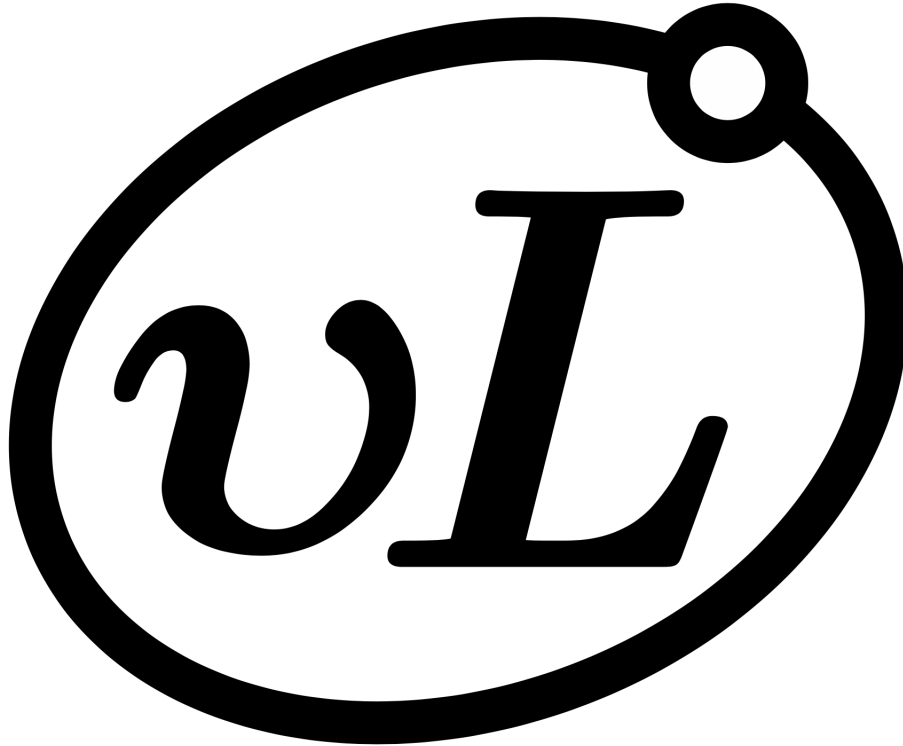# Upsilon Lab *Getting Started with LaTeX*
an Official UCLA Physics & Astronomy Department Sponsored Organization

Jacob Saret, Founder-President
May 17, 2018

"A lab for undergraduates, by undergraduates."

1   Summary

The goal of this guide is to help the reader begin using LaTeX in the context of most things a Physics & Astronomy Department major will do as an undergraduate. For now, you will use most tenets of LaTeX in lab reports, CVs, etc, but all of these tenets can easily be translated to a whole host of other things, from writing papers to other academic work. This technical guide is designed to take you from zero to LaTeX with ease.

Contents

2   Getting Started & Using a Compiler

I personally prefer a desktop-based compiler, but for that, the learning curve is a bit steeper, and there is wide variety of different choices. As a Mac user, my go-to is TeXShop, which can be downloaded at the TeX Users Group (TUG) site, `http://www.tug.org/mactex/morepackages.html`.

Of my colleagues here at UCLA, I do not know any Windows users who use a desktop compiler. They, and generally most people, use an online compiler like Overleaf (`overleaf.com`) or ShareLaTeX (`sharelatex.com`). Getting started with either of these services is quite straightforward.

2.1   TeX is Completely Open-Source

Never pay **anything** for anything associated with using LaTeX. Everything is completely open-source. Fortunately, most developers abide by this, but anything you will need can be found easily at no cost.

2.2   Internet Resources

Search engines are your friend. I learned everything about what I know of LaTeX through scouring the internet for help. Chances are, if you want to do it, it's already been done, and there's no need to reinvent the wheel. If there's something you want to do in LaTeX which I don't cover in this guide, I am almost certain you will be able to find an example of code with it already done.

It's also worth noting that ShareLaTeX has a great repository of helpful templates and guides, and of course you can't go wrong trying a solution found on StackExchange (`tex.stackexchange.com`).

Another excellent resource is the Comprehensive TeXArchive Network, or CTAN. CTAN has many packages (á la \usepackage{}) which you need to download if they are not included in your desktop compiler. It is worth noting that I have never needed to use a package not included in TeXShop, but they exist to fill their niches.

3   Coding

Every document is made up of four pieces:

1. The preamble. All package imports, page geometries, and other document-defining matter.
2. \begin{document}
3. The document. All body text, figures, graphics, etc.
4. \end{document}

Whenever I reference a package by calling \usepackage{}, it goes in the preamble. Most packages can be thrown in without issue, but there may be conflicts. I rarely experience these, and they can always be solved by a web search. Everything else goes between \begin{document} and \end{document}.

3.1 A Little Parameter Called Badness

The way LaTeX decides how to format your document, especially with all the automatic spacing it does, is by minimizing a parameter the creators call "badness."

Badness is an inherently positive quantity, that is, a badness of zero is awarded to "perfect" documents. Don't worry about seemingly excessive badness scores, as a single "underfull hbox"- the most common badness award- is 10,000 badness. Usually badness is irrelevant to the casual TeXer, but it's worth knowing the terminology, and I will reference badness again throughout the guide when explaining the logic of LaTeX.

4 The Preamble

There are several pieces to the preamble: document class, package imports, and document-defining commands. This "mini-preamble" will do just about everything you need to get started.

1. `\documentclass[11pt, oneside]{extarticle}`
2. `\usepackage{geometry}`
3. `\geometry{letterpaper,margin=1in} %\geometry{landscape}`
4. `\begin{document}`

Of course, you'll want to add lots of packages, such as `xcolor`, `fancyhdr`, `graphicx`, or `hyperref`:

`\usepackage[optional-packageoptions]{packagename}`: the command to use to import any package you need.

5 The Document

Here, you can pretty much just write plain text as you would into word, but without fiddling with the formatting in the meantime. Regular text will do exactly what you expect. Accents are easy- don't type in the character as you would in Word or other such software, but they are straightforward. For example, "ç" is printed by typing `\c c`. Yes, that's two "c"s in a row. Other modifiers are similarly self-explanatory, for example, `\noindent` at the beginning of a line will remove all indentation.

If you want to start a new paragraph, just add a blank line between the lines of text. You can write long paragraphs by splitting the text between lines with **no** blank lines, just a single return.

5.1 Basic Body Formatting

Use one or two ` (on the same key as ∼) to make the left side of a 'quote' or "quote".

Text &s and $s require a backslash since those characters have their own special purposes- table and equation delimiters.

Force a new paragraph by adding a blank line between lines of text, or a double backslash: \\.

\vspace{5mm} (with any length) is a very useful command to fiddle with spacing- if you use it often, you're doing something wrong, but it's useful on occasion.

Bold, *italics*, *slanted text* **__and__** underlining? It's just as easy as it is garish. All you need is \textbf{Bold}, \textit{italics}, \textsl{slanted text} and \underline{italics}. Note that *italics* match the variables in equations, but *slanted text* is a bit cleaner.

5.2 Itemization

Make lists with \begin{enumerate} or \begin{itemize}, and delineate each item with a \item. These environments don't even require new lines. Don't forget to close out the environment with a \end{enumerate} or \end{itemize}.

5.3 Equations

All you need for math is an opening and closing \$ for inline math, or \$\$ for separated equations.

Within the math environment, almost everything is self-explanatory. Some examples include ^ for exponents; _ for subscripts; \int, \iint, \iiint for $\int$, $\iint$ and $\iiint$.

Just about any symbol you'd need to type in LaTeX's math mode can be found from a quick web search.

One of the biggest "rookie mistakes" TeXers make is bad parenthetical formatting, for example, Equation (1).

$$\omega^2 = (gk + k^3 \frac{\sigma}{\rho}) \tanh(kH) \tag{1}$$

Contrast this to Equation (2).

$$\omega^2 = \left(gk + k^3 \frac{\sigma}{\rho}\right) \tanh(kH) \tag{2}$$

Let's compare the math code for both equations- I've highlighted the differences.
(1). \omega^2=   (gk+k^3\frac{\sigma}{\rho}   )\tanh(kH)
(2). \omega^2=\left(gk+k^3\frac{\sigma}{\rho}\right)\tanh(kH)

I recommend using \left and \right on every parenthetical, including square and curly brackets- it will never hurt, and if you modify your equations you'll never have to worry about it.

5.4 Multi-Columns

Most academic documents are typeset in a double-column format. It's quite simple to put a block containing text, tables, and equations into multiple columns with \usepackage{multicols}. There's two choices when it comes to writing in a multi-column format: flat and jagged, as shown below, each separated by a \hrulefill.

Writing in the \begin{multicols}{} environment will produce a flat-bottom result, i.e. it will fiddle with margins around figures, equations, and between paragraphs to make sure the columns are all the same height. Of course, if there aren't any margins for LaTeX to tweak, the rightmost column may be a little short, as it is here.

Writing in the \begin{multicols*}{} environment will fill the leftmost column to the bottom of the page before starting the next column, repeating until the rightmost column is filled, then the columns would begin filling on the next page. As you can see here, there's barely enough text in this \begin{multicols*}{} environment to spill into the second column.

It's worth noting that ending a \{multicols*}{} environment forces a \newpage.

5.5 Figures & Graphics

Adding figures and graphics into your document is a breeze with \usepackage{graphicx}.

Make sure the image you want to include is in the same directory as your `.tex` file, then import using a modified version of this "MWE" (minimal working example):

1. \begin{figure}
2. \includegraphics[scale=0.05]{jacobsaret} % the image extension is optional
3. \caption{A picture of the author taken in Alum Rock Park in San Jose, CA.}
4. \label{fig:author} % label goes here, see Section 7.
5. \end{figure}

This will give you the graphic seen below in Figure 1.

Figure 1: A picture of the author taken in Alum Rock Park in San Jose, CA.

5.6 Tables

Tables are perhaps the greatest shortcoming of LaTeX in its current form. My go-to solution is a downloadable Excel macro package called `Excel2LaTeX.xla`. Fortunately, this comes from a very reputable source, CTAN, as discussed in Section 2.2. It is found at https://ctan.org/tex-archive/support/excel2latex?lang=en.

This Excel-LaTeX Table converter is pretty easy to use- just type your table into Excel, then open the `.xla` file and enable the macro via pop-up. Once your table is ready, select the cells you want to be converted, and go to `Format > Convert Table to LaTeX`, then paste the code from the pop-up into your `.tex` file. *Protip: make sure each column is entirely aligned to left, center or right.*

You will want to do some reformatting, so let me show you the basic pieces of the \tabular environment. The MWE I use below is from Lab 4 of Physics 18L with Prof. Huang, with some nonsensical modifications to demonstrate the use of multicols{}{}{} and multirow{}{}{}. Note that \usepackage{mulitrow} is required for the latter.

1. `\begin{table}[htbp]`
2. `\centering`
3. `\caption{Fundamental Constants}`
4. `\begin{tabular}{l|r|r|r|c|l|l}`
5. `Const.  & Meas.  & Unc.  & Acc.  & Scale & Units & \% Disc.\\\hline`
6. `\(e\) & -1.46 & 0.09 & -1.60 & 10\(\^-19\) & C &  9.2\\\hline`
7. `\(m\) & \multirow{2}{*}{MRow} & 0.53 & 9.11 & 10\(^-31\) & kg &  4.0\\\hline`
8. `\(h\) & & 0.48 & \multicolumn{2}{r|}{MultiColumn} & Js & 13.5\\\hline`
9. `\end{tabular}`
10. `\label{table:fundamentals}`
11. `\end{table}`

Let's go through it line-by-line.

1. Starting a table and specifying the desired float. [h] puts the float "right here"; [t], top of this or next page; [b], bottom of a page; [p], on a separate page for floats, e.g. if you want to have color figure pages, for example. Providing more than one option allows LaTeX to pick from the ones you allow to minimize the badness; usually, it will pick the first or second one since the ones further down the list are given additional badness.

2. Center the table. Of course you can use `\raggedright` or `\raggedleft`.

3. A caption.

4. Begin the table itself, and specify the alignment of the columns with `l`, `c`, or `r`, and lines between columns with a bar: `|`.

5. A data line. Use a `&` to delineate columns. Of course, `\&` can put ampersands in tables. A double backslash `\\` delineates a new line, and each `\hline` adds a line between rows.

6. Another regular data line.

7. Here we can see an example of a `\multirow{2}{*}{MRow}` which creates a cell two lines high.

8. Here we can see an example of a `\multicolumn{2}{r|}{MultiColumn}` which creates a cell two lines wide with a bar on the right side. Note that any `multicols{}{}{}` always eats the bar to the left of the cell and must be replaced this way, if desired.

9. End the data table.

10. Cross-referencing label. See Section 7.

11. End the table float environment.

Table 1: Fundamental Constants

Const.	Meas.	Unc.	Acc.	Scale	Units	% Disc.
e	-1.46	0.09	-1.60	10^{-19}	C	9.2
m	MRow	0.53	9.11	10^{-31}	kg	4.0
h		0.48	MultiColumn		Js	13.5

6 Making a Table of Contents & Section Numbering

Making a table of contents is very simple and a default-LaTeX one is just fine- it's what I used on page 2 of this document. Simply drop a \tableofcontents where you want it. Following the command with a \newpage is optional.

The formatting of the table of contents changes as you change the section numbering, done with \renewcommand{}{} in the preamble. A MWE of the command I used to make the subsubsection numbering format seen here is given in Subsubsection 6.0*A*.

6.0*A* Changing the Section Numbering Format

Add a command of this form to the preamble:

\renewcommand{\thesubsubsection}{\thesubsection\textsl{\small\Alph{subsubsection}}}

7 Cross References

In almost every document, you'll want to point readers to a specific equation, figure, or section. LaTeX handles this beautifully. All your references will even automatically be linked to the specific location of the item in the document with the useful \usepackage{hyperref}.

All you need is a \label{example-label} hanging out on the same line as a \section{Example Section}, or within your figure or equation environment, both using the \begin{} and \end{} delimiters.

All you need to reference items, as I'm showing you here in Section 7: Section \ref{cross-ref}. This works just as well for figures, tables and equations, especially with \eqref{eqn:example-equation}, as it will match the formatting shown in Equation (3).

$$\int x dx = \frac{x^2}{2} + C \tag{3}$$

8 Exercises, or, Good Uses for LaTeX

1. Make a formula sheet for your next physics exam.

   - I recommend turning the fontsize down to `6pt` and enabling the landscape command from the example preamble, and using the full-line math mode. A corollary to this is finding the commands to change text size within a document in LaTeX via the internet.

   - Taking advantage of `\usepackage{xcolor}` to color-code all your important equations with `\textcolor{blue}{}` will make reading much easier.

   - Similarly, using 5-7 columns wide will greatly increase the packing efficiency. Choose the size based on the length of the individual equations- a 1-series student may be able to get away with seven columns, but a 115-series student may only be able to use four.

2. Write a lab report or reformat one from another document software like Word.

   - Make sure to practice with multiple columns, and add plenty of equations, tables and figures.

3. Transcribe your résumé or CV to LaTeX.

   - Take advantage of the easy formatting LaTeX provides, especially `\hfill` and `\hrulefill`.

4. *For the ambitious.* Try live-TeXing your notes in a math or physics class.

   - Setting up some stationary (i.e. format) you like beforehand is crucial.

   - Getting the greek letters can be time-consuming, but once you get the muscle memory and touch-typing down, it's a reliable way of taking notes!

5. *Also for the ambitious.* Define your own command- usually a shortcut for an oft-used clunky command- and implement it in your document.

   - You're on your own for this one!