

# *L**A**T**E**X* Start

---

## Install

下载安装 *T**E**X* 发行版看这篇博客足够了: <https://liam.page/texlive/>

## Tutorials

可以先看这篇了解: <https://liam.page/2014/09/08/latex-introduction/>

Overleaf指南: 30分钟*L**A**T**E**X*入门: <https://imztj.cn/?p=3073>

## vscode 配置

- 下载 **Latex workshop** 和 **Latex Language support**
- 打开vscode的setting.json配置文件, 用以下代码 **覆盖** 或 **追加** 。

```
"latex-workshop.view.pdf.viewer": "tab",
"latex-workshop.latex.tools": [
  {
    "name": "xelatex",
    "command": "xelatex",
    "args": [
      "-synctex=1",
      "-interaction=nonstopmode",
      "-file-line-error",
      "-pdf",
      "%DOC%"
    ]
  },
  {
    "name": "latexmk",
    "command": "latexmk",
    "args": [
      "-synctex=1",
      "-interaction=nonstopmode",
      "-file-line-error",
      "-pdf",
      "%DOC%"
    ]
  },
  {
    "name": "pdflatex",
    "command": "pdflatex",
    "args": [
      "-synctex=1",
```

```

        "-interaction=nonstopmode",
        "-file-line-error",
        "%DOC%"
    ]
},
{
    "name": "bibtex",
    "command": "bibtex",
    "args": [
        "%DOCFILE%"
    ]
}
],
"latex-workshop.latex.recipes": [
    {
        "name": "xelatex",
        "tools": [
            "xelatex"
        ]
    },
    {
        "name": "xelatex -> bibtex -> xelatex*2",
        "tools": [
            "xelatex",
            "bibtex",
            "xelatex",
            "xelatex"
        ]
    },
    {
        "name": "pdflatex -> bibtex -> pdflatex*2",
        "tools": [
            "pdflatex",
            "bibtex",
            "pdflatex",
            "pdflatex"
        ]
    }
],
"latex-workshop.latex.clean.fileTypes": [
    "*.aux",
    "*.bbl",
    "*.blg",
    "*.idx",
    "*.ind",
    "*.lof",
    "*.lot",
    "*.out",
    "*.toc",
    "*.acn",
    "*.acr",
    "*.alg",

```

```
    "*.glg",
    "*.glo",
    "*.gls",
    "*.ist",
    "*.fls",
    "*.log",
    "*.fdb_latexmk"
],
```

Hello World,Hello *L<sup>A</sup>T<sub>E</sub>X*.

```
\documentclass{ctexart}
% 导言区
\begin{document}
%正文
Hello World,Hello  $\LaTeX$ .
\end{document}
```

## 命令与环境

---

### 命令

类似HTML的标签，使文本有不同类型展示效果；

```
% 常见语法结构：
cmd[option]{para}{para}

欧几里得\footnote{欧几里得，约公元前 330--275 年。}是一名伟大的数学家。
接下来要强调一段文本\emph{被强调的文本}。
```

### 环境

对特定输入所处的环境约束，如表格，图片，公式，矩阵等

```
% 语法结构
\begin{env}
...
\end{env}

% 如
\begin{document}
...
\end{document}
```

# document

```
\begin{document}

\end{document}

\begin{}

\end{}
```

## 文档结构层次

### 划分章节

根据文档类型不同，有 `report`，`book`，`article(ctexart)`，`letter` 等；

#### article/ctexart

最高层：节

- section
- subsection
- subsubsection

段：

- paragraph
- subparagraph

```
\documentclass{aritle} % -> ctexart
% 导言区
\title{一起学习 \LaTeX{}} \thanks{本文由xxx赞助}}
\author{steven\footnote{xxx大学研究生，方向：深度学习} \\\ steven@163.com \and jobs \\\ jobs@apple.con}
\date{\today{}}

\begin{document}
    \maketitles % 显示导言区内容
    \tableofcontents % 创建目录
    \section{第一节}
    \subsection{1.1 xxxx}
    \subsubsection{1.1.1 xxxx}

    \section{第二节}
    \subsection{2.1 xxxx}
    \subsubsection{2.1.1 xxxx}

    \paragraph{一段文字}
    \subparagraph{子段文字}
```

```
\end{document}
```

## 排版分两栏

直接在该文档类前使用 `twocolumn` 参数：`\documentclass[twocolumn]{ctexart}`，可以使用 `balance` 宏包来平衡两栏内容，导入该包后，需要在 导言区 使用 `\balance` 命令。

```
\documentclass[twocolumn]{ctexart}
% 导言区
\title{一起学习 \LaTeX{}} \thanks{本文由xxx赞助}}
\author{steven\footnote{xxx大学研究生，方向：深度学习} \\\ steven@163.com \and jobs \\\ jobs@apple.com}
\date{\today{}}

% 平衡分栏内容
\usepackage{balance}
\balance

\begin{document}
  \maketitles % 显示导言区内容
  \tableofcontents % 创建目录
  \clearpage % 目录与正文分隔开
  \section{第一节}
  \subsection{1.1 xxxx}
  \subsubsection{1.1.1 xxxx}

  \section{第二节}
  \subsection{2.1 xxxx}
  \subsubsection{2.1.1 xxxx}

  \paragraph{一段文字}
  \subparagraph{子段文字}
\end{document}
```

## 段落与文本环境

### 正文文本

分段需要两个 回车

```
\begin{document}
这是第一段文字。

这是第二段文字。
这还是属于第二段文字。

所以必须输入两个\emph{回车}才能分段。
\end{document}
```

一些转译字符

```
\# \textbackslash \$
```

## 列表环境

Latex标准文档提供三种列表环境：

- 不编号的 **itemize**
- 编号的 **enumerate**
- 使用关键字的 **description**

```
\begin{document}
% itemize
\begin{itemize}
  \item 中文
  \item English
  \item Français
\end{itemize}
% enumerate
\begin{enumerate}
  \item 中文
  \item English
  \item Français
\end{enumerate}
% description
\begin{description}
  \item[中文] 中国的语言文字
  \item[English] The Language of English
  \item[Français] La Langue de France
\end{description}
\end{document}
```

当然，上述的三种环境都可 **独自或者相互嵌套** 使用

## 抄入和代码环境

### 抄入

排版计算机程序源代码时，需要字符等宽，或者需要使用打字机字体，就需要抄入功能。

使用： **\verb**命令 或 **verbatim** 环境

```
% \verb语法
\verb(开始符号)(抄入内容)(结束符号)
\verb|#include<stdio.h>|

% verbatim环境
\begin{document}
  \begin{verbatim}
    # include <stdio.h>
```

```

int main(void){
    printf("hello world.\n");
    return 0;
}
\end{verbatim}
\end{document}

```

可使用 **listings** 宏包和 **minted** 宏包做语法高亮

## 算法结构

对应解决方案的宏包：**clrscode**、**algorithm2e**、**algorithmicx** 下的algpseudocode格式

```

% 算法结构
\begin{document}
    \begin{codebox}
        \Procname{$\proc{Merge-Sort}(A,p,r)$}
        \li \If $p<r$
        \li \Then $q \gets \lfloor (p+r)/2 \rfloor$
        \li $\proc{Merge-Sort}(A,p,q)$
        \li $\proc{Merge-Sort}(A,p+1,r)$
        \li $\proc{Merge-Sort}(A,p,q,r)$
        \End
    \end{codebox}
\end{document}

```

## 注脚

```

\begin{document}
    欧几里得\footnote{欧几里得，约公元前 330--275 年，著作：《几何原本》}是一名伟大的数学家。
\end{document}

```

## 数学公式

数学公式必须在 **数学模式** 下输入，包括单个符号，如： $\pi, n$

## 数学结构

- 上标与下标：分别用  $\wedge$  和  $_$  表示
- 上下划线与花括号：**\overline**，**underline**，**\overbrace**，**underbrace**
- 分式：**\frac{分子部分}{分母部分}**
- 根式：**\sqrt[次数]{根号下内容}**
- 矩阵：一般使用 **amsmath** 宏包提供的 **matrix**，**pmatrix**，**bmatrix** 等矩阵环境，特别复杂的矩阵（如分块矩阵）可以使用 **array** 环境作为表格画出

## 数学符号

- 数学字母：  $a, b, \alpha$ ； 数学字母：  $\mathbb{R}, p$  等
- 普通符号： 如  $\infty, \angle, \int, \iint, \Sigma$
- 二元运算符：  $a + b, a - b$
- 二元关系符：  $a = b, a \leq b$
- 括号：  $\langle a, b \rangle$
- 标点： 逗号， 分号， 冒号等

公式编辑辅助工具：[www.latexlive.com](http://www.latexlive.com) , <https://latex.codecogs.com/eqneditor/editor.php>

## 行内公式

与文字处于同一行

```
% 语法格式
$ 定义的公式 $

\begin{document}
    爱因斯坦提出的质能方程是：  $E=MC^2$ 。

    同样对于 $e=2.17\text{dots}$ ，在数学模式下：  $e=2.17\text{dots}$ 
\end{document}
```

## 显示公式

公式单独在一行

```
% 语法格式
\[ 定义的公式 \] % 不要使用 $$ 定义的公式 $$

\begin{document}
    对于二元一次方程，其求根公式为： 
$$x_{1,2} = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$$

\end{document}
```

- 基本的编号的公式： 使用 `equation` 环境；

```
\begin{document}
    如果两物体质量分别为 $m_1$ 和 $m_2$ ，之间距离为 $r$ ，则根据牛顿的万有引力公式有：
    \begin{equation}
        F_1 = F_2 = \frac{Gm_1m_2}{r^2}
    \end{equation}
\end{document}
```

- 更复杂的结构，使用 `amsmath` 宏包提供的专门的数学环境（不要使用 `eqnarray` 环境）；

```
% 稍微复杂一点的 数学公式 与 矩阵
\usepackage{amsmath}
\begin{document}
    % 数学公式
```



`\emph{稍微复杂的数学公式：}`牛顿-莱布尼茨公式（Newton-Leibniz formula），通常也被称为微积分基本定理，揭示了定积分与被积函数的原函数或者不定积分之间的联系。其定义为：

```
\begin{equation}
\int_a^b f(x) \mathrm{d}x = F(b) - F(a) = F(x) |_a^b
\end{equation}
```

% 矩阵

`\emph{矩阵-单位阵：}`

`\[`

% 有matrix, bmatrix, vmatrix, pmatrix, Bmatrix, Vmatrix等环境应对不同场景需

要

```
\begin{bmatrix}
1 & 0 & \cdots & 0 \\
0 & 1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 1
\end{bmatrix}
```

`\]`

`\end{document}`

## 科技功能

各种单位：使用 `siunitx` 包，提供一揽子解决方案

```
\begin{document}
科学记数法：\num{-1.23e45}
光的速度为：\SI{299752458}{m/s}\
```

不同行数据按小数点对齐：

```
\begin{tabular}{|S|}
\hline
-234234\\13.45\\ .9e37km\\
\hline
\end{tabular}
\end{document}
```

## 数字单位

参考使用：`siunitx` 这个宏包

## 图表和浮动环境

### 表格

在LaTeX中，可用 `tabular` 或 `array` 画表格，但通常使用前者，后者主要排版包含数学符号的公式，如复杂矩阵等。

% 表格语法

```
\begin{tabular}[垂直对齐]{列格式说明}
```

```

    表项 & 表项 & ... & 表项\\
    ...
\end{tabular}

\begin{document}
% 记得演示边框
\begin{tabular}{lcr}
    left & center & right\\
    文本左对齐 & 文本居中对齐 & 文本右对齐
\end{tabular}

\begin{table}[!t]
    \renewcommand{\arraystretch}{1.3}
    \caption{your description on table}
    \label{table_example}
    \centering
    \begin{tabular}{cccc}
        \hline
        \bfseries Do & \bfseries You & \bfseries Love & \bfseries Me\\
        \hline
        Yestoday & Yes & Yes & Yes\\
        Today & Of Course & Of Course & Of Course\\
        Tomorrow & Definitely Yes & Definitely Yes & Definitely Yes\\
        \hline
    \end{tabular}
\end{table}
\end{document}

```

表格生成工具: [www.tablesgenerator.com/latex\\_tables](http://www.tablesgenerator.com/latex_tables)

各种功能的表格宏包:

- 单元格处理: `multirow`, `makecell`
- 长表格: `longtable`, `xtab`
- 定宽表格: `xtabular`
- 表线控制: `booktabs`, `diagbox`, `arydshln`
- 表列格式: `array`
- 综合应用: `tabu`

## 三线表

使用 `booktabs` 宏包

```

% 导入宏包
\usepackage{booktabs}
\begin{document}
    三线表:
    \begin{tabular}{cccc}
        \toprule
        \bfseries Do & \bfseries You & \bfseries Love & \bfseries Me\\
        \midrule
    \end{tabular}

```

```

        Yestoday & Yes & Yes & Yes\\
        Today & Of Course & Of Course & Of Course\\
        Tomorrow & Definitely Yes & Definitely Yes & Definitely Yes\\
        \bottomrule
    \end{tabular}
\end{document}

```

## 插图

使用：`graphicx` 宏包

命令：`\includegraphics[width=2cm]{img_name_path}`

xelatex支持格式：`EPS, PDF, PNG, JPEG, BMP`

```

% 语法格式
\includegraphics[格式]{文件}

\usepackage{graphicx}
\begin{document}
    \includegraphics{img_path}
\end{document}

```

## 浮动体

在LaTeX中，插入 **一个图或者表** 其实是作为 **一个字符** 来使用的

在排版过程中，因图表大都占有较大的一块，直接放在文档位置就可能会造成分页困难，即前一页放不下，放后一页又会造成很大的留白；其他一些大块内容（如：程序算法，大型公式等），也有可能出现这种问题。在  $TeX$  中，使用 **浮动体** 解决。

**浮动体** 另一个重要用途是给图表添加一个标题，LaTeX浮动体环境可为图表标题提供了专门的命令进行自动编号，自动生成目录等功能。

浮动体环境：`figure`、`table`

```

% 浮动体语法格式
\begin{figure}[允许的位置]
    内容
\end{figure}
允许位置：
h(ere)：代码所在上下文
t(op)： 代码所在环境页面或之后页面的顶部
b(ottom)：一页的底部
p(age)：一个或多个浮动体被放在单独的页面中
!：忽略参数，使得位置更加灵活

```

浮动体内常用命令：`\centering`，`\caption{}`，`\label{}`

`\label{}` 命令要放在 `\caption{}` 之后，才能做交叉引用，不然会报错。

```
% caption语法格式
\caption{标题}
\caption[短标题]{长标题}

% label语法格式
\label{自定义标签名字，方便交叉引用}
```

一个典型的完整浮动体图片可以如下：

```
\usepackage{graphicx}
\begin{document}
  \begin{figure}
    \centering
    \includegraphics{img_path}
    \caption[小狮子]{\TeX{} 系统的吉祥物--小狮子}
    \label{fig-lion}
    % \caption[小狮子]{\babel{fig-lion}\TeX{} 系统的吉祥物--小狮子}
  \end{figure}
\end{document}
```

## 标题控制与caption宏包

在排版文档时，我们经常需要修改浮动体标题的字体，间距，对齐方式等，然后 LaTeX2e 内核及标准文档类并没有提供直接修改浮动标题格式的命令，这时就可以使用 `caption` 宏包。

### 双语标题

使用 `\bicaption` 完成需求。

```
% bicaption语法
\bicaption[短标题1]{长标题1}[短标题2]{长标题2}

\begin{document}
  \begin{figure}
    \centering
    Some Table or Figure
    \bicaption{中文标题}{English title}
    \label{fig-lion}
  \end{figure}
\end{document}
```

## 并排与子图表

实际中，可能需要把几个图表并列放在一起输出，因为浮动环境并不对环境内容加以限制，所以只要直接把图表放在一个浮动体里面就可以了。

## 两个表并列基本形式

```
\begin{document}
  \begin{table}[h]
    \centering
    \caption{并排的表格}
    \begin{tabular}{|c|c|}
      \hline
      图 & 表 \\
      \hline
    \end{tabular}
    \quad
    \begin{tabular}{|c|c|}
      \hline
      Figure & Table \\
      \hline
      A & B \\
      \hline
    \end{tabular}
  \end{table}
\end{document}
```

## 两个表并列常见形式

需要给整个浮动体加一个概括性的标题，同时对浮动体内的每个子图表也可以自己编号和加标题。这时可以使用 `caption` 宏包的一个附加宏包 `subcaption` 来完成。

```
\usepackage{caption, subcaption}
\begin{document}
  \begin{table}
    \caption{对两个图表的综合性描述内容}
    \begin{subtable}[b]{.5\textwidth}
      \centering
      \begin{tabular}{|c|c|}
        \hline
        图 & 表 \\
        \hline
      \end{tabular}
      \caption{文字表格}
    \end{subtable}

    \begin{subtable}[b]{.5\textwidth}
      \centering
      \begin{tabular}{|c|c|}
        \hline
        e & 2.71 \\
        \hline
        $\pi$ & 3.14 \\
        \hline
      \end{tabular}
      \caption{数字表格}
    \end{subtable}
  \end{table}
```

```
\end{table}
\end{document}
```

## 图片并列

```
\begin{figure}[htbp]
  \centering      %居中

  \subfigure[name of the subfigure] %第一张子图
  {
    \begin{minipage}{7cm}
      \centering      %子图居中
      \includegraphics[scale=0.5]{pic1.jpg}    %以pic.jpg的0.5倍大小输出
    \end{minipage}
  }

  \subfigure[name of the subfigure] %第二张子图
  {
    \begin{minipage}{7cm}
      \centering      %子图居中
      \includegraphics[scale=0.8]{fig2.jpg}    %以pic.jpg的0.5倍大小输出
    \end{minipage}
  }

  \caption{name of the figure} %  %大图名称
  \label{fig:1}    %图片引用标记
\end{figure}
```

<https://www.zhihu.com/search?type=content&q=Latex%E5%9B%BE%E7%89%87>

## 参考文献

### 直接引用

在文章 **参考文献** 部分将本文所有引用的文献列出来，然后引用，这个比较麻烦，需要自己调整字体，引用风格啥的。

### 使用BibTex文献数据库

步骤：

1. 对应位置创建 **.bib** 文件
2. 在 **.tex** 文件里使用

```
\documentclass{ctexart}

\bibliographystyle{plain}

\begin{document}
  引用文献处 \cite{...}。

  \nocite{*}
  \bibliography{cite_file}
\end{document}
```

利用BibTex的方法引入文献：

分为以下四步（即需要四步编译）：

1. 用LaTex编译写好的 .tex 文件，这是会生成一个 .aux文件，这是告诉BibTex将使用哪些引用
2. 用BibTex编译 .aux文件（有些仅编译 .bib文件，也可以）
3. 再次用LaTex编译 .tex 文件，这是文档已经插入了参考文献，但是引用的编号不对
4. 最后在用LaTex编译一次 .tex文件，这是会自动生成正确的引用顺序（引用顺序是与在文章中引用文献的先后顺序有关）

可以了解文献管理工具：**JabRef**

## 期刊模版使用

### 以IEEE Access 为例

下载模版：[http://www.ieee.org/publications\\_standards/publications/authors/author\\_templates.html](http://www.ieee.org/publications_standards/publications/authors/author_templates.html)