

# L<sup>A</sup>T<sub>E</sub>X Tutorial for Beginners

Shane Miller

July 16th, 2019

# Contents

<b>1</b>	<b>Installing <math>\text{\LaTeX}</math></b>	<b>1</b>
1.1	Windows Instructions . . . . .	1
1.2	Mac Instructions . . . . .	2
1.3	Linux Instructions . . . . .	2
<b>2</b>	<b><math>\text{\LaTeX}</math> Important Notes</b>	<b>3</b>
<b>3</b>	<b>General Formatting of a <math>\text{\LaTeX}</math> Document</b>	<b>3</b>
<b>4</b>	<b>Title Page</b>	<b>5</b>
<b>5</b>	<b>Table of Contents and Hyperlinking</b>	<b>6</b>
<b>6</b>	<b>Section Types: Main Section</b>	<b>6</b>
6.1	Section Types: Subsection . . . . .	7
6.1.1	Section Types: Subsubsection . . . . .	7
<b>7</b>	<b>Paragraphs</b>	<b>7</b>
<b>8</b>	<b>Text Formatting</b>	<b>8</b>
8.1	Font Size . . . . .	8
8.2	Font Style . . . . .	8
8.3	Verbatim . . . . .	8
<b>9</b>	<b>Lists</b>	<b>9</b>
9.1	Bulleted List . . . . .	9
9.2	Ordered List . . . . .	9
9.3	Combined Lists . . . . .	10
<b>10</b>	<b>Math</b>	<b>10</b>
<b>11</b>	<b>Other Resources</b>	<b>11</b>

**Important Note:** A few of the examples in this document are specific to the article documentclass. For example, in the article documentclass, sectioning starts at 1. In other things such as report, it starts at 0.1. Due to this, I would suggest starting your first project using the article document and then experimenting with other documentclasses once you are comfortable with L<sup>A</sup>T<sub>E</sub>X. When you eventually switch to trying out different documentclasses, don't take everything here as exact. More or fewer things may be available for that documentclass. For example, I give an example of three commands you can use, but in other documentclasses more or fewer commands may be available to use. \subtitle is available in the book documentclass, but not the article documentclass. In the intermediate tutorial I will go over the use of a different package that is commonly used instead of the default documentclass options L<sup>A</sup>T<sub>E</sub>X provides. Please take these things into consideration when using this document for reference.

# 1 Installing L<sup>A</sup>T<sub>E</sub>X

## 1.1 Windows Instructions

1. Go to [miktex.org](http://miktex.org).
2. Navigate to [Downloads](#).
3. Download the MiKTeX installer for Windows.
4. Run the MiKTeX installer once it finishes downloading.
5. Agree and continue.
6. Set preferred paper size and set “Install Missing Packages” to automatic.
7. When this is finished you will have the MiKTeX Console and the TeXworks editor. Make sure you have both.
8. Run the MiKTeX Console.
9. Select “Check for Updates” and allow it to install any it finds.
10. Once updates are finished installing, close the MiKTeX Console and open TeXworks.
11. Begin writing your first L<sup>A</sup>T<sub>E</sub>X document.
12. When saving, I recommend creating a LaTeX folder in which to store all of your L<sup>A</sup>T<sub>E</sub>X documents and then creating a subfolder for that specific document (eg: LaTeXDocuments → MyLaTeXProject → project files). When you save your .tex file, it will save a couple others there as well as the PDF when compiled.

## 1.2 Mac Instructions

1. Go to the MacTeX website [here](#).
2. Click [MacTeX Download](#).
3. Click on [MacTeX.pkg](#) to download MacTeX.
4. MacTeX.pkg will appear in your downloads folder once it is finished downloading. Double-click it to run the installer.
5. Follow the instructions in the installer to install MacTeX.
6. Once complete. navigate in finder to Applications.
7. In Applications, open the folder named “TeX.”
8. Inside it you will see an application named “TeX Shop.” Open it.
9. Begin writing your first  $\text{\LaTeX}$  document.
10. When saving, I recommend creating a LaTeX folder in which to store all of your  $\text{\LaTeX}$  documents and then creating a subfolder for that specific document (eg: LaTeXDocuments  $\rightarrow$  MyLaTeXProject  $\rightarrow$  project files). When you save your .tex file, it will save a couple others there as well as the PDF when compiled.

## 1.3 Linux Instructions

1. In the command line type: `sudo apt-get install texlive-full`.
2. In the command line type: `sudo apt-get install texmaker`.
3. In the command line type: `texmaker`.
4. The UI for TeXmaker should have booted. From there select: File  $\rightarrow$  New.
5. Begin writing your first  $\text{\LaTeX}$  document.
6. When saving, I recommend creating a LaTeX folder in which to store all of your  $\text{\LaTeX}$  documents and then creating a subfolder for that specific document (eg: LaTeXDocuments  $\rightarrow$  MyLaTeXProject  $\rightarrow$  project files). When you save your .tex file, it will save a couple others there as well as the PDF when compiled.

## 2 L<sup>A</sup>T<sub>E</sub>X Important Notes

- A L<sup>A</sup>T<sub>E</sub>X file might need to be compiled more than once.
  - This can happen for a couple of reasons, the main one being if something you use relies on something further down in the code.
  - The most common example of this is the title page. You will make sections after you use `\maketitle` to have L<sup>A</sup>T<sub>E</sub>X automatically make your title page. Because of this, your L<sup>A</sup>T<sub>E</sub>X will need to be compiled once to compile the rest of the document, and then one more time for your title page to add the sections to itself.
  - On Mac, if you add `% !TEX TS-program = pdflatexmk` to the beginning of your L<sup>A</sup>T<sub>E</sub>X document, that may compile it the correct number of times automatically if you have latexmk installed (which is often automatically installed with the distribution of L<sup>A</sup>T<sub>E</sub>X I included above).
  - On Linux, in TeXmaker, navigate to Options → Configure TeXmaker → Quick Build and add `latexmk -pdf %.tex` as a quick build command. If you need to for something else in the future, you can edit this command or make a different one.
- L<sup>A</sup>T<sub>E</sub>X is very powerful. It can do much more than make a simple text-based document. You can add images, hyperlinks, flow charts, musical scores, circuit diagrams, and much more. I will **not** cover everything L<sup>A</sup>T<sub>E</sub>X can do in this three-part tutorial series. I am only covering some of the more important basics and a select few of the more advanced features. If there is something you would like to do and it is not included in this tutorial series, please look it up online; there is often a package that allows you to do whatever you need doing.
  - A good source to look for answers is on the T<sub>E</sub>X StackExchange [here](#).
- After compiling your .tex file, some auxillary files will be added into the location your .tex file is. These auxillary files are simply used by L<sup>A</sup>T<sub>E</sub>X to help compile. You can ignore these files and only need to include the .tex when sending it to someone. It will also create a PDF file in the same location. It is up to you whether or not you want to include this when sending someone the .tex file.

## 3 General Formatting of a L<sup>A</sup>T<sub>E</sub>X Document

- When writing in L<sup>A</sup>T<sub>E</sub>X, indentation does not matter, but if you open the .tex file associated with this PDF, you can see the indentation convention I follow.
  - I indent based on “level.” This means that each time I go a level deeper (eg: section to subsection), I add one more indent.

- Additionally, I add an empty line between each unrelated piece of the document.
- A  $\text{\LaTeX}$  document's first line should always be a declaration of the document class.
  - This is done by typing `\documentclass{x}` where x is the type of document you are making.
  - The different document classes and their uses are as follows:

article	For articles in scientific journals, presentations, short reports, program documentation, invitations, et cetera.
IEEEtran	For articles with the IEEE Transactions format.
proc	A class for proceedings based on the article class.
report	For longer reports containing several chapters, small books, thesis, et cetera.
book	For full books.
slides	For slides. The class uses big sans serif letters.
memoir	For changing sensibly the output of the document. It is based on the book class, but you can create any kind of document with it.
letter	For writing letters.
beamer	For writing presentations.

- For more information please look [here](#).
- When you want to begin actually writing your document you use `\begin{document}`.
  - This allows you to start writing out text and using other commands shown later in this tutorial (such as `\section`).
- The area between `\documentclass` and `\begin{document}` is called the preamble.
  - This is the area where you set up the various things you may need for your document. You can use things like `\usepackage`, `\author`, and other things. I will cover these in more detail later.
- Anything that uses `\begin{x}` needs a `\end{x}` after all of the content inside that block.
- When trying to use reserved characters (characters reserved for commands in  $\text{\LaTeX}$ ), you need to use `\` to escape them. (Ex: To type \$ you need to type`\$`)

- Reserved characters include \$, {, }, #, %, \, and many others.
- An easy way to tell if the character you are using is a reserved character is the fact that it will be a different color from the rest of your text.
- The one exception to the rule of using \ to escape reserved characters is \ itself. To type \ in text you would use \backslash.
- To comment out a line in L<sup>A</sup>T<sub>E</sub>X (so that line isn't read at compile-time), preface the line with %.
- To type a new-line character in L<sup>A</sup>T<sub>E</sub>X, you use a \\. This is the equivalent of hitting return in various other word processors.
- If you wish to add text directly after a command that does not use {}, you need to encapsulate the command in \$'s.
  - To type \example in L<sup>A</sup>T<sub>E</sub>X you need to encapsulate the \backslash command in \$'s like this: `$\backslash$example`

## 4 Title Page

- To add a title page, simply add \title{Title Name Here}, \author{Author Name Here}, and \date{Date Here} to the preamble.
  - Note: If you would prefer to have the date be updated at compile-time rather than you having to change the date each day you are working on a document, leave the \date{Date Here} out of the preamble. In this case, the date will still be created and added to the title page, but it will automatically update what date it shows each time you compile your PDF.
- After your \begin{document} declaration add the following just below it:
  - \maketitle
  - \newpage
- If you don't want your title page numbered, add this command just above the \maketitle declaration:
  - \pagenumbering{gobble}
- You can also use \pagenumbering to begin the numbering in arabic numbering or roman numeral numbering. This can be done with either of the following commands:
  - \pagenumbering{arabic}
  - \pagenumbering{roman}
- If you call \pagenumbering{gobble} on your title page, your document will not start numbering things again until you call \pagenumbering{arabic/roman}.

## 5 Table of Contents and Hyperlinking

- To hyperlink in L<sup>A</sup>T<sub>E</sub>X, you need the hyperref package.
  - Get this by typing `\usepackage{hyperref}` in your preamble.
- To set up your hyperlinks to look nicer, add the following to your preamble after your `\usepackage` declarations:

```
\hypersetup{
  colorlinks = true,
  urlcolor = blue
}
```

- `colorlinks` colors your links rather than adding a box around them.
  - `urlcolor` sets the color of URL links to whatever color you choose. Default is a reddish-pink and here I set it to be a more common blue.
- If you plan on having a Table of Contents as well, use the following instead:

```
\hypersetup{
  colorlinks = true,
  linktoc = all,
  linkcolor = blue,
  urlcolor = blue
}
```

- `linktoc` determines what is linked on your table of contents. Here I set it to all, and thus everything on the table of contents for this document is linked.
  - `linkcolor` sets the color your table of contents links are. This color selection is separate from the URL hyperlink color set with `urlcolor`.
- To add a Table of Contents to your document simply use these two commands after the `\newpage` of your title page declaration:
  - `\tableofcontents`
  - `\newpage`

## 6 Section Types: Main Section

- Made by using `\section`.
- Sections change the number that comes before the first decimal and resets any subsections that come after it.



- If you were on section 1.2.6 and you called `\section`, you would be on section 2.
- Any text you want written in the section can go directly below your `\section` declaration.

## 6.1 Section Types: Subsection

- Made by using `\subsection`.
- Subsections change the number that comes after the first decimal and resets any subsections that come after it.
  - If you were on section 1.2.6 and you called `\subsection`, you would be on section 1.3.
- Any text you want written in the subsection can go directly below your `\subsection` declaration.

### 6.1.1 Section Types: Subsubsection

- Made by using `\subsubsection`.
- Subsubsections change the number that comes after the second decimal.
  - If you were on section 1.2.6 and you called `\subsubsection`, you would be on section 1.2.7.
- Any text you want written in the subsubsection can go directly below your `\subsubsection` declaration.
- There is nothing lower than a subsubsection. A subsubsubsection does not exist.

## 7 Paragraphs

### Main Paragraph

- Made by using `\paragraph`.
- Any text you want written in the paragraph can go directly below your `\paragraph` declaration.

### Subparagraph

- Made by using `\subparagraph`.
- Any text you want written in the subparagraph can go directly below your `\subparagraph` declaration.

## 8 Text Formatting

### 8.1 Font Size

- The following commands allow you to change the font size in L<sup>A</sup>T<sub>E</sub>X relative to the default document font size:

Command	Example	Description
<code>\tiny</code> Enter Text Here	Example Text	Makes the font size two units smaller.
<code>\small</code> Enter Text Here	Example Text	Makes the font size one unit smaller.
<code>\large</code> Enter Text Here	Example Text	Makes the font size one unit larger.
<code>\huge</code> Enter Text Here	Example Text	Makes the font size two units larger.

### 8.2 Font Style

- The following commands allow you to use different font styles in L<sup>A</sup>T<sub>E</sub>X:

Command	Example	Description
<code>\textbf{}</code> Enter Text Here	<b>Example Text</b>	Bold
<code>\textit{}</code> Enter Text Here	<i>Example Text</i>	Italic
<code>\texttt{}</code> Enter Text Here	<b>Example Text</b>	Typewriter
<code>\textrm{}</code> Enter Text Here	Example Text	Serif (Roman)
<code>\underline{ }</code> Enter Text Here	<u>Example Text</u>	Underline

### 8.3 Verbatim

- If you want to type a block of code or anything that may contain L<sup>A</sup>T<sub>E</sub>X commands, you can use the verbatim environment using `\begin{verbatim}` and `\end{verbatim}`.
  - The verbatim environment will write whatever is in it as text without running any commands that may be in it.

- The verbatim environment can't recognize tabs, but can recognize spaces. If you need to use a tab, convert them into spaces. I've found that six spaces indents similarly to a tab in the L<sup>A</sup>T<sub>E</sub>X editor.
- If you want to use the verbatim environment in-line, then you can use `\verb|text here|`.

## 9 Lists

### 9.1 Bulleted List

- To make a bulleted list in L<sup>A</sup>T<sub>E</sub>X, you need to use the *itemize* declaration within `\begin{}` and `\end{}`.
  - This will look like this: `\begin{itemize}` and `\end{itemize}`.
- For each bullet point you need to use `\item` followed by a space and then whatever it is you want after the bullet. This will go between the `\begin` and `\end`.
- If you want to make a sub-bullet point (the dashes in this document), within the `\begin{itemize}` and `\end{itemize}`, you need to add another `\begin{itemize}` and `\end{itemize}`.
  - Between the second set of *itemize*, you use `\item` followed by a space and then whatever it is you want after the bullet.

### 9.2 Ordered List

- To make a numbered list in L<sup>A</sup>T<sub>E</sub>X, you need to use the *enumerate* declaration within `\begin{}` and `\end{}`.
  - This will look like this: `\begin{enumerate}` and `\end{enumerate}`.
- For each bullet point you need to use `\item` followed by a space and then whatever it is you want after the bullet. This will go between the `\begin` and `\end`.
- If you want to make a sub-enumeration point (indenting and labeling with a, b, c, et cetera instead of 1, 2, 3, et cetera), within the `\begin{enumerate}` and `\end{enumerate}`, you need to add another `\begin{enumerate}` and `\end{enumerate}`.
  - Between the second set of *enumerate*, you use `\item` followed by a space and then whatever it is you want after the bullet.

### 9.3 Combined Lists

- You can use both itemize and enumerate together. This will give you a mix of bulleted and numbered lists.
  - For Example: If you have a numbered list, but want to include a few bullet points regarding one of the numbers, you can nest an itemized list in it if you don't want to number those points in an order.
- You can nest as many lists as you would like.

## 10 Math

- To begin you should use the package `amsmath` by calling `\usepackage{amsmath}` in the preamble.
  - While  $\text{\LaTeX}$  does have a way to type an equation without a package (using `$$equation$$`), it is highly discouraged as `amsmath` is better in every way. It uses the default  $\text{\LaTeX}$  math as a base, so there shouldn't be anything in default  $\text{\LaTeX}$  math that you can't do in `amsmath`.
- When writing math, you want to use `\begin{equation}`, your equation, and then `\end{equation}`.
- Using this without a package will automatically number each line.
- To avoid this, you can use `\begin{equation*}`, your equation, and then `\end{equation*}`.

Here is an example of  $f(x) = x^2/4^\sigma$  as a  $\text{\LaTeX}$  equation:

$$f(x) = x^2/4^\sigma \tag{1}$$

Here is an example of  $f(x) = x^2/4^\sigma$  as an `amsmath` equation:

$$f(x) = x^2/4^\sigma$$

- In  $\text{\LaTeX}$ , you can type a bunch of mathematical symbols using various commands. As seen above, `\sigma` will get you  $\sigma$ . Please see [here](#) or [here](#) for information relating to what commands can be used.
- To use an equation or any math command within a normal sentence (such as this one), use `\(equation here\)` wherever you want the equation in the sentence.

- The parenthesis escaped by the `\` will not show up in the equation.  
If you want any parenthesis in your equation, use non-escaped parenthesis inside the escaped parenthesis.
- Here are a few example commands:
  - You can use the command `\frac{x}{y}` to make a fraction look like  $\frac{x}{y}$  instead of  $x/y$ .
  - You can use the command `x \times y` or `x \cdot y` to make multiplication look like  $x \times y$  or  $x \cdot y$  instead of  $x * y$ .
  - `\sqrt{x}` will make  $\sqrt{x}$ .
  - `x_{y}` will make a subscript such as  $x_y$ .
  - `x^y` will make a superscript (or exponent) such as  $x^y$ .
  - `\text{text here}` will allow you to type regular text within an equation block.
  - Encapsulating your equation in `\boxed{equation here}` will allow you to put a box around your equation.

## 11 Other Resources

Here is a list of other great resources regarding learning LaTeX. Most of these were used throughout making this tutorial and contain some information not included in this tutorial.

- [RPI L<sup>A</sup>T<sub>E</sub>X Intro](#)
- [A Simple Guide to L<sup>A</sup>T<sub>E</sub>X](#)
- [L<sup>A</sup>T<sub>E</sub>X Wikibook](#)
- [University of Edinburgh's L<sup>A</sup>T<sub>E</sub>X for Beginners](#)
- [T<sub>E</sub>X StackExchange](#)
- [Short Math Guide for L<sup>A</sup>T<sub>E</sub>X](#)
- [L<sup>A</sup>T<sub>E</sub>X Math Command Cheat Sheet](#)
- The .tex file included in the zip for this document has everything that was used to write this tutorial. Please look through it to see how things in this document were used in practice.