An Introduction to Writing Documents in LATEX

Alison Sun

So , why learn to use LATEX? Unlike Python, C/C++, and other programming languages you may have worked with, LATEX is a document markup language used to write documents in TeX, a *typesetting* system — instead of coding applications, you use markup tagging conventions to format and stylize a text document, usually for scientific papers. You'll need to write assignments in LATEX for some of your upper-division classes, and it's a very valuable tool for writing research articles if you later go on to pursue a career in academia.

However, regardless of whether you go on to use it in your career, LATEX is a super useful tool because it has the most built-in functionalities for writing symbols and formulas in your documents. Can you imagine having to type out integrals in Google Docs every time?

1 Creating Your .tex File

The way writing files using LATEX works is that you'll write your content in a .tex file, and your LATEX editing software will work its magic and compile it into a PDF document with everything formatted correctly. There are a number of different LATEX editors out there — I prefer to use Overleaf, which is online, but some other good ones include Texmaker and TeXstudio, which are programs that you download to your computer. For the purposes of this tutorial, we'll be using Overleaf.

It's not uncommon for universities to provide free Overleaf licenses to students, so definitely check if that resource is available to you!

1.1 Preamble

Before you actually start writing your LaTeX document, there's some important setup information to include at the top — this is called the preamble, and it includes your document class, packages, and document environment.

1.1.1 documentclass and Class Files

The very first thing in your LaTeX document, before any content, before any headers, before even the title and the other setup commands, is the `\documentclass{}` command. Every .tex file should include it — in fact, it's so important that Overleaf automatically includes it at the top of your file whenever you create a new project.

The document class specifies the standard layout of your document via a class file — commonly used document classes include `article`, `book`, and `letter`. Each document class has its own specifications for font size, paper size, page layout, and more. If your document requires making alterations to the default settings, you can adjust them via optional parameters, which would look something like this:

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

Many academic journal publishers also build their own document classes to standardize their article submissions. For instance, this tutorial uses the `IEEEtran`, or IEEE Transactions, document class, which is the document class used for publications and conferences for the IEEE Translations journal from the Institute of Electrical and Electronics Engineers (IEEE).

1.1.2 Packages

More often than not, you'll end up also including packages in your LaTeXdocuments. Packages are collections of additional commands and features that are imported into documents to add extra functionalities, including everything from adding images or math symbols to printing a large coffee stain over your text. This document utilizes the package `graphicx`, which allows you to insert images. You'll define packages between the `\documentclass{}` and `\begin{document}` commands.

1.1.3 Environments

Some of the most important commands in LaTeX are the `\begin{}` and `\end{}` commands, which mark the beginnings and ends of environments. This includes, most importantly, your whole document: `\begin{document}` and `\end{document}` mark the beginning and end of the content in LaTeX documents. Notice that when initializing an environment, you'll specify its name inside the curly braces.

Documents aren't the only notable LaTeX environments, though: some other environments that we'll go over in this tutorial include `\figure`, `\table`, and `\thebibliography`.

1.1.4 User-Defined Environments

In addition to LaTeX's predefined environments, you can also opt to create your own environments with the command

```
\newenvironment{name}[numarg]
[optarg_default]{begin_def}{end_def}
```

where 'name' is the name of your new environment, 'numarg' specifies the number of arguments to pass into the curly braces, 'optarg_default' specifies an optional default argument, and 'begin_def' and 'end_def' include code to begin and end the environment with the `\begin{}` and `\end{}` commands.

2 The Actual Content

Congratulations, you've made it through the document setup and preamble information! Now it's time to dive into the body text of your document. We'll be covering things like how to format your text, how to insert figures and tables, and more.

2.1 Titles and Headings

Titles are super important for any kind of document. You can add titles with the command `\title{YOUR TITLE HERE}`, your name with `\author{YOUR NAME HERE}`, and the date with `\date{ADD THE DATE HERE}`. A handy little date macro in LaTeX is `\date{\today}`, which autofills the current date. Here's how I would set the title information for a typical LaTeX document:

```
\title{The Coolest Document Ever}
\author{Alison Sun}
\date{\today}
```

2.1.1 Maketitle

Another super important command in a LaTeX document is `\maketitle`, which properly renders and formats not only your title, but also other crucial information like author name and article date. Without it, all of your title information simply disappears, so make sure to include it right after your title!

2.2 Sections

Let's be honest — nobody wants to read a paper that's just a gigantic wall of text. Dividing your document into cohesive sections, subsections, and even subsubsections helps break things up and add structure. To indicate a subsection within your section, simply use `\subsection{Subsection Title Here}`.

2.2.1 Subsubsections

To make a *sub*-subsection (like this one!), all you need to do is throw another 'sub' in front of the subsection command: `\subsubsection{Subsubsection Title Here}`.

2.3 Body Text

Now, you can really go ham on writing your content. As you do so, here are a few more things to keep in mind.

2.3.1 The Verbatim Environment

While writing this guide, I often had to display lines of LaTeX code for instructional purposes without having the compiler run them as actual code. Here's where the verbatim environment comes in handy. To write verbatim LaTeX inline (in line with the text), use `\verb`, with your choice of delimiters between +, ^, _, —, and # to surround the text you want to display. You can also use verbatim in a display style (set apart from the text), by calling the environment: `\begin{verbatim}`.

2.3.2 Reserved Characters

Some characters can't be encoded when included in the document on their own, because they have special purposes reserved for LaTeX commands. These characters, also known as "reserved characters," are "\", "\\", "~", and "%". Single backslashes placed before text indicate commands, while double backslashes insert line breaks. Tildes insert spaces, and percentage signs indicate comments in LaTeX. If you need to print a reserved character, simply type a \ before it.

2.4 Figures

You can use the `figure` environment to add images and graphics to your document. Here, it's particularly important to have the `graphicx` package installed, as it supports the `\includegraphics{}` command, which allows you to insert files in common formats like jpg, pdf, png, and more. The `figure` environment also allows you to set the placement (`[h]`), center the graphic, specify the width, and add a caption and a label.

For example, the following code inserts a fun little picture of Sammy the Slug in your LaTeXdocument:

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

Fig. 1. Sammy the Slug

In a similar vein, the `figure` environment can also help us add graphs into our LaTeXdocuments:

```
\begin{figure}[h]
\centering
\includegraphics[width=2.25in]
        {Titration_Plot.png}
\caption{This graph plots titration
data in terms of the volume of the
base vs. its pH. Titration is a
chemical process in which a
solution of a known concentration
is used to determine the
concentration of another solution.
Here, the known solution is NaOH
(sodium hydroxide), while the
unknown solution is a mystery acid.}
\label{Titration}
\end{figure}
```

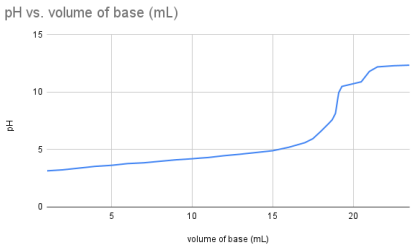

Fig. 2. This graph plots titration data in terms of the volume of the base vs. its pH. Titration is a chemical process in which a solution of a known concentration is used to determine the concentration of another solution. Here, the known solution is NaOH (sodium hydroxide), while the unknown solution is a mystery acid.

2.5 Tables

Tables are a common way to express data across any discipline. Expressing them in LaTeXis pretty simple — you have to call the `table` environment, which allows you to specify things like centering, a label, and table style.

Then, within the `table` environment, the `tabular` environment is also required to actually populate the values into your table. Here, you would set the number of columns with either l, c, or r to denote left, center, or right alignment, as well as `|` symbols as vertical separators. When filling in values, keep in mind that & represents shifting to the next column, while `\\` moves you to the next row. New horizontal lines can be inserted by typing `\hline`.

```
\begin{table}[h]
\renewcommand{\arraystretch}{1.3}
\caption{An Example of a Table}
\label{table_example}
\centering
\begin{tabular}{|c||c||c|}
\hline
Alpacas & Bears & Cats\\
\hline
Dogs & Elephants & Frogs \\
\hline
Giraffes & Hippos & Iguanas \\
\hline
\end{tabular}
\end{table}
```

TABLE 1
An Example of a Table

Alpacas	Bears	Cats
Dogs	Elephants	Frogs
Giraffes	Hippos	Iguanas

2.6 Mathematical Formulas

One of LaTeX's biggest strengths is its typesetting capabilities for mathematical characters and formulas, which is super important for creating scientific documents. These can be displayed in one of two modes: `inline` with the text: $x^n + y^n = z^n$, or `display`, set apart from the text:

$$x^n + y^n = z^n$$

Inline math can be denoted with any of the following:

- `\(...\)`
- `$...$`

Similarly, you can denote display math with

- `\[...\]`

Or, you can utilize the environment `math` for inline equations, and `displaymath` or `equation` for display equations.

Once you've mastered the basics of formatting equations and formulas, feel free to explore more advanced symbols and representations, like Greek letters (`\alpha` yields the symbol α), fractions (`$a=\frac{b + c}{d}$` gives you $a = \frac{b+c}{d}$), square roots (`$\sqrt{x + 1}$` for $\sqrt{x+1}$), and even integrals (`$\int{x^2}$` for $\int x^2$). Most of these symbols are inbuilt in LaTeX, but you can also look into installing packages to use the ones that aren't.

With that, let's put (almost) all of that together. Here's how to express the quadratic formula in LaTeX.

```
\begin{displaymath}
x = -b \pm \frac{\sqrt{b^2 - 4ac}}
    {2a}
\end{displaymath}
```

$$x = -b \pm \frac{\sqrt{b^2 - 4ac}}{2a}$$

3 THE STUFF AT THE END

You've made it to the end of your main content. We're in the home stretch! Now is the time to think about wrapping it up by writing things like appendices, acknowledgements, and your bibliography.

3.1 Acknowledgements

Here's where you thank all the people who helped make your paper possible, or supported you during the process — your advisor, your instructor, your mom, your best friend, your dog... the list goes on. Nothing much to keep in mind here that we haven't discussed already, but a small formatting note: to get rid of the section number (i.e. 4.1) for your Acknowledgements section, simply add an asterisk between `section` and the first curly brace: `\section*{}`.

3.2 References

As with any piece of writing you produce, you'll need to reference content you used throughout the process. Here, we'll go over how to create bibliographies, and how to use the `ref`, `cite`, and `label` commands.

3.2.1 Bibliographies and Cite Commands

The environment `thebibliography` helps to denote your bibliography, which is made up of `bibitems` of each of your sources. You can refer back to each `bibitem` with the `\cite{reference}` command, where "reference" is the name of the source you're referencing. For instance, here I could cite the source I used to write this section: `\cite{Bibliography}`, and it shows up as the index number of the source in the bibliography: [5].

3.2.2 Label and Ref Commands

Just like `cite`, the `label`and `ref` commands are a great way to refer back to things throughout the text. You can assign labels to your figures (these are different from captions), and then utilize them to refer back to the figures as you write the rest of your content. For instance, here I can reference the `fig_slug` label that I gave the Sammy Slug image like so: `\ref{fig_slug}`, which appears as 1.

4 Conclusion

As we come to the end of this tutorial, you've most likely learned the basics of creating and organizing things like preambles, packages, environments with the help of LaTeX's powerful typesetting capabilities. Armed with this knowledge, hopefully you'll be inspired to embark on a journey of how to create beautifully formatted documents — from books to scientific papers to class reports and beyond. Happy typesetting!

Appendix A

P.S. Adding appendices is pretty similar to adding regular sections: just remember to add the command `\appendix` at the front to signal to the compiler that it's different from an ordinary section.

Acknowledgements

The author would like to thank...

Her partner, her best friends, her family, her CSE 185 TA, and the friendly neighborhood cat.

References

[1] "Document Structure." *Wikibooks*. Accessed on February 2, 2024. https://en.wikibooks.org/wiki/LaTeX/ Document_Structure.

[2] "Floats, Figures, and Captions." *Wikibooks*. Accessed on February 2, 2024. https://en.wikibooks.org/wiki/LaTeX/ Floats,_Figures_and_Captions.

[3] "Inline Verbatim." *LaTeX Stack Exchange*. Accessed on February 1, 2024. https://tex.stackexchange.com/ questions/387684/inline-verbatim.

[4] "Mathematical Expressions." *Overleaf*. Accessed on February 3, 2024. https://www.overleaf.com/learn/ latex/Mathematical_expressions.

[5] "Should I cite using `\begin{thebibliography}` or `\begin{filecontents*}`." *LaTeX Stack Exchange*. Accessed on February 3, 2024. https://tex.stackexchange.com/questions/482318/should-i-cite-using-beginthebibliography-or-beginfilecontents.