

L^AT_EX 101

L^AT_EX for absolute beginners

Ankur Sinha (FranciscoD) @ Fedora

January 10, 2019

Setup

Before we begin

- Please turn off your web cam, and mute your microphone.
- Please install the required packages:

```
sudo dnf install texlive-latex texlive-bibtex git
```

L^AT_EX?

What is \LaTeX ?

\TeX (pronounced *tech*)¹:

- a **typesetting** program:
 - typesetting: formatting documents (articles, presentations, posters, pamphlets ...).
 - focusses on appearance, rather than structure/content.

¹ Pronunciation of \TeX .

² Levels of \TeX .

What is \LaTeX ?

\TeX (pronounced *tech*)¹:

- a **typesetting** program:
 - typesetting: formatting documents (articles, presentations, posters, pamphlets ...).
 - focusses on appearance, rather than structure/content.

\LaTeX (pronounced *la-tech*)²:

- set of **macros** (commands) built over \TeX :
 - makes it easier to use, to quickly carry out common tasks.
 - lets the author focus on content.

¹ Pronunciation of \TeX .

² Levels of \TeX .

Why should one use \LaTeX ?

\LaTeX :

- is **Free software**: it is free to use, share, modify, and study:
 - enables **Open Science**.

³This presentation is written in \LaTeX

Why should one use \LaTeX ?

\LaTeX :

- is **Free software**: it is free to use, share, modify, and study:
 - enables **Open Science**.
- is available on Linux, MacOSX, and Windows.

³This presentation is written in \LaTeX

Why should one use \LaTeX ?

\LaTeX :

- is **Free software**: it is free to use, share, modify, and study:
 - enables **Open Science**.
- is available on Linux, MacOSX, and Windows.
- is easy to use (as you will see).

³This presentation is written in \LaTeX

Why should one use \LaTeX ?

\LaTeX :

- is **Free software**: it is free to use, share, modify, and study:
 - enables **Open Science**.
- is available on Linux, MacOSX, and Windows.
- is easy to use (as you will see).
- has minimum requirements:
 - is written in **plaintext**: so **any** editor will do, and **version control** can be used (git!),
 - required \LaTeX packages.

³This presentation is written in \LaTeX

Why should one use \LaTeX ?

\LaTeX :

- is **Free software**: it is free to use, share, modify, and study:
 - enables **Open Science**.
- is available on Linux, MacOSX, and Windows.
- is easy to use (as you will see).
- has minimum requirements:
 - is written in **plaintext**: so **any** editor will do, and **version control** can be used (git!),
 - required \LaTeX packages.
- generates easy to read, clean documents³.

³This presentation is written in \LaTeX

Using \LaTeX

Requirements

- a text editor, **any** text editor⁴:
vim/emacs/gedit/atom/geany/....
- \LaTeX packages.

⁴Microsoft Word is not an editor. It is a text processor—it does not generate plaintext as output.

Requirements

- a text editor, **any** text editor⁴:
vim/emacs/gedit/atom/geany/....
- \LaTeX packages.
- IDEs speed up writing:
 - **Overleaf** (web based).
 - **Texmaker**.
 - **LyX**.
 - Vim/Emacs/Atom/... + plug-ins.

⁴Microsoft Word is not an editor. It is a text processor—it does not generate plaintext as output.

Document structure

```
% Comments start with a %  
% Comment blocks are not supported!
```

```
\documentclass{...}
```

```
% This region is called the preamble.  
% Commands that setup the whole document go here.  
% Such as inclusion of packages.
```

```
\begin{document}
```

```
% The text of the document goes here.
```

```
\end{document}
```

```
% Anything here will not be processed.
```

Document classes

- standard layout for \LaTeX to use for the whole document.
- classes are provided as `.cls` files.

Document classes

- standard layout for \LaTeX to use for the whole document.
- classes are provided as .cls files.
- standard classes:
 - article.
 - report.
 - letter.
 - ...
- packages provide more:
 - beamer (used to make this presentation).

Document classes

- standard layout for \LaTeX to use for the whole document.
- classes are provided as .cls files.
- standard classes:
 - article.
 - report.
 - letter.
 - ...
- packages provide more:
 - beamer (used to make this presentation).

% Let us write an article

```
\documentclass[a4paper]{article}
```

The preamble

- Commands for the whole document: commonly: inclusion of packages.
 - Packages provide extra functionality⁵.

```
% Use Opensans fonts
\usepackage[default,osfigures,scale=0.95]{opensans}
% Use the hyperref packages for better links
\usepackage{hyperref}
% A title?
\title{An example \LaTeX{} document}
% The author?
\author{A. Human}
```

⁵Search CTAN for packages, and documentation

The text!

```
% Let LaTeX format the title
\maketitle
% Some text?
\LaTeX{} is easy!
```

The source file so far

```
% Filename: example-doc.tex
\documentclass[a4paper]{article}
\usepackage[default,osfigures,scale=0.95]{opensans}
\usepackage{hyperref}

\title{An example \LaTeX{} document}
\author{A. Human}

\begin{document}
\maketitle

\LaTeX{} is easy!

\end{document}
```

Generating the document: I

```
$ ls
example-doc.tex
$ latex example-doc
...
$ ls
example-doc.aux  example-doc.dvi  example-doc.log
example-doc.out  example-doc.tex
```

Generating the document: I

```
$ ls
example-doc.tex
$ latex example-doc
...
$ ls
example-doc.aux  example-doc.dvi  example-doc.log
example-doc.out  example-doc.tex
```

- No PDF file?

Generating the document: II

- The default output from \LaTeX is **DVI** (Device independent file format).

Generating the document: II

- The default output from \LaTeX is **DVI** (Device independent file format).
- But, PDF is most common now. So:

```
$ pdflatex example-doc
```

```
...
```

```
$ ls
```

```
example-doc.aux  example-doc.dvi  example-doc.log
```

```
example-doc.out  example-doc.pdf  example-doc.tex
```

- View it with your favourite PDF viewer (Evince/Okular/Adobe/Zathura/...)

Text structures: sections, lists

```
\section{Our first section}
```

This is a new section.

```
\subsection{Lists}
```

```
\begin{itemize}
```

```
  \item An itemised list!
```

```
\end{itemize}
```

```
\begin{enumerate}
```

```
  \item A numbered list
```

```
\end{enumerate}
```

- Save, re-run pdf_latex, view.

Text structures: mathematics

```
\begin{equation}
```

$$h^2 = b^2 + p^2$$

```
\end{equation}
```

where $\backslash(h\backslash)$, $\backslash(b\backslash)$, and $\backslash(p\backslash)$ are the lengths of the hypotenuse, the base, and the perpendicular of a right angled triangle respectively.

- Save, re-run pdflatex, view.

More text structures

- tables,
- figures, subfigures,
- headers, footers,
- source code snippets,
- algorithms,
- theorems,
- timelines,
- margin comments,
- indexes, glossaries,
- ...

Citations and referencing

Citations and referencing with BibTeX

- Most standard publishers provide BibTeX citation information for their articles (so does [Google Scholar](#)).

Citations and referencing with BibTeX

- Most standard publishers provide BibTeX citation information for their articles (so does [Google Scholar](#)).
- A plethora of Free/Open source bibliography managers are available, almost all of which, support BibTeX:
 - [JabRef](#).
 - [Wikindx](#) (Web based).
 - [Zotero](#) (Local and Web interfaces).

Citations and referencing with BibTeX II

- An example BibTeX entry:

```
% Save as mybib.bib in the same directory  
% as the document  
@Misc{RedHat2008,  
  author = {RedHat},  
  title  = {Fedora Project},  
  date   = {2008},  
  year   = {2008},  
}
```


Citations and referencing with BibTeX III

% Cite a reference in the text:

```
\section{A reference}
```

The Fedora project community\cite{RedHat2008} is committed to promoting the Free/Open source philosophy.

% A list of citations

```
\bibliographystyle{plain}
```

```
\bibliography{mybib}
```

- Save, re-run pdf_latex, view.

```
$ pdflatex example-doc
```

```
LaTeX Warning: There were undefined references.
```

2 A reference

The Fedora project[?] is committed to promoting Free/Open source.

Citations and referencing with BibTeX V

- Multiple passes are needed to generate the document.
Simply:
 - the locations of the citations are stored in the first pass,
 - the bibliography is processed next,
 - the locations are completed with the required text.

```
$ pdflatex example-doc && bibtex example-doc \N  
    && pdflatex example-doc && pdflatex example-doc
```

Citations and referencing with BibTeX V

- Multiple passes are needed to generate the document.
Simply:
 - the locations of the citations are stored in the first pass,
 - the bibliography is processed next,
 - the locations are completed with the required text.

```
$ pdflatex example-doc && bibtex example-doc \N  
&& pdflatex example-doc && pdflatex example-doc
```

- Most IDEs permit setting up the complete chain of necessary commands.
- Terminal users, try [Latexmk](#) (Yes, it's available in Fedora).

Collaborative writing

Using Git/Github

- Plaintext: multiple people can work on different parts of the text together.
- Use the power of **version control** (instead of **this!**)
- Can follow the standard **pull/merge** model.

⁶Include vs input

Using Git/Github

- Plaintext: multiple people can work on different parts of the text together.
- Use the power of **version control** (instead of **this!**)
- Can follow the standard **pull/merge** model.
- A few tips:
 - Write **each sentence on a new line**: this helps git to merge easily, since git looks for differences between lines.
 - Break the main text into **smaller files** using `include` or `input` commands⁶.

⁶Include vs input

Using Git/Github II

```
\begin{document}
% Let LaTeX format the title
\maketitle
% The text of the document goes here.

% sections in different files
\input{section1}
\input{section2}
\input{section3}

\bibliographystyle{plain}
\bibliography{mybib}

\end{document}
```


Epilogue

Common issues

- It isn't WYSIWYG (what you see is what you get, like Word)
 - That is the whole point: **stop thinking about how it'll look.**
 \LaTeX will take care of that. Focus on writing the text!

Common issues

- It isn't WYSIWYG (what you see is what you get, like Word)
 - That is the whole point: **stop thinking about how it'll look.**
 \LaTeX will take care of that. Focus on writing the text!
- Command writing is tedious!
 - Use an IDE that provides **completion**. If you write code, the same IDE that you use for programming will most probably also work for writing \LaTeX .

Common issues

- It isn't WYSIWYG (what you see is what you get, like Word)
 - That is the whole point: **stop thinking about how it'll look**.
 \LaTeX will take care of that. Focus on writing the text!
- Command writing is tedious!
 - Use an IDE that provides **completion**. If you write code, the same IDE that you use for programming will most probably also work for writing \LaTeX .
- I don't know Git:
 - Use **Overleaf**: it provides a web interface where you don't need to use Git explicitly. (but learn Git if possible)

Common issues

- It isn't WYSIWYG (what you see is what you get, like Word)
 - That is the whole point: **stop thinking about how it'll look**.
 \LaTeX will take care of that. Focus on writing the text!
- Command writing is tedious!
 - Use an IDE that provides **completion**. If you write code, the same IDE that you use for programming will most probably also work for writing \LaTeX .
- I don't know Git:
 - Use **Overleaf**: it provides a web interface where you don't need to use Git explicitly. (but learn Git if possible)
- Error messages are **hard to understand**:
 - Remove bits of the text to isolate what is causing the error.
If that doesn't work, **ask for help**.

fedoraproject.org/wiki/Classroom

docs.fedoraproject.org/en-US/neurofedora/latex/

Creative Commons Attribution-ShareAlike 4.0 International
License.

