A Quick Introduction

Sean Allred

June 19, 2014

Abstract

This article aims to be a model LaTeX document while teaching you the basics of what it is and how to use it. It contains all of the basic constructs you are likely to encounter as you write your first papers and articles. This article will not go into detail about how to get started with a local installation of LaTeX.

Contents

1 Introduction

Your introduction goes here! Some examples of commonly used commands and features are listed below, to help you get started. If you have a question, please use the help menu ("?") on the top bar to search for help or ask us a question.

I've sent you this link because you've demonstrated frustration with Word or curiosity in other ways of doing things. If you want to see how something in the output was done, click on it and writeLaTeX.com will highlight the appropriate line. You can clone this example to your own, editable copy using the writeLaTeX.com interface, and I encourage you to do so. The best way to learn is to do, and usually to mess things up. In the immortal words of Mrs. Frizzle:

> Take chances! Make mistakes! Get *messy*!

2 What is LaTeX?

LaTeX is, in the words of its creator, a 'document preparation system'. It takes the raw text that you see here on the left and it produces the formatted document that you see on the right. It's important to note that LaTeX is *not* writeLaTeX.com; rather, writeLaTeX.com is *powered* by the underlying LaTeX program. Every time you change the plain text, this is run through the LaTeX program and the server displays the image for you to see. It is not just another format available for you to write your papers in (like using OpenOffice instead of Word), but it's rather a different way of thinking about what you as an author should be doing with your document.

Let's face it: you are neither a typographer nor a publisher, but you're at least pretty OK at writing. LaTeX knows this and knows that it's far more important for you to understand *what you mean* in your document. If you know what you *mean* and can tell LaTeX how to demonstrate that meaning on the page, then you have a very consistent and easy-to-use *document preparation system.* It is not just a format; it is a system. Use the system.

LaTeX comes with a lot of this 'meaning' markup baked-in.[1] For instance, I can emphasize a piece of text by surrounding it with the `\emph{stuff}` markup; LaTeX will do the 'right' thing. For instance, if I want to emphasize part of an emphasize *it will actually* switch back *to demonstrate that* additional *emphasis.* There are many commands that do multiple things all at once: if I wanted to start a new section—like those you see in this article—all you need to do is say `\section{title}` and LaTeX will do the 'right' thing(s):

- It will increment the section counter.
- It will set the title as it understands how sections are set.
- It will insert the title into the table of contents.

This part of LaTeX is about what we call 'typographical consistency'. I can change the behavior of *all* sections with a few short lines. If you're reading along in the 'source document' (the plain text to the left), I can change how 'terms' are set everywhere by changing exactly one line. The possibilities really do not stop there, but in the interest of keeping this article relatively short, I will leave that research to the reader.

3 Some LaTeX Examples

Now that you know the idea behind LaTeX, let's get into some concrete examples. If you haven't read the introduction (section 2), I strongly urge you to; things from here on out will make more sense. I encourage you to read along with the source; if you have been you already know how to emphasize text with `\emph`, start a bulleted list with `{itemize}`, start a section with `\section`, give a footnote with `\footnote`, and use a cross-reference using `\label` and `\autoref`.[2] You also know that LaTeX doesn't really pay attention to inconsistent

[1] This 'meaning' markup is often called 'semantic' markup.

[2] Actually, this command is given by the `hyperref` package. The standard command to use is `\ref`, but `\autoref` will insert the appropriate label in front (like 'section') when `hyperref` knows it. Of course, you can teach the package new things.

inter-word spacing or random line breaks. The only real 'rule': if it encounters a *blank* line, it starts a new paragraph.

3.1 How to Organize Your Document

As you've already seen, LaTeX comes with at least one command to organize your document: `\section`. There are, in fact, many others:

1. `\part`
2. `\chapter`
3. `\section`
4. `\subsection`
5. `\subsubsection`
6. `\paragraph`
7. `\subparagraph`

Actually, the availability of these commands[3] depends on the 'document class' you use. For example, `\documentclass{article}` doesn't define `\part` or `\chapter`, but `\documentclass{book}` does.

3.2 How to Make Lists

You can make lists with automatic numbering using `{enumerate}`...

1. Like this,
2. and like this.

...or bullet points using `{itemize}`...

- Like this,
- and like this.

...or with words and descriptions using `{description}`...

Word Definition

Concept Explanation

Idea Text

3.3 How to Include Figures

See the code for Figure 1 in this section for an example.

1. Upload the image file (JPEG, PNG or PDF) from your computer to writeLaTeX.com using the upload link the project menu.
2. Use the `\includegraphics` command to include it in your document.
3. Use the `{figure}` environment and the `\caption` command to add a number and a caption to your figure.

[3] Or 'control sequences' as they are more properly called.

Figure 1: This frog was uploaded to writeLaTeX.com via the project menu.

Item	Quantity
Widgets	42
Gadgets	13

Table 1: An example table.

3.4 How to Make Tables

Use the `{table}` and `{tabular}` commands for basic tables—see Table 1 on page 4, for example. `\toprule`, `\midrule`, and `\bottomrule` are all provided by `booktabs`. The standard command to use is `\hline`, but see the `booktabs` documentation[4] for some nice reading on why the tables it suggests are better.

3.5 How to Write Mathematics

LaTeX is great at typesetting mathematics. Let $X_1, X_2, \ldots, X_n$ be a sequence of independent and identically distributed random variables with $\mathrm{E}[X_i] = \mu$ and $\mathrm{Var}[X_i] = \sigma^2 < \infty$, and let

$$S_n = \frac{X_1 + X_2 + \cdots + X_n}{n} = \frac{1}{n}\sum_{i}^{n} X_i$$

denote their mean. Then as n approaches infinity, the random variables $\sqrt{n}(S_n - \mu)$ converge in distribution to a normal $\mathcal{N}(0, \sigma^2)$.

I have no idea what *any* of that means, so see Equation 1. If you want to be concise about it, you can also use `\eqref`: see (1).

Theorem 1. *If this, then that.*

Proof. Since we know that

$$\text{this} \implies \text{that}, \tag{1}$$

the result directly follows. □

There is a lot more to mathematics typesetting, but that's beyond the scope of this article. They may be another that looks at this in-depth.

[4] `http://texdoc.net/texmf-dist/doc/latex/booktabs/booktabs.pdf`

4 Getting Help

LaTeX comes with very thorough documentation. For online use, simply Google-search `texdoc (package)` to look up the official documentation for `package`. This mirrors the command-line tool, `texdoc`, which is used in the same way. For everything else, usually a simple search for `LaTeX (thing)` for information on `\thing` or `{thing}`. For everything else, including specific help, there's the TeX StackExchange.[5]

5 Next Steps

5.1 Getting LaTeX

writeLaTeX.com is a useful resource and I encourage you to use it to the fullest. However, LaTeX is a program that can run on *your* computer just as easily as it can run online. It is absolutely free. My recommendations are as follows:

Windows TeX Live[6] (but MiKTeX[7] is another popular option).

Mac MacTeX[8] is by *far* the best. I highly recommend it.

Linuxes TeX Live[9] is your *only* option, and I actually recommend against using your distribution's package manager to install it. The versions are almost always several years behind, and LaTeX package development is very active. If you must though, install `texlive-full` or similar—trust me, it makes life *much* easier.

To make links easier to work with, download the PDF using the appropriate button above on the top bar. This is what LaTeX actually produced for writeLaTeX.com, and is fully clickable and otherwise interactive. (It also just looks *nicer* somehow...)

5.2 Choosing an Editor

Your experience with LaTeX will largely be impacted by your choice of editor. For learning on any platform, TUG's own TeXworks is a good starting point. When you find you're repeating yourself a lot, I recommend moving onto TeXmaker (also cross-platform). If you are comfortable using it—I won't lie and say it's immediately intuitive for a beginner—the venerable Emacs editor has an excellent mode called AucTeX. It is *extremely* good. TeXmaker and Emacs will see you through from then on, and both are absolutely free.

Happy TeXing!

[5] `http://tex.stackexchange.com`

[6] `http://www.tug.org/texlive/windows.html`

[7] `http://www.miktex.org`

[8] `http://www.tug.org/mactex`

[9] `http://www.tug.org/texlive`