

Sample project

Andrew Lin

April 2024

This is a quick document I've put together to help explain how to use my style package (and for those newer to LaTeX, a few key facts that are helpful to know)! If there's anything else you think would be useful to add to this document, feel free to email me at `lindrew@stanford.edu` and I'll include it :) **There's an FAQ at the end with questions I've been asked.**

1 General overview

The `\usepackage{lindrew}` line above basically means that all of the commands in the `lindrew.sty` file are automatically imported into this document. The first 50 or so lines of that file are some common packages that are broadly useful for the documents I create, as well as a few customizable settings (for example, if you instead typed something like `\usepackage[serif]{lindrew}` in line 2, line 16 of my sty file is not applied and you get the default font instead). The middle part of the sty file is basically a long list of various shortcuts / adjustments that I've found useful for my own sake – you can add your own or remove them as you see fit. Everything after that is for the colored theorem boxes (also customizable).

Many people start their LaTeX documents with a bunch of commands in the preamble (loading in relevant packages, defining shortcuts) which they copy-paste into every new document they make. (This basically has the same effect as putting everything in a sty file.) In particular, if you're just making a bunch of documents for yourself (like I do for lecture notes), it's fine if you just keep adding more and more shortcuts even if you don't end up using them all the time; it shouldn't change the compile time too significantly.

1.1 Commands explanation

The main advantage of taking LaTeX notes is that it's fairly easy to include formulas. If you just want to include it seamlessly in your text, do something like $\sqrt[3]{2} + \pi \approx 4.4$, while if you want to separate it out onto its own line, you can either use double dollar signs or (my preference for code readability) brackets as shown below:

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1-2 \\ 3-4 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \end{bmatrix}.$$

If you want a formula or string of equalities to span multiple lines, consider using the `align*` environment and putting ampersands where you want the different lines to line up:

$$\begin{aligned} 88^2 &= (88 + 12)(88 - 12) + 12^2 \\ &= 100 \cdot 76 + 12^2 \\ &= 7600 + 144 \\ &= 7744. \end{aligned}$$

(Some people prefer to use `equation` instead of `align`, and there’s some reasoning [here](#) that might be relevant for that too. Personally I don’t really care about these kinds of details, and by extension some of the stuff in this package might be a little hacky; if that bothers you, feel free to make your own version of my package!)

You can take a look at the `sty` file for examples of how to create new shortcuts – probably by far the most common types are `DeclareMathOperator` (so you can type something like `\ord` and have it show up as `ord`), `newcommand` (to create a new shortcut), and `renewcommand` (if the command is already defined). When defining commands, you can either have them be faster re-definitions of symbols like \mathbb{E} , or you can use them to simplify common formats you repeatedly need to type like $\langle \psi | \phi \rangle$. In the latter case, you can specify a number of arguments (like the `braket` command does) and then specify where they should show up in the format (that’s what `#1`, `#2` are for).

Remark. *Do keep in mind that while shorthand like `\RR` for \mathbb{R} or using `paren` to get large parentheses that automatically wrap around taller expressions like*

$$\left(1 + \frac{1}{1 + \frac{1}{1 + \dots}}\right) = \frac{1 + \sqrt{5}}{2}$$

*are both very “standard” shortcuts that many people use, some of my shortcuts are **not** standardly named (like `parsh`, which I found useful in one physics class and don’t think I’ve ever used since). I should definitely put the disclaimer that nothing I do is particularly “optimized,” and you should feel free to explore adjustments or alternatives because there probably are many of them that I just haven’t needed.*

1.2 Using theorem boxes

(Note that if you don’t want the sections to be numbered, you can use `section*` instead of `section`. The same is true for all of the boxes below.)

Example 1

Here is an example box.

Theorem 2 (Pythagorean theorem)

Here is an example box with a particular label.

Definition 3

Here is a definition that I might want to label later on.

Later in the document, you can reference Definition 3, and clicking on the hyperlink will take you back to where it is in the document. The way those references are displayed is pretty customizable as well – feel free to look into the `hyperref` and `cleveref` packages if you have strong opinions.

Many people prefer to number their examples vs theorems separately (so they don't count up with the same counter), or they prefer to number with respect to the section or subsection number (so perhaps the above would read Example 1.2.1 instead of Example 1). Those can be controlled by adjusting the `numberwithin` or `sibling` parts of the style package (for example, not having `sibling=theorem` would make them count up separately, and using `numberwithin=section` would make everything in Section 1 be Example 1.1, Theorem 1.2, and so on). You can also load my package as `\usepackage[formal]{lindrew}`, which will automatically change the numbering.

2 Other FAQs

I'll add answers to any questions that people send by email!

How do you get more comfortable with LaTeX commands, formatting, etc?

If you're unsure how to write a particular symbol, try drawing it on [Detexify](#). For a cool typing game which gets you familiar with some common symbols, try [TeXnique](#). And if you want to make a particular kind of diagram (e.g. tables, tikz pictures, etc.), [Overleaf](#) tends to have good documentation. In practice, **I don't think there's any real reason to "learn all the commands"**¹ ahead of time – I only figured ever figured things out when they came up during classes or problem sets I was writing up. It's a bit of a tough learning curve at first, but you'll find yourself becoming more comfortable pretty quickly if you keep asking the question "how can I do this more easily?".

I'll point out in particular that Overleaf's explanation of [bibtex](#) and citation in general is very useful. Citing papers, books, etc. is pretty easy with LaTeX, since you can usually find the BibTeX citations linked on the pages where they are hosted.

Why Overleaf instead of something offline? (TeXstudio, VSCode, Emacs, etc.?)

Many of my friends who write lots of math documents aren't the biggest fan of Overleaf because it doesn't allow as much customization, and it doesn't work if the server goes down or if you have a spotty internet connection. I think the common advice is just "use whatever works best for you" – I've never had a need for super fancy macros and I've always preferred having everything in the cloud, so I've just stuck with Overleaf because it's made organization easy. Plus, it makes it extremely easy to share course notes with other people

¹Notice that you have to use different quotes for left and right quotes...

(I can just send the view link to a group chat while the class is ongoing). But if you prefer something else, I don't think there's any reason not to make that change!

Do you think live-TeXing has helped you learn better?

I personally don't recommend starting off by live-TeXing lectures in "full transcription" like I do. Unless you're already experienced and can type relatively quickly (see below for a quick one-try monkeytype I just did), it's very possible to fall behind or lose track of what the lecture is saying because you're trying to fix what was just written on the board or follow some chain of equalities.

(Typically people will use **figures** if they want to add captions, extra bordering, etc. to their images. That's not something I've had to pay too much attention to myself, but it might be a good read.)

I would absolutely say that I learn less well during lecture by doing this, and it's only traded off by being able to reread the lectures clearly afterward. (In particular, I often go back to past courses' lecture notes during school breaks and reread them to fix up formatting, fill in proofs, and refresh my knowledge – it's been a nice sense of progression for my learning.) However, I do know of a lot of people who find success in taking notes as usual but just doing them in bullet points on a LaTeX document, because it makes it easier to keep everything in one place. Either way, it can be useful to be able to ctrl-F for key definitions or phrases without needing to comb through the whole document. Really, it comes down to what purpose you specifically want your notes / other documents to serve!

When did you start live-TeXing?

I started using LaTeX around middle school, back when I was exploring competition math and browsing the AoPS forums a lot. I only tried live-TeXing notes for the first time at a summer camp before I entered undergrad (in the summer of 2018). Those notes were not very good, but it gave me enough practice that I wanted to try doing it in college too; I've been doing it ever since in every math, physics, and computer science class that I've taken. I'd estimate it took me about 2 years of college (so probably over a thousand pages of notes) to find a rhythm of formatting and "mathematical voice," but it has been very helpful for learning how to do this whole math communication thing.