

Introduction to \LaTeX

Tejas Sanap

December 28, 2019

How do we write a document?

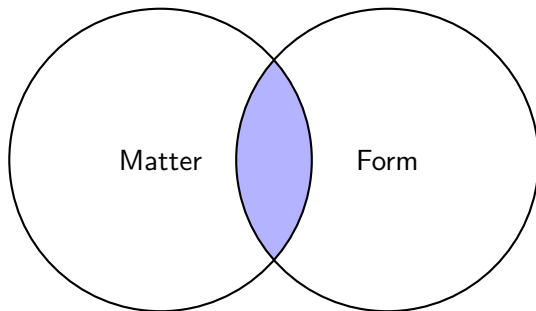
The process of creating a document comprises of two components:

- ▶ Deciding **what** the content **should be**...
- ▶ Deciding **how** the content **should look**...

How do we write a document?

The process of creating a document comprises of two components:

- ▶ Deciding **what** the content **should be**...
- ▶ Deciding **how** the content **should look**...



Type-setting?

What is type-setting?

Type-setting?

What is type-setting?

- ▶ The process of arranging the various objects on a page.
- ▶ It is process that takes place after the manuscript has been written.

Type-setting?

What is type-setting?

- ▶ The process of arranging the various objects on a page.
- ▶ It is process that takes place after the manuscript has been written.

How do we typeset?

MS WORD!

Type-setting?

What is type-setting?

- ▶ The process of arranging the various objects on a page.
- ▶ It is process that takes place after the manuscript has been written.

How should we typeset?

L^AT_EX!

Type-setting?

What is type-setting?

- ▶ The process of arranging the various objects on a page.
- ▶ It is process that takes place after the manuscript has been written.

Unlike WYSIWYG softwares like MS Word and Open Office, \LaTeX separates the process of typesetting from the process of inserting content.

So, what exactly is \LaTeX ?

- ▶ \LaTeX is a general type-setting system.

So, what exactly is \LaTeX ?

- ▶ \LaTeX is a general type-setting system.
- ▶ It allows markup to describe the structure of the document, so that the user does not have to think about it.

So, what exactly is \LaTeX ?

- ▶ \LaTeX is a general type-setting system.
- ▶ It allows markup to describe the structure of the document, so that the user does not have to think about it.
- ▶ Thus, by using document classes and additional packages the same document can be generated in several different formats.

So, what exactly is \LaTeX ?

- ▶ \LaTeX is a general type-setting system.
- ▶ It allows markup to describe the structure of the document, so that the user does not have to think about it.
- ▶ Thus, by using document classes and additional packages the same document can be generated in several different formats.
- ▶ \LaTeX is technically a \TeX macro package.

So, what exactly is \LaTeX ?

- ▶ \LaTeX is a general type-setting system.
- ▶ It allows markup to describe the structure of the document, so that the user does not have to think about it.
- ▶ Thus, by using document classes and additional packages the same document can be generated in several different formats.
- ▶ \LaTeX is technically a \TeX macro package.
- ▶ In computer-science-y terms, \TeX is a *macroprocessor*.

What makes \LaTeX different from WYSIWYG text processors?

- ▶ WYSIWYG systems make two claims:

What makes \LaTeX different from WYSIWYG text processors?

- ▶ WYSIWYG systems make two claims:
 1. You type what you want to print.

What makes \LaTeX different from WYSIWYG text processors?

- ▶ WYSIWYG systems make two claims:
 1. You type what you want to print.
 2. You see on your screen the closest approximation of what will finally be printed.

What makes \LaTeX different from WYSIWYG text processors?

- ▶ WYSIWYG systems make two claims:
 1. You type what you want to print.
 2. You see on your screen the closest approximation of what will finally be printed.
- ▶ However, \LaTeX is not a WYSIWYG text processor.

What makes \LaTeX different from WYSIWYG text processors?

- ▶ WYSIWYG systems make two claims:
 1. You type what you want to print.
 2. You see on your screen the closest approximation of what will finally be printed.
- ▶ However, \LaTeX is not a WYSIWYG text processor.
- ▶ Instead, \LaTeX defines a logical model of a document.

What makes \LaTeX different from WYSIWYG text processors?

- ▶ WYSIWYG systems make two claims:
 1. You type what you want to print.
 2. You see on your screen the closest approximation of what will finally be printed.
- ▶ However, \LaTeX is not a WYSIWYG text processor.
- ▶ Instead, \LaTeX defines a logical model of a document.
- ▶ Where, each object is labelled for what it is, *not what it should look like*.

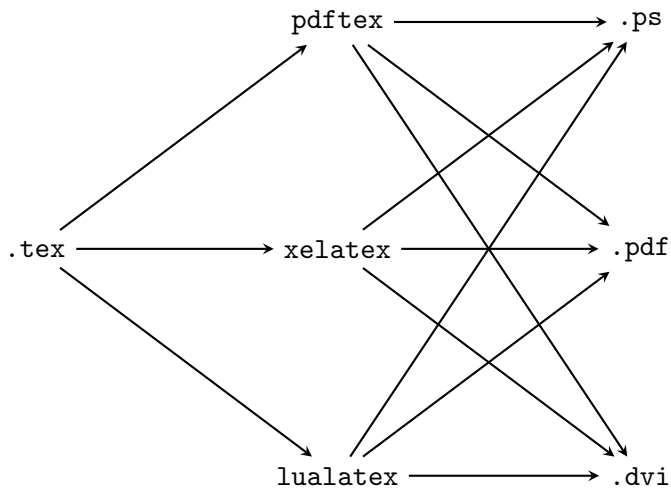
What makes \LaTeX different from WYSIWYG text processors?

- ▶ WYSIWYG systems make two claims:
 1. You type what you want to print.
 2. You see on your screen the closest approximation of what will finally be printed.
- ▶ However, \LaTeX is not a WYSIWYG text processor.
- ▶ Instead, \LaTeX defines a logical model of a document.
- ▶ Where, each object is labelled for what it is, *not what it should look like*.
- ▶ This is the role that "markup" plays.

How do I use \LaTeX ?

- ▶ A \LaTeX file or source code is always a plaintext file, that is processed by a \TeX engine.
- ▶ This is one of the most important features of \LaTeX as it frees the user using a particular version or text editor.

How do I use L^AT_EX?

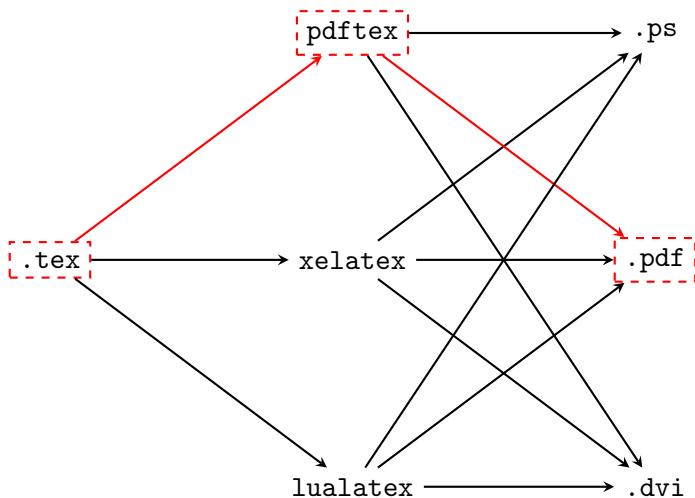


Source Code

L^AT_EX engine

Final Output

How do I use \LaTeX ?



Source Code

\LaTeX engine

Final Output

How many T_EX engines are available?

There are multiple T_EX engines:

How many $\text{T}_{\text{E}}\text{X}$ engines are available?

There are multiple $\text{T}_{\text{E}}\text{X}$ engines:

- ▶ **Knuth's $\text{T}_{\text{E}}\text{X}$:**

This is original $\text{T}_{\text{E}}\text{X}$ engine which serves as the lowest layer of \LaTeX 's software architecture.

How many $\text{T}_{\text{E}}\text{X}$ engines are available?

There are multiple $\text{T}_{\text{E}}\text{X}$ engines:

- ▶ **Knuth's $\text{T}_{\text{E}}\text{X}$:**

This is original $\text{T}_{\text{E}}\text{X}$ engine which serves as the lowest layer of \LaTeX 's software architecture.

- ▶ **pdf $\text{t}_{\text{E}}\text{X}$:**

This engine adds a bunch of primitives related to the PDF and DVI extension.

How many $\text{T}_{\text{E}}\text{X}$ engines are available?

There are multiple $\text{T}_{\text{E}}\text{X}$ engines:

- ▶ **Knuth's $\text{T}_{\text{E}}\text{X}$:**

This is original $\text{T}_{\text{E}}\text{X}$ engine which serves as the lowest layer of \LaTeX 's software architecture.

- ▶ **pdf_{tex} :**

This engine adds a bunch of primitives related to the PDF and DVI extension.

- ▶ **xetex :**

This engine provides better font support.

How many T_EX engines are available?

There are multiple T_EX engines:

- ▶ **Knuth's T_EX:**

This is original T_EX engine which serves as the lowest layer of L^AT_EX's software architecture.

- ▶ **pdf_tex:**

This engine adds a bunch of primitives related to the PDF and DVI extension.

- ▶ **xetex:**

This engine provides better font support.

- ▶ **lua_tex:**

Originally, meant to replace pdf_tex, but, now moving in a very different direction. This engine also better font support (like, xel_atex) through Lua code.

How many T_EX engines are available?

There are multiple T_EX engines:

- ▶ **Knuth's T_EX:**

This is original T_EX engine which serves as the lowest layer of L^AT_EX's software architecture.

- ▶ **pdf_tex:**

This engine adds a bunch of primitives related to the PDF and DVI extension.

- ▶ **xet_ex:**

This engine provides better font support.

- ▶ **lua_tex:**

Originally, meant to replace pdf_tex, but, now moving in a very different direction. This engine also better font support (like, xel_atex) through Lua code.

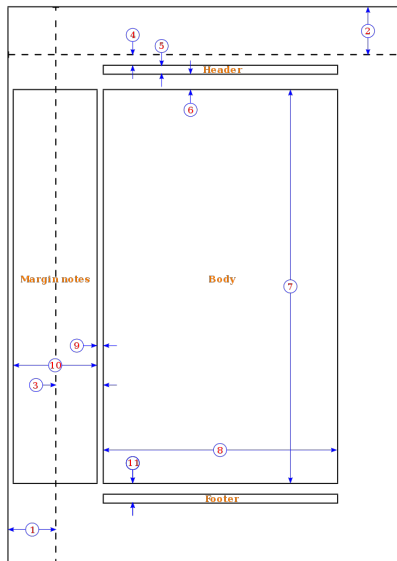
- ▶ and, a few more...

Structure of \LaTeX source file.

```
1  \documentclass{article}
2
3  \usepackage{hyperref}
4
5  \title{A critical analysis of Naruto: the Manga}
6  \author{Tejas Sanap}
7
8  \begin{document}
9      \maketitle
10
11 \end{document}
```

Page Geometry

- 4. `\topmargin`
- 7. `\textheight`
- 8. `\textwidth`
- 11. `\footskip`



Environments

- ▶ $\text{T}_{\text{E}}\text{X}$ give us the ability to directly use commands.
- ▶ \LaTeX builds up on that and provides us with *environments*.
- ▶ Environments perform an action on a block rather than just doing something in one place.
- ▶ Environments puts its contents inside a $\text{T}_{\text{E}}\text{X}$ group.
- ▶ And, prevents commands from leaking out.
- ▶ Environments restrict their effects to their own contents.

Environments

Example:

```
1 \begin{center}
2     The Jinchuriki.
3 \end{center}
```

Environments

Example:

```
1 \begin{center}
2     The Jinchuriki.
3 \end{center}
```

A few of the most common environments we see are:

- ▶ document.
- ▶ figure.
- ▶ align.
- ▶ table.