

Help Topic: Getting Started with Latex

Spring 2020

Michael Benjamin, mikerb@mit.edu
Department of Mechanical Engineering
MIT, Cambridge MA 02139

Getting Started with Latex

We encourage the use of LaTeX in part because its quality, ubiquity, and natural support for writing across a team using version control software like SVN. Lastly, for students pursuing a graduate degree at MIT, you will very likely write your thesis using LaTeX, and there are good MIT thesis templates to help make this easy.

There is *sooo* much on the web to help you learn LaTeX, add new packages, and example documents and forums. We won't try to replicate that. However it seems a bit harsh to just say "go learn LaTeX on the web or the bookstore". So we have a bit of information here to get you started:

- Tips for installing LaTeX on your Mac or GNU/Linux machine,
- A few example LaTeX documents, including an MIT thesis template,
- A page of personal must-know features, tips, and pointers to resources on the web.

The best way to get started with LaTeX is to put it on your computer and start experimenting with working document examples. Even better if the examples are similar to what you may be creating in your immediate future. Most of basic workings will be apparent to you, e.g., that super results in super being underlined. Some of the more advanced features you can figure out by just googling the snippet of code you don't understand.

How to tell if LaTeX is already installed

Your machine may already have a version of `latex` installed. To check, try:

```
$ which latex
/usr/texbin/latex
```

If you get a response like the one above, you're all set. If `latex` is not installed you will get a response like the one below:

```
$ which latex
latex: Command not found
```

If the `which` command is telling you that it cannot find the `latex` command, you probably just don't have `latex` installed. However it may be that you simply don't have the directory containing the `latex` executable in your shell path. To rule this out, try the following (if using GNU/Linux):

```
$ cd /usr/bin
$ which ./latex
./latex
```

or (if using OSX):

```
$ cd /usr/texbin/latex
$ which ./latex
./latex
```

If you find the `latex` executable as above, add the directory to your shell path. If you're not sure how to do this see the help topic on augmenting your shell path:

<http://oceanai.mit.edu/ivpman/help/help-add-shellpath>

Getting LaTeX in GNU/Linux

If you're running GNU/Linux you probably have access to a package manager such as `apt-get` and already know how to use it to some degree. To install packages like LaTeX, you need root privileges. Assuming your user account has root privileges, do the following, and enter your normal login password if prompted:

```
$ sudo apt-get install texlive-latex-base
$ sudo apt-get install texlive-latex-extra
```

See the discussion above on how to confirm whether `latex` has been successfully installed, and adjust your shell path if need be.

Getting LaTeX in OSX

If you're running on a Mac (OSX) you can use MacPorts to install LaTeX. If you haven't installed MacPorts, see the separate help topic on this. To install packages like LaTeX, you need root (admin) privileges. Assuming your user account has admin privileges, do the following, and enter your normal login password when or if prompted:

```
$ sudo port install texlive-latex
$ sudo port install texlive-latex-extra
```

See the discussion above on how to confirm whether `svn` has been successfully installed, and adjust your shell path if need be.

Other tools you will need to use LaTeX

LaTeX works by processing a raw text file and producing a file in a viewable format like `dvi` or `pdf` format. It makes no assumptions about what tool you use to build that raw input file. Nor does it

make any assumption about the tool you use to view the output document. So you need to have a couple additional tools handy. You likely already have them:

- On the front end you need to have a text editor like Emacs or vim. Both editors have support for LaTeX and will render components of the language in proper colors and provide some support for auto formatting. In our class we recommend using Emacs for those not already using another good text editor.
- On the back end you need to have a tool for viewing pdf documents. In OS X, the Preview tool works just fine. Just type `open file.pdf` on the command line. There are several other pdf viewer tools available on the Mac. In GNU/Linux, there are tools such as `xpdf` and `xdvi` available through a package manager like `apt-get`.

In my opinion this is one of the many reasons for wide adoption of LaTeX. The users have *choices* in the tools for composing the raw files, as well as in the tools for viewing finished documents. Compare this to MSWord where everything is all wrapped into one tool.

Finally, we note here that this web page was compiled in LaTeX format, using Emacs. If you're viewing this as a pdf, the pdf was generated by `pdflatex` directly to a pdf. If you're viewing this in a browser, the wiki page was auto-generated by a home brewed program for converting LaTeX files to wiki page format.

Use the `pdflatex` command instead of `latex`

The `pdflatex` command should also be installed when you install LaTeX alongside the `latex` command.

```
$ which latex
/usr/texbin/latex
$ which pdflatex
/usr/texbin/pdflatex
```

You can think of `pdflatex` as an extension of `latex` that generates pdf files directly and contains certain typesetting improvements and extensions. It has become the de facto standard for many LaTeX users including myself. For me the biggest advantage is the better support for including png formatted graphics in figures.