

# 笨办法学 L<sup>A</sup>T<sub>E</sub>X

Liam Huang

2016 年 10 月 4 日



# 前言

现在市面上已经出版的中文  $\text{\LaTeX}$  入门书有胡伟老师的《 $\text{\LaTeX} 2_{\epsilon}$  完全学习手册》，还有刘海洋前辈的《 $\text{\LaTeX}$  入门》；网络上流传的中文入门资料，比较优秀的，则有黄新刚的《 $\text{\LaTeX}$  笔记》、台湾李果正老师的《大家来学  $\text{\LaTeX}$ 》以及不才编写的一份其实很短的  $\text{\LaTeX}$  入门文档。这些入门资料都不错，但任然不断有同好表示  $\text{\LaTeX}$  入门难。这让我感到，仅仅有这些资料是不够的，遂动了再编写一份对新人更友好的入门资料的念头。

这本书的灵感来自于 Zed A. Shaw 所著的《Learn Python the Hard Way》。在这本书中，Shaw 精心设计了一系列的练习题，将 Python 的基本概念穿插其中，让读者在练习的过程中，逐渐上手 Python。诚然，这本书没有完全介绍 Python 的所有语言特性，也没有涉及到太多的用法技巧。但是，Shaw 通过练习，将编程的精要娓娓道来。我相信，哪怕是对编程毫无所知的人，仔细阅读并认真实践了书中每一个练习后，都能具备基础的代码编写能力。

如 Shaw 在书中所言，所谓「笨方法」指得是教授的方式，而不是教授的内容。在阅读本书的时候，你需要

1. 做每一道习题；
2. 一字不差地写出每一份手稿；
3. 编译手稿，得到正确的结果。

刚开始这对你来说会非常难，但你需要坚持下去。如果你通读了这本书，每晚花个一两小时做做习题，你可以为自己读下一本  $\text{\LaTeX}$  书籍打下良好的基础。通过这本书，你不可能学到  $\text{\LaTeX}$  的方方面面，但是将能习得最基本的学习方法。

这本书的目的是教会你  $\text{\LaTeX}$  新手所需的三种最重要的技能：读和写、注重细节、发现不同。

## 读和写

在  $\text{\LaTeX}$  中，有许多形如  $\text{\code{\command[⟨oarg⟩}{⟨marg⟩}}$  的控制序列（命令）。显然，如果你连这些带着特殊符号的命令都打不出来，那就别想学好  $\text{\LaTeX}$  了。

为了让你记住各种符号的名字并对它们熟悉起来，你需要将代码写下来并且运行起来。这个过程也会让你对  $\text{\LaTeX}$  更加熟悉。

## 注重细节

注重细节的程度，几乎是任何行业评判雇员能力的通用标准。是否注重细节以及注重细节的程度，决定了你是否能顺利从本书毕业，也决定了你编写的手稿排版出来的效果。如果你不够重视细节，那么，也许你能够用  $\text{\LaTeX}$  排版出一些手稿，但是在排版的要求上，你的手稿可能问题百出。

你需要将本书里的示例一字不差地打出来。通过这样的实践，你才能训练自己将精力集中在细节上的能力。

## 发现不同

在你对着示例或者做练习的时候，不可避免地，你会打错一些命令。我希望你不会因为频繁地见到错误提示而沮丧——要知道，这是不可避免的，哪怕是十分有经验的  $\text{\LaTeX}$  使用者，也会出错。你的任务，是把自己打的东西和示例/标准答案做对比，找出任何细微的不同，然后把所有的差异都修正好。

这样的练习，会让你对手稿里的错误和缺陷更加敏感。

## 不要复制粘贴

复制粘贴本书提供的代码，就和学生时代抄袭作业一样，是一种自欺欺人的行为。我希望你不要这样做。

本书的练习都经过了仔细的设计，目的是锻炼你的大脑和双手，让你有能力读代码、理解代码、边写代码。如果复制粘贴本书提供的代码，那么一切就失去了意义。

## 许可协议

版权所有侵权必究 (C) 2016 by Liam Huang.

你可以在不收取任何费用，而且不修改任何内容的前提下自由分发这本书给任何人。但是本书的内容只允许完整原封不动地进行分发和传播。也就是说如果你用这本书给人上课，只要你不向学生收费，而且给他们看的书是完整未加修改的，那就没问题。



# 第一章 第一篇手稿

我们假设你已经安装好了 TeX 系统，现在我们来看一下你接触到的第一份 L<sup>A</sup>T<sub>E</sub>X 代码。

## 1.1 代码和解释

---

代码清单 1.1 Hello L<sup>A</sup>T<sub>E</sub>X

---

```
1 \documentclass[a4paper, 12pt]{article}
2 \title{My First Manuscript}
3 \author{Liam Huang}
4 \date{\today}
5 \begin{document}
6 \maketitle
7 % THIS LINE IS LEFT BLANK INTENTIONALLY.
8
9 Hello \LaTeX. I have a mathematics equation here:  $E = mc^2$ .
10 \end{document}
```

---

`\documentclass[a4paper, 12pt]{article}` 是我们接触到的第一个 L<sup>A</sup>T<sub>E</sub>X 命令。所谓 L<sup>A</sup>T<sub>E</sub>X 命令，是指源代码中形如 `\<命令>[<可选参数>]{<必选参数>}` 的字符串。它们以一条反斜线 `\` 开头，之后跟着一个符号或者一串英文字母；可以接受若干个参数，其中用花括号包裹的是 `{<必选参数>}`，用方括号包裹的是 `[<可选参数>]`。这里我们遇到的是 `\documentclass` 命令，它的作用是载入必选参数中指定的文档类，并接受方括号中的可选参数来调整文档类的效果。这里我们载入的是 `article` 文档类，它将以 `12pt` 的字号，将内容打印在 `a4paper` 上。

所谓文档类，即是文档的类型。它定义了文档的基本格式，储存在 `.cls` 文件当中。例如此处使用的 `article` 文档类，就储存在 `article.cls` 当中。载入文档类，其实就是将 `.cls` 文件载入内存的过程。

如果你使用的是 TeX Live, 那么可以在命令行中执行 `kpsewhich article.cls` 找到 `article.cls` 的具体为止。

接下来, 我们遇到了 `\title`、`\author` 和 `\date` 三个命令。显而易见, 他们分别是用来设置文章的标题、作者以及写作日期的命令。注意, 遵循内容与格式分离的原则, 在这三个命令的参数中, 我们应当只填写标题、作者以及日期的实际内容, 而不应包含它们的格式 (如字体、字号等)。这些格式设置, 应该用额外的方式来控制。

控制文章标题、作者和日期格式的代码, 在 `\@maketitle` 中。

再接着, 我们遇到了 `\begin{document}` 命令, 它与 `\end{document}` 配对, 组成了 `document` 环境。这是 L<sup>A</sup>T<sub>E</sub>X 的正文环境, 文档所有的内容都应该放在这个环境中。

每个 L<sup>A</sup>T<sub>E</sub>X 手稿, 都应该用 `\documentclass` 载入文档类, 并将内容写在 `document` 环境当中。

`\begin{document}` 之前的部分, 是导言区。导言区通常用来做一些对文档格式的设置。`\end{document}` 之后的部分, 会被 L<sup>A</sup>T<sub>E</sub>X 忽略。

在正文中, 我们首先看到 `\maketitle` 命令。这个命令的作用, 是以既定格式将文章的标题、作者和日期等内容打印出来。

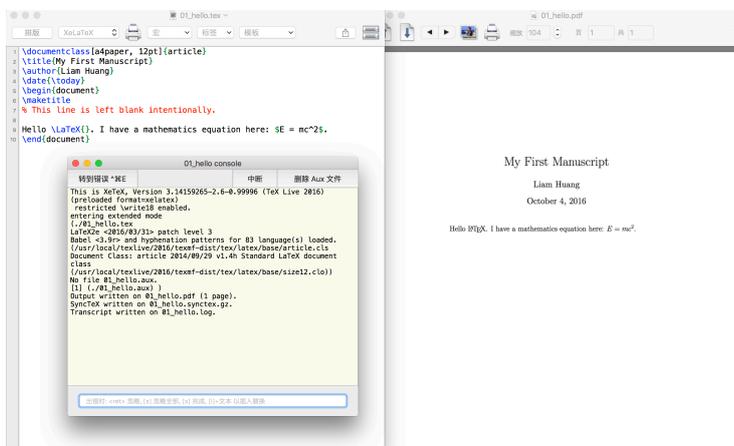
随后我们看到一行以 `%` 开头的行。在 L<sup>A</sup>T<sub>E</sub>X 中, `%` 是注释符号。意思是, 从 `%` 开始, 到 `%` 所在的行结尾, 这部分内容都会被 L<sup>A</sup>T<sub>E</sub>X 忽略。而将它们写在手稿里的目的, 是提示「人们」这段代码的作用是什么。

最后, 我们的目光看到倒数第二行的内容。这一行的内容, 会被 L<sup>A</sup>T<sub>E</sub>X 打印出来。不过, 其中还有两个值得一提的地方。其一是 `\LaTeX` 命令。从这个命令的名字就可以看出, 它的作用是打印 L<sup>A</sup>T<sub>E</sub>X 这个高低不平的 logo。其二是用 `$ ... $` 符号包裹起来的质能公式。因为公式总是价值连城, 所以我们用美元符号将他们包裹起来。

## 1.2 操作和输出

请将代码清单 1.1 中的内容, 用键盘逐字母地输入进电脑, 并保存在文件中 `01_hello.tex`。注意, 这个习惯很重要: L<sup>A</sup>T<sub>E</sub>X 代码应该以 `.tex` 结尾。点击「排版」, 以 X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X 编译, 你应该看到如图 1.1 所示的样子。

如果你看见类似 1.2 的输出, 这表示你在源代码里写错了一些东西。请不要慌张, 你在学习的过程中会经常遇到类似的问题, 最重要的是学会如何阅读错误提示。

图 1.1: Hello L<sup>A</sup>T<sub>E</sub>X

---

**代码清单 1.2 错误输出**


---

```

1 ! Undefined control sequence.
2 1.9 Hello \LaTeX
3      . I have a mathematics equation here: $E = mc^2$.
4 ?

```

---

我们一行一行地看。

1. 以一个叹号开头，说明 L<sup>A</sup>T<sub>E</sub>X 遇到的错误的名字。这里是「未定义的控制序列」，这个错误通常表示你输错了命令，或者使用了一个未经定义的命令。
2. 1.9 表示问题出在第九行，Hello \LaTeX 表示 L<sup>A</sup>T<sub>E</sub>X 在处理到这里的时候，遇到了问题。
3. 第三行是遇到问题时，后续尚未处理的内容。
4. L<sup>A</sup>T<sub>E</sub>X 打印了一个问号，这表示 L<sup>A</sup>T<sub>E</sub>X 正在等待我们的输入。如果你输入 x 则会终止 L<sup>A</sup>T<sub>E</sub>X 进程。

这里显而易见，我们将 \LaTeX 错误地写作了 \LaTeX。注意，L<sup>A</sup>T<sub>E</sub>X 对命令是区分大小写的。这里我们将之更正就可以了。

### 1.3 额外习题

你需要完成这些额外习题，帮助你巩固学到的知识。如果你觉得这部分习题很难，可以先看后面的内容——但一定要记着回来，解决这些问题。

1. 使用搜索引擎（推荐使用 Google），看看 `article` 文档类还有什么可选参数。修改你的手稿，实际看看这些参数有什么效果。

2. 删除或者使用 `%` 注释掉 `\maketitle` 命令，看看会发生什么。思考一下背后的原因。
3. 尝试打印出更多的内容。