

Help Topic: Examples for Getting Started with Latex

Spring 2020

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

Examples for Getting Started with LaTeX

The best way to get started with latex is to download it and start working with examples. So let's get started. We provide three examples:

- A dead simple bare bones example.
- A simple one-page memo in the style of a Navy memo.
- A template for an MIT thesis.

All three are available in an SVN repository available for download:

```
$ svn co https://oceanai.mit.edu/svn/latex_examples_2680
```

If svn or the above syntax is unfamiliar to you, read more at the link below. Getting svn is pretty easy, even if all you do with it for now is checkout the above repository.

<http://oceanai.mit.edu/ivpman/help/subversion>

A simple bare bones LaTeX example

Below is a very simple example. This example is in the tree mentioned above: `latex_examples_2680/barebones`. It contains the file `example.tex`. If you don't have the svn tree checked out, you can just cut and paste this small file from the clip below into a file called `example.tex`.

```
\documentclass[11pt]{article}

\begin{document}
\title{A LaTeX Example}
\author{Evan Eugene Samuelson III}
\maketitle

This is the first paragraph.

This is the second paragraph

\end{document}
```

You should be able to build and open the file with:

```
$ pdflatex example.tex
$ open example.pdf      (In OS X)
$ xpdf example.pdf     (In GNU/Linux with xpdf installed)
```

The output should look like:

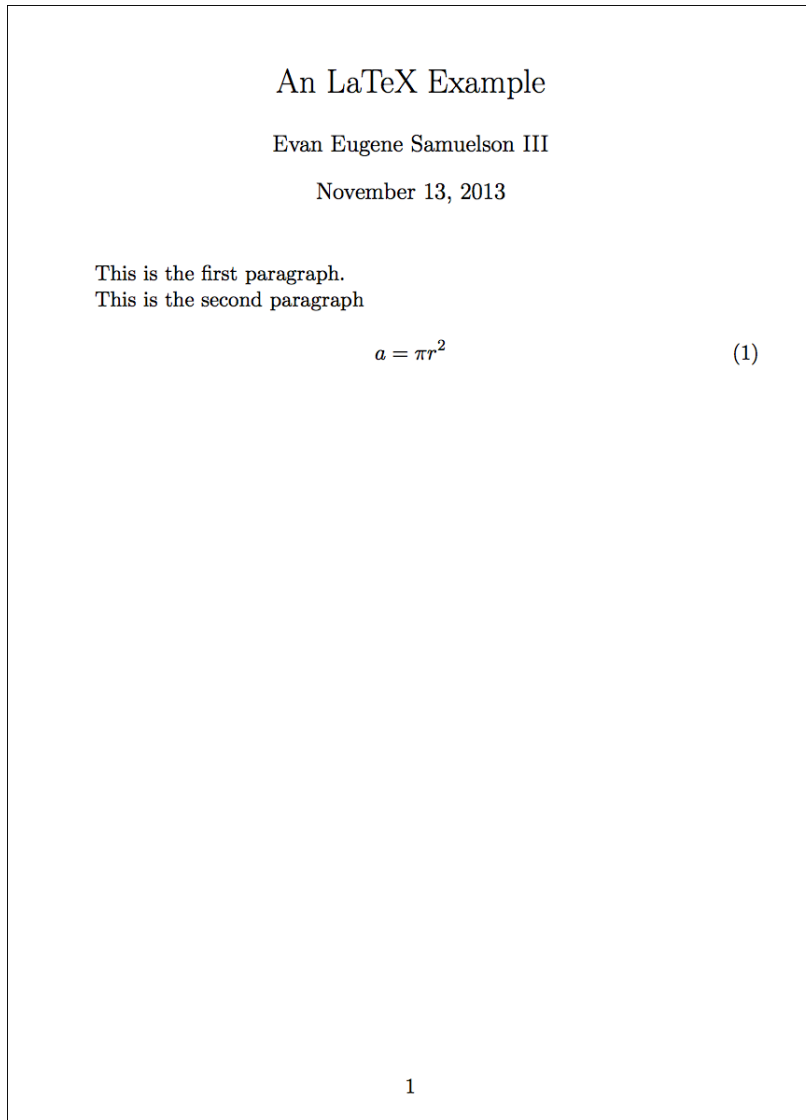


Figure 1: The single page generated from a bare bones example latex document.

A simple example using a Navy memo format

One of my more well-worn comments about about LaTeX used to be: "Using LaTeX is certainly how you want to write your thesis or journal paper, but if you want to write a letter to your mom, perhaps MSWord was more suitable".

Then at one point I found myself needing to write Navy memos rather frequently and I developed a simple template for this. Now I no longer recommend MSWord even for writing letters to your mom (although I recommend using a different format than the Navy memo format, but you get the idea). This example can be found in the latex examples svn tree.

You should be able to build and open the file with:

```
$ cd latex_examples_2680/navy_memo
$ pdflatex memo.tex
$ open memo.pdf          (In OS X)
$ xpdf memo.pdf          (In GNU/Linux with xpdf installed)
```

The output should look like:



Figure 2: The single page generated from a Navy memo example latex document.

An MIT thesis example

The last example is the most extensive and perhaps most useful for MIT students. It is also in the svn tree mentioned above. As before you should be able to build it with:

```
$ cd latex_examples_2680/mit_thesis
$ pdflatex main.tex
$ open main.pdf      (In OS X)
$ xpdf main.pdf      (In GNU/Linux with xpdf installed)
```

However - I recommend you begin using the GNU command line tool, `make`, to build the document instead. The benefits of `make` are discussed below.

```
$ cd latex_examples_2680/mit_thesis
$ make
$ open main.pdf      (In OS X)
$ xpdf main.pdf      (In GNU/Linux with xpdf installed)
```

Using `make` and Makefiles to build LaTeX documents

The GNU `make` utility and Makefiles have been around for very long time and are still essential for building code today. They are also useful for building LaTeX documents. Once you start producing documents with indices, bibliographies, and cross references, the build evolves from:

```
$ pdflatex paper.tex
```

to:

```
$ pdflatex paper.tex
$ bibtex paper.tex
$ makeindex paper.tex
$ pdflatex paper.tex
```

Putting these steps into a Makefile and invoking `make`, simplifies this process. The `make` command defaults to looking for a file named `Makefile` in the directory in which it was invoked. You can also launch a pdf viewer as part of the `Makefile`. In a sense, here `make` is acting only as a script. Here it could be replaced with other script tools such as a bash script, or perl script. However, `make` also has powerful tools for building sub-components of the thing your building and auto-detecting when those sub-components need to be re-built.

In the three examples we have discussed here, there are Makefiles for each. We do recommend you get in the habit of using `make`, and encourage you to read up a bit on this tool.