

Quick Start Guide to L^AT_EX

Samuel Peana

February 2, 2012*

1 Why L^AT_EX?

L^AT_EX is essentially a scripting language which allows a user to specify how their document should be formatted by explicitly telling each element where to be on paper using built in functions. This is useful since it gives you a great deal of control over the formatting of your document by removing the automatic formatting. I am sure that we are all familiar with the joys of formatting WORD documents when any kind of slightly complex formatting is required, particularly for math. Thus we use L^AT_EX. This document is intended to give you the most basic introduction to L^AT_EX possible so that you can go out and start typing documents. However, first you need a L^AT_EX compiler and editing environment. To find one just go to google and type in “`latex editor`”. You will find them in abundance. Once you have set up your editor and L^AT_EX compiler, you are ready to type your first document.

2 Format of a Basic Document

A document consists of a header which tells the L^AT_EX compiler what type of a document this is and what settings to apply to the whole document. Imagine templates from WORD. Now, once you have finished specifying document global settings to your hearts delight, you probably want to include the actual document. To do this, you use the `\begin{document}` command followed by `\maketitle`. (This renders the title you created in the header). At this point

*The author learned L^AT_EX last week!

editing should get a lot more intuitive (as long as it is not math). Just type your document text and \LaTeX will automatically make the page and line breaks. Now, assumably you want to end your document at some point. To do this, at the end of the document we simply add `\end{document}`. That's all there is to basic \LaTeX document creation. Below is an example of a basic document in \LaTeX .

```
\documentclass[12pt]{report}
\title{Title Text Here}
\author{Your Name Here}
\begin{document}
\maketitle
Pulitzer Prize winning document.
\end{document}
```

Now, you must be wondering, “Why all the (irritating) `\`s?” Well, these `\` characters are used to delimit a \LaTeX command or function from regular text to the \LaTeX engine. As for the choice of indicator, I have no idea why Don Knuth chose `\`, but mine is not to question why . . . simply to do and to die.

3 Basic Mathematics Typesetting

Mathematics typesetting is actually very well done in \LaTeX , which is why it is frequently used to write mathematical documents. First, we have to tell \LaTeX that we will be typing math symbols. This is done by putting a `$` before and after the equation. For example `$$\frac{25}{13}$$` gives $\frac{25}{13}$. Note that `{}` is where the arguments of a function go. In the example above, the `\frac` function makes a fraction bar where the first set of brackets holds the numerator and the second set of brackets holds the denominator. You can also put other functions inside of these parameters. For example `$$\frac{\partial f}{\partial x}$$` will give $\frac{\partial f}{\partial x}$. Also notice the behaviour of `x`, `f` when placed besides the `\partial` function: essentially the whole function `\partial` is essentially treated as a character and the `x` and `f` add on to it without spaces. That is essentially how you typeset mathematics in \LaTeX . Have fun.

4 Basic Command List

4.1 General Functions and Common Symbols

`^{}:` superscript

`_{}:` subscript

`\begin{center} [text or equation] \end{center}:` Centers the text or equation between the begin and end command

`\bar{variable}:` \bar{y}

`\epsilon:` ϵ

`\eta:` η

`\theta:` θ

`\phi:` ϕ

`\newpage:` Makes a new page in the document

4.2 Mathematical Operators

`\frac{numerator}{denominator}:` $\frac{y}{x}$

`\partial:` ∂

`\int_{a}^{b}:` \int_a^b

`\mathrm{d}:` d

If you need other characters or operators not listed here, a quick google should give you plenty of results on how to generate that character or operator.