Getting Started with LaTeX

Steven Au

1 Introduction

LaTeX is a typesetting system used to stylized and format documents. It is commonly used to create academic papers. It is pronounced as lay-tech and not made from the rubber tree. LaTeX is a markup language similar to HTML/CSS, a programming language for designing websites, where you specify text with commands that are compiled into a document. You can think as this as the magic that happens behind a word processor. Instead of clicking on a GUI (Graphical User Interface) to make a text bold, you can type the command `\textbf{text}` to produce **text**.

This is will be an entry level tutorial to familiarize yourself with LaTeX. I will be walking you through how to start your first latex document and common commands needed to make a basic document.

You might of heard that saying, if everyone is jumping off a cliff, would you? LaTeX is the same way. A lot of people use it in the academic space and there's no reason to use it over any other text editor. Consider it as plus to be proficient with LaTeX and judge if it is suitable for you. The main benefits are typing efficiency, more freedom, and no format issues. No need to move your mouse around, add intricate styling and hassle free formatting. Everything is technically in text, there won't be any weird version issues with other formats.

2 Getting Started

For this tutorial, we will being using overleaf.com to edit and compile our LaTeX. I will be mostly summarizing their resources for learning LaTeX. When you make a new blank project your file will generate a basic document. It will look something like this.

```
\documentclass{article}
\usepackage[utf8]{inputenc}

\title{Test}
\author{sttau}
\date{October 2022}

\begin{document}

\maketitle

\section{Introduction}

\end{document}
```

2.1 Preamble

The preamble is things that go before the document. They will typically be lines of code to help set up our document. Think of it as housekeeping, we need to tell the compiler what type of documents we are using and what is needed for them. The first line tells our compiler what type of document we are going to use. There are other predefined document templates we can use.

The command `\usepackage` imports a package for us to use and in this case we are importing the encoding for Unicode UTF-8. This package is necessary at this translate symbols into binary for the computer to compile. There a lot of packages you can use to help stylize

your latex document [1].

2.2 Special Characters

First off we see the usage of special characters such as \ (backslash) and { } (curly brackets). These are reserved characters that change our output. The backslash are for commands and the curly brackets are the parameters. The `\\` is used to end a table row and can be used as a forced line break [2] You can use `\hfil\break` for a similar effect. See **Paragraphs** for more ways to end a paragraph. . The % makes everything after it a comment, text the compiler ignores. Since the compiler sees the special character as a command there are special commands just to print these special characters stand alone (depicted in the chart below).

Character	Name	Print Command
\	Backslash	`\textbackslash`
~	Tilde	`\textasciitilde`
\\	Double Backslash	do backslash twice
%	Percent Sign	`\%`

2.3 Header

Simple enough the `\title`, `\author`, and `\date` do what they are respectively named. You can change the parameters as you see fit. If you are not using an article document you can use `\maketitle` to generate a title on a new page.

```
\title{PPT}
\author{UCSC}
\thanks{John Cena}
\date{October 2022}
\maketitle
```

1. Check out *https://www.ctan.org/* for an archive of TeX packages

2. Do **NOT** use it as a line break as this overrides formatting for certain You will get a warning of "Underfull hbox (badness 10000)"[1].

2.4 Environments

Environments are used to apply specific typesetting effects to a section of your document's content. The `\begin{document}` will contain all our elements till we end the document at `\end{document}`. This will tell the compiler to stop reading any more and is the last line of our .tex file. If you try to add commands outside of the document wrapper, Overleaf will give you a warning.

3 Adding Elements

3.1 Sections

Overleaf will create a file outline for you to easily navigate our file by tracking all sections commands. Use the `\section{}` and name the parameter whatever you want to call the section. You can section the document into smaller header by adding the prefix "sub." There is additional options to divide your document, but those are mostly used in a book format.

depth	command
-1	`\part{part}`
0	`\chapter{chapter}`
1	`\section{section}`
2	`\\subsection{subsection}`
3	`\subsubsection{subsubsection}`
4	`\paragraph{paragraph}`
5	`\subparagraph{subparagraph}`

3.2 Body Text

This section we will be looking at how to align text and space paragraphs. You can start writing text and the compiler will simply output it as is. You can also wrap the text with center, flush left, and flush right.

- `\begin{center}...\end{center}`
- `\begin{flushleft}... \endflushleft}`
- `\begin{flush right}... \end{flushright}`

Alignment	Example
center	foo
flush right	foo
flush left	foo

LaTeX text is justified as default, and using any other alignment will override it. Justified means the spacing is adjusted so the text spans over the width of the document. You can use these switch commands to alter any text as a global rule or encapsulate it in an environment to be local. `\RaggedRight` will align left and similar to flushleft environment, but has no justification adjustment. `\RaggedLeft` will do the opposite and align right and `\centering` will center.

The command `\par` will start a new paragraph with an indent. You can also use `\break` to get a similar effect but add spacing after the paragraph. The quick commands are `\smallbreak`, `\medbreak`, and `\bigbreak`. If you want to skip a line at the end of line use the skip suffix such as `\smallskip`, `\medskip`, and `\bigskip`.

3.3 Tables

Tables are the most annoying thing you will encounter in LaTeX. Its hard to see what you are editing in LaTeX opposed to a word processor where you can click and edit any property of the table.

There are two environments, **tabular** is the actual table itself with columns, rows, and cells. While **table** allows us to abstract it for locating purpose and appends captions and labels.

First we need to define the amount of columns when calling the tabular environment. `\being{tabular} { X X X }` We can replace X with either l, c, or r to represent column aligning. L is left align, c is center and r is right align. Here we have 3 columns that are assigned. We can insert a vertical bar(|) between any of our columns to add a vertical line.

To add the cells we need to define its contents and separate each cell by an ampersand (`&`) and end the row with two backslashes (`\\`). Adding `\hline` between rows give a horizontal line.

```
\begin{tabular}{|c|c|}
\hline
One & Two\\
\hline
Three & Four\\
\hline
\caption{Example of a Table}
\end{tabular}
```

One	Two
Three	Four

TABLE 1: Example of a Table

3.4 Figures

Just like how you can label tables, you can also label images, and charts. This allows us to easily reference figures later on in the document. This is useful in longer papers where tons of figure can be confusing to manage. Having a good name for the label will help in identifying them later on. I usually like to name it by its type, function, and description.

The parameter "h" on figure means that LaTeX tries to put the image where it is mentioned in the code. Annoyingly it tries to move items around for spacing issues. If you tried complying your document and the spacing looks awkward, you might want to restructure your document.

```
\begin{figure}[H]
\centering
\includegraphics[width=1.5in]{slug.pdf}
\caption{Sammy the Slug}
\label{img_fig_slug}
\end{figure}
```

Figure Parameters

Parameter	Position
h	Tries to place the float where it is in code.
t	Position at the *top* of the page.
b	Position at the *bottom* of the page.
p	Put on a *special* page for floats only.
!	Overrides LaTeX's parameter for placement.
H	Place float at code location.—

These figure parameters can be applied to any floats. Basically any command the has

Fig. 1: Sammy the Slug, UCSC beloved mascot since 1986.

`\begin`. Its good to check the logs of the compiler to check for any issues with formatting. It explains if certain elements are too wide or had to be adjusted to compile.

Below is an example of the capabilities of graphing a data in LATEX with the pfgplots package.

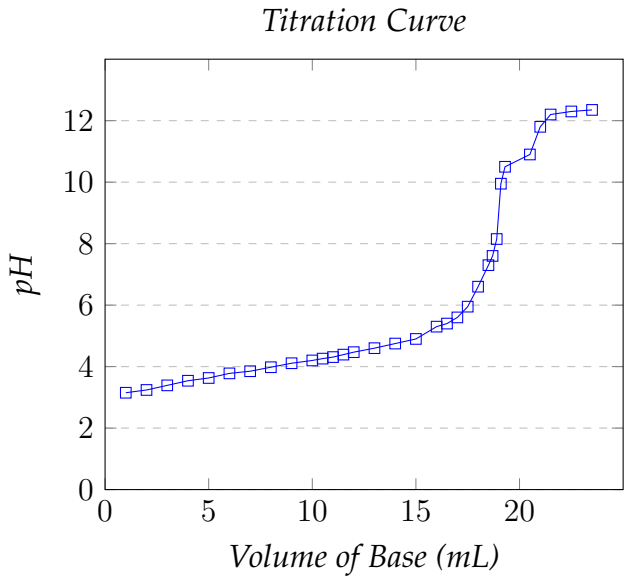

Fig. 2: The pH is greater than 7 at the equivalence point so this is a weak acid-strong base titration

3.5 Mathematical Formulas

The typesetting mathematics in LATEXis superb to any other document editor I seen. There is not much you can get wrong as commands make sense when doing math. You type the commands in order of the operations.

3.5.1 Display Mode

There are two ways to display mathematical expression inline and display. You can use any delimiter to produce the same results

Inline Math Mode:

- `\(...\)`
- `$...$`
- `\begin{math}...\end{math}`

Inlines allows mathematical expression to be seamlessly incorporated in-text.

> In physics, the mass-energy equivalence is stated by the equation $E = mc^2$, discovered in 1905 by Albert Einstein.

Display Math Mode:

- `\[...\]`
- `\begin{displaymath}... \end{displaymath}`
- `\begin{equation} ...\end{equation}}`

Display allows the expression to standalone and centered. Using the equation numbers our equations.

> The mass-energy equivalence is described by the famous equation
>
> $$E = mc^2$$
>
> discovered in 1905 by Albert Einstein. In natural units (c = 1), the formula expresses the identity
>
> $$E = m \tag{1}$$

3.5.2 Common Symbols

There are tons of symbols that LATEXuses. It will take a while to remember them are, but here are some commonly used ones.

Greek letters:

α `\alpha` β `\beta` γ `\gamma` ρ `\rho` σ `\sigma` δ `\delta` ϵ `\epsilon`

Binary operators:

$\times$ `\times` $\otimes$ `\otimes` $\oplus$ `\oplus` $\cup$ `\cup` $\cap$ `\cap`

Relation operators:

$<$ `<` $>$ `>` $\subset$ `\subset` $\supset$ `\supset` $\subseteq$ `\subseteq` $\supseteq$ `\supseteq`

3.5.3 Operations and Scripts

Fractions are denoted by the command `\frac{}{}`. The first curly braces are for the numerator and the second one for the denominator.Just like in math you can nest operations within themselves with curly braces.

$$\frac{x^{2+a}}{\sqrt{x_{2+a}}}$$

4 Finishing Touches

4.1 How to: Acknowledgments

Acknowledgments are a great way to thank anyone who has helped you in anyway. However, you want to sound concise and not overly informal. Be sure to be descriptive if possible to prevent sounded redundant. Use the `\section*{Acknowledgements}` to create an unnumbered section. The point of this is to give credit to those who have helped you on your project or thesis paper. Remember to think about who your audience is when writing your acknowledgements. If you wrote a book, talk about your family and friends, and if it is a group research paper, talk about your colleagues and supervisor.

4.2 How to: References

At then end of the document it is important to cite all the sources of information you found. In the academic setting plagiarism is a sin and any information must be given some truth. Therefore references provide a way for your reader to find papers about any ideas mention in your own academic paper.

4.2.1 Bibliography

LaTeX uses the thebibliography environment to hold all the items it itemizes them by the command `\bibitem`.

References

[1] Donald E. Knuth (1986) *The TeX Book*, Addison-Wesley Professional.

```
\begin{thebibliography}{9}
\bibitem{texbook}
Donald E. Knuth (1986) \emph{The \TeX{} Book},
Addison-Wesley Professional.
\end{thebibliography}
```

This is a simple way of doing the reference, there is a better way incorporating bibTeX. A special way LaTeX handles bibliography entries.

4.2.2 bibTeX

Just like how we talked about labels being able to reference items, there is a specific reference for bibliographies called cites. We can call a cite key to refer back to our bibliographies that are formatted in bibTeX bibliography database files (.bib files).

```
  @book{texbook,
  author = {Donald E. Knuth},
  year = {1986},
  title = {The {\TeX} Book},
  publisher = {Addison-Wesley}
}
```

An example how they are stored in a .bib file [2].

```
\begin{thebibliography}{widest entry}
 \bibitem[label1]{cite_key1} biblio info
 \bibitem[label2]{cite_key2} biblio info
 ...
\end{thebibliography}
```

4.2.3 Cross-referencing

`\label{marker}`

Used to give the object you want to reference a marker — a name which can be used to refer to that object later.

`\ref{marker}`

Used to reference an object with the specified marker. This will print the number that was assigned to the object.

`\pageref{marker}`

Used to print the page number where the object with the specified marker is found [**?**].

`\cite{cite_key}` Used to cite a bibliography by its cite key that appears in the `\bibitem` or bibTeX[2].

5 CONCLUSION

If you made it this far, feel proud. LaTeX has a steep learning curve. It takes practice to get better at coding. Its OK to make mistakes so you can learn from them. I personally avoided learning LaTeX since it is so time consuming. I learned to appreciate it for what it does best. If you want professionalism in you document LaTeX is the way to go. I have previously used other mediums like Adobe InDesign and Microsoft Word, and I like how robust LaTeX is. You can easily reference figures and citations then any other platform. I hope you gained something out of this tutorial.

ACKNOWLEDGEMENTS

I would like to thank John Hammersley for founding and creating Overleaf. His platform streamlines and simplifies the process to write detailed documents. I used a lot of their resources to learn LaTeX. I also thank Donald Knuth the creator of TeX which LaTeX was based on. He was a Stanford University professor that wanted to improve the quality of mathematical notation in his books.

REFERENCES

[1] Andrew Swann "When to use `\par` and when `\\`, `\newline`, or blank lines" tex.stachexchange.com https://tex.stackexchange.com/questions/82664/when-to-use-par-and-when-newline-or-blank-lines (accessed Oct. 20, 2022)

[2] Illinois Tech, "Creating Bibliography with LaTeX", iit.edu https://webmaster.iit.edu/files/graduate-academic-affairs/latex-bibliography-help.pdf (accessed Oct. 20, 2022)

[3] Wikibooks "LaTeX/Labels and Cross-referencing iit.edu en.wikibooks.org/ https://en.wikibooks.org/wiki/LaTeX/Labels_and_Cross-referencing (accessed Oct. 20, 2022)