

# Introduction to $\text{\LaTeX}$

## Lecture 1: Hello, $\text{\LaTeX}$

Liu Yihao

SJTU-UMJI Technology Department

March 5, 2019

- 1 Getting Started
  - What is  $\text{\LaTeX}$
  - Installation of  $\text{\LaTeX}$
  - Selection of IDEs
  - $\text{\LaTeX}$  Online
  - Documentation
- 2 Layout of the Document
- 3 Document Body
- 4 Learn more - syntax of  $\text{\LaTeX}$

# What is L<sup>A</sup>T<sub>E</sub>X

From Wikipedia, the free encyclopedia<sup>1</sup>

L<sup>A</sup>T<sub>E</sub>X (lah-tekh, lah-tek or lay-tek, a shortening of Lamport T<sub>E</sub>X) is a document preparation system. When writing, the writer uses plain text in markup tagging conventions to define the general structure of a document (such as [article](#), [book](#), and [letter](#)), to stylize text throughout a document (such as **bold** and *italic*), and to add citations<sup>1</sup> and cross-references.

A T<sub>E</sub>X distribution such as T<sub>E</sub>XLive or MikT<sub>E</sub>X is used to produce an output file (such as PDF or DVI) suitable for printing or digital distribution.

Within the typesetting system, its name is stylized as L<sup>A</sup>T<sub>E</sub>X.

---

<sup>1</sup>L<sup>A</sup>T<sub>E</sub>X - <https://en.wikipedia.org/wiki/Latex>

# A brief History of T<sub>E</sub>X

Donald Kunuth from Stanford University is the specialist in programming art. In year 1977, he had just received his first samples from the new typesetting system of the publisher's, and its quality was so far below that of the first edition of Volume 2 that he couldn't stand it. Kunuth decided to implement a mathematical composition system by himself (since he is a computer scientist). He figured that this would take about 6 months (Ultimately, it took nearly 10 years). The system is named as T<sub>E</sub>X, of both the meaning of Greek letters  $\tau\epsilon\chi$ , and "technical".

L<sup>A</sup>T<sub>E</sub>X was created in 1983 by Leslie Lamport, when he was working at SRI. He needed to write T<sub>E</sub>X macros for his own use, and thought with a little extra effort he could make a general package usable by others. Then L<sup>A</sup>T<sub>E</sub>X developed rapidly and now there are thousands of packages written in T<sub>E</sub>X macros available for direct usage.

# Installation of $\text{\LaTeX}$

Though there are some other distributions of  $\text{\LaTeX}$  (like Mik $\text{\TeX}$ ),  $\text{\TeX}$ Live is recommended in this lecture.

## Windows & Linux

Download  $\text{\TeX}$ Live on the [tuna mirrors](https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/Images/)

<https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/Images/>

## MacOS

Download Mac $\text{\TeX}$  on the [tuna mirrors](https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/mac/mactex/)

<https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/mac/mactex/>

## Linux (Debian/Ubuntu)

Enter the command (fast with apt source mirror)

```
sudo apt-get install texlive-full
```

# Selection of IDEs

There are various IDEs recommended that support  $\text{\LaTeX}$ , for example

Texmaker

<http://www.xmlmath.net/texmaker/>

Sublime Text

<http://www.sublimetext.com/>

Follow the instructions on <https://www.zhihu.com/question/36038602>

Visual Studio Code

<https://code.visualstudio.com/>

Follow the instructions on <https://zhuanlan.zhihu.com/p/38178015>

They all have cross-platform support for Windows, Linux and MacOS.

# Write L<sup>A</sup>T<sub>E</sub>X on Overleaf

Another alternative choice is to write L<sup>A</sup>T<sub>E</sub>X online with the technology of [Overleaf](#). It's free for personal usage and supports share editing which is very useful in group work.

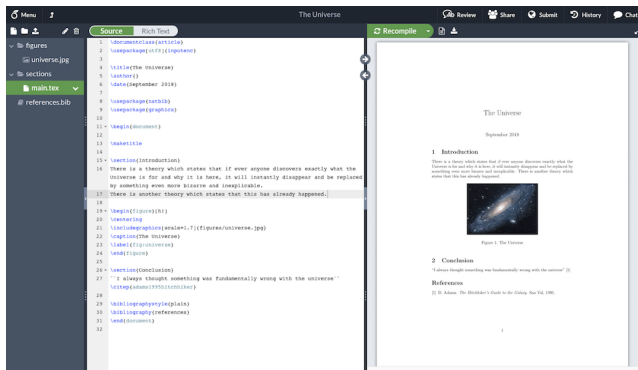


Figure: Layout of the Overleaf Online L<sup>A</sup>T<sub>E</sub>X Editor.

# Documentation on your computer

If you've installed a full version of T<sub>E</sub>XLive (as strongly recommended), the L<sup>A</sup>T<sub>E</sub>X documentation about all you want to is in front of you.

Open the command line and input the command

```
texdoc <docname>
```

You can also use the online version <http://www.texdoc.net/>

For example, you can use the following types for the `docname`

`tex` about T<sub>E</sub>X

`article` about documentclass `article`

`beamer` about documentclass `beamer` (used to create slides)

`pgf` about TikZ and PGF (used to draw graphs)

Try to `texdoc` about all new things and then you'll be an expert in L<sup>A</sup>T<sub>E</sub>X.



- 1 Getting Started
- 2 Layout of the Document
  - A simple document
  - Basic document classes
  - Use packages
  - Title page
- 3 Document Body
- 4 Learn more - syntax of  $\text{\LaTeX}$

# A simple document

A typical (simplest)  $\text{\LaTeX}$  example is presented here.

## Example

```
1  \documentclass[a4paper]{article}
2  \usepackage{amsmath} % Define various maths environments
3  \usepackage{amssymb} % Define various maths symbols
4  \usepackage{geometry} % Adjust the margin, paper size, and etc.
5  \usepackage{enumerate} % Provide different style of lists
6  \usepackage{graphicx} % Insert image of all types
7  % Use other packages and setup them here
8  \title{A simple \LaTeX\ document}
9  \author{XX XXX}
10 \date{\today}
11
12 \begin{document}
13     \maketitle
14     Hello, \LaTeX !
15 \end{document}
```

# All begins with documentclass

## Definition

In a  $\text{\LaTeX}$  file, the **first** line must be

```
\documentclass[options]{class}
```

For example, you can use the following types for the **class**

**article** Write a report or an science article

**report** Write a report

**beamer** Produce a lecture silde like this!

Some options can be added, for example, a typical case can be

```
\documentclass[11pt,twoside,a4paper]{article}
```

Some details about the **article** class will be introduced on the next page.  
More features about other classes and options can be found in the  $\text{\LaTeX}$  Document on your own.

# The article class

The `article` class is the most basic class in  $\text{\LaTeX}$ , it provides you with some normalized structure and format for report writing. So usually you will use the following command as the first line of your tex document

```
\documentclass[options]{article}
```

Some of the options values are listed below (the default values are **alerted**)

- **10pt**, **11pt**, **12pt** or other sizes - the font size of the document
- **a4paper**, **a5paper**, **letterpaper** - the size of paper
- **fleqn** - make the math equations left aligned (default middle aligned)
- **leqno** - display the serial numbers of math equations on the left (default on the right)
- **titlepage**, **notitlepage** - whether to make the title an entire page
- **onecolumn**, **twocolumn** - the number of columns of the document
- **twoside**, **oneside** - influence the position of something on the page

# Other classes

Read the source code of this project and you will learn much about the `beamer` class.

When writing a `long` report, `report` class can be used to provide some more layers of document and different type settings. It's very similar to the `article` class, so it won't be specified.

There are some other document classes such as `book`, `letter` and etc., but I think you will never use them.

# Magic of packages

L<sup>A</sup>T<sub>E</sub>X is a macro-based language, where most of useful commands are not built-in commands. These commands are defined in various packages, which should be included between `\documentclass` and `\begin {document}`.

## Command

```
\usepackage[options]{package}
```

There are some very useful packages that you may **ALWAYS** include:

`amsmath` Define various maths environments

`amssymb` Define various maths symbols

`geometry` Adjust the margin, paper size, and etc.

`enumerate` Generate a list like this!

`graphicx` Insert image of all types

The usages of these and more packages will be introduced further.

# Common packages

Here I provide a list of common used packages, you can start from using the commands of them after the lecture.

```

1  \usepackage{hyperref}      % Extensive support for hypertext
2  \usepackage{float}         % Improved interface for floating objects
3  \usepackage[margin=2.5cm]{geometry} % Flexible document
   ↪ dimensions
4  \usepackage{enumerate}     % Enumerate with redefinable labels
5  \usepackage{multirow}      % Tabular cells spanning multiple rows
6  \usepackage{multicol}      % Intermix single and multiple columns
7  \usepackage{ulem}          % Package for underlining
8  \usepackage{graphicx}      % Enhanced support for graphics
9  \usepackage{subfig}         % Figures broken into subfigures
10 \usepackage{amsmath}       % AMS mathematical facilities
11 \usepackage{amssymb}       % AMS symbols
12 \usepackage{amsfonts}      % AMS fonts
13 \usepackage{mathrsfs}      % Support for using RSFS fonts in maths
14 \usepackage{latexsym}      % LaTeX symbols
15 \usepackage{verbatim}      % Reimplementation of LaTeX verbatim

```

# Title, Author and Date

It's very useful to generate a title on the first page of a document, in order to achieve it, these commands should first be added between

`\documentclass` and `\begin{document}`.

## Example

```
1 \title{title}
2 \author{author name}
3 \date{\today}
```

You can simply use `\date{\today}` to display your system date now.

Then in the `document environment` (will be introduced in the next section), use the command `\maketitle` to generate the title page.



- 1 Getting Started
- 2 Layout of the Document
- 3 Document Body**
  - Body
  - Layers
- 4 Learn more - syntax of  $\text{\LaTeX}$

# Main body of document

The main body of your document which starts with `\begin{document}` and ends with `\end{document}` is called the `document` environment. All of the contents you'd like to display should be in it, and it **MUST** be **unique** in the whole file.

## Example

```
1 \begin{document}
2   \maketitle
3   \newpage    %start the following contents in a new page
4   \tableofcontents %automatically generate a table of content
5   \newpage
6   Hello, \LaTeX !
7   % TODO: Add more contents
8   ...
9 \end{document}
```

The position and order of title page and table of contents can be arbitrary, and there can be multiple table of contents in one document.

# Dividing into sections

## Command

```
1 \section{name}  
2 \subsection{name}  
3 \subsubsection{name}
```

```
1 \section*{name}  
2 \subsection*{name}  
3 \subsubsection*{name}
```

The default style of sections is like

1 Example Section Name

1.1 Example Subsection Name

1.1.1 Example Subsubsection Name

If a star(\*) is added, the sequence number will be hidden, and it won't be added to the table of contents.

**Note:** (Sub)sections can be sorted into commands, not environments, so it doesn't have `\begin` and `\end` clauses. However, the whole contents between two (sub)sections is belonged to the former (sub)section.

# Other structures - Chapter, Part and Paragraph

## Command

```
1 \chapter{name}  
2 \part{name}  
3 \paragraph{name}  
4 \subparagraph{name}
```

```
1 \chapter*{name}  
2 \part*{name}  
3 \paragraph*{name}  
4 \subparagraph*{name}
```

In document classes such as `report` and `book`, some outer structures of section ( `\chapter` and `\part` ) can be used.

`\paragraph` and `\subparagraph` are used for the title of small paragraphs in a (sub)section.

If a star(\*) is added, the effect will be the same as in the sections (sequence numbers will be hidden).

- 1 Getting Started
- 2 Layout of the Document
- 3 Document Body
- 4 Learn more - syntax of  $\text{\LaTeX}$** 
  - Command
  - Environment

# The common syntax of $\text{\LaTeX}$ commands

All  $\text{\LaTeX}$  commands have the following syntax

$\backslash\text{commandName}$ <specialArgs>[optionalArgs]{requiredArgs}

**specialArgs** Seldom used in basic usage, for certain special usages in some packages

**optionalArgs** Used to define mode of the command, if not specified,  $\text{\LaTeX}$  will use the default mode

**requiredArgs** Must be filled

If you want to connect a letter after a command, a space must be appended after the command or  $\text{\LaTeX}$  won't be able to compile it correctly. But two commands can be directly connected since there is a  $\backslash$  before each command.

# The common syntax of $\text{\LaTeX}$ environments

All  $\text{\LaTeX}$  environments have the following syntax

```
1  \begin{environmentName}<specialArgs>[optionalArgs]{requiredArgs}
2      % ...
3  \end{environmentName}
```

$\text{specialArgs}$ ,  $\text{optionalArgs}$ ,  $\text{requiredArgs}$  are similar to those in a [command](#)

It is recommended to have a tab indent in each environment or your tex codes will be difficult to read by others or even **yourself**.