面也没有, 比如: ∯, ∳, ...

- 先不要去互联网提问
- texdoc comprehensive

texdoc comprehensive 要是还没有……

- 恭喜你,可以去互联网提问了!
- 使用一些「dirty tricks」拼接符号 🔗
- 甚至可以自己做一个字体

Thank you!