Getting something out of LaTeX

Jim Hefferon

This is for people considering using LaTeX. This is not a tutorial. Instead, it takes you through making a first document. If this quick taste leaves you wanting more then you are ready to go through a tutorial.

Because this is a just a taste, we will skip many things. And, for those things that we will discuss, we will see only one way to do them. For instance, in the first section there are a number of ways to get the software but we will just name one. This approach has the disadvantage of leaving you without a full understanding of the system but has the advantage that in a couple of hours you will know whether LaTeX is a tool that can help you.

Get the software

LaTeX is how we will use the TeX suite of programs. So you must download that suite, if you don't already have it. We will describe only options that are free.

If your computer system is Windows then download MiKTeX [**?**].[1] If you have a Unix-like system such as GNU/Linux then get TeX Live [**?**]. For a Macintosh get MacTeX [**?**], a version of TeX Live with some Mac-specific add-ons.

All three downloads are big but all three install easily (of course, you must carefully follow the directions; in particular on a Unix-like system you will need to alter your `PATH`). Here we are only giving LaTeX a try so we will skip most things.

Use an editor

We don't write LaTeX with a word processor. A word processor combines many of the jobs that must be done to produce a document, such as entering and moving the text, formatting it into paragraphs, producing a PDF file, etc. In a TeX-based system those jobs are done separately.

Instead, we write LaTeX with an editor, a program that is specialized at moving text around in a computer file. There are many editors, including some specifically for writing LaTeX, but below our documents are small so any editor will do. Just to name a couple of names: on a Windows system you can use Notepad, while on a Unix-like system or a Mac you can use Emacs.

Get it to work

Having picked an editor, you can write a first document. Make a new directory named `latex-first` (your system might use the term "folder"). Open a terminal window and in that window, change into that directory (possibly you would use the command `cd latex-first`).

Next start your editor. Open the new file `latex-first.tex` in the `latex-first` directory. Enter the text below, line for line, as it is written (without spaces on the left margin).

[1] For people with a printed version of this document we list links at the end.

```
\documentclass{article}
\usepackage{geometry}
\usepackage{fancyhdr}
\usepackage{amsmath,amsthm,amssymb}
\usepackage{graphicx}
\usepackage{hyperref}

\begin{document}
Hello world!
\end{document}
```

Many of those things are boilerplate — lines that I put into every LaTeX file. We'll ignore what they mean for a while to focus on getting out a first document.

Save that file. Go back to your terminal window and enter this command.

```
pdflatex latex-first
```

If that worked then you should see perhaps forty lines of text, starting with something like

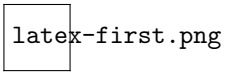
```
This is pdfTeXk, Version 3.1415926-1.40.9 (Web2C 7.5.7)
 %&-line parsing enabled.
entering extended mode
```

and ending with something like this.

```
Output written on latex-first.pdf (1 page, 6195 bytes).
Transcript written on latex-first.log.
```

If you had errors, see the subsection below.

You can view the output file `latex-first.pdf` with whatever program your system uses to view PDF.



Handle errors

Check that you have typed the lines exactly. Some seemingly small typing changes give large changes in the result, including causing a document to not work. So that you can be sure there isn't a typing discrepancy, you can get a known-good copy of `latex-first.tex`; see this document's source [**?**].

If your run ends with a question mark, then you can type '`x`' and hit the 'Enter' key to get out.

LaTeX's error messages can be hard to understand. If you know someone with some experience, of course that's great. If not, I've had good luck with putting the error message into a search engine.

Get more out

The first document is short so that it has fewer parts to go wrong. But we've already seen some basics. The file you type mixes text and commands. The commands for LaTeX, such as

`\begin{document}`, start with a backslash and sometimes have arguments contained in curly braces (or, we'll see below, sometimes square brackets).

The document we've made starts out with margins, typeface, etc., specified in the class `article`. We've modified the behavior in small ways by bringing in some packages such as `graphicx`, which will allow us to include graphic files.

Next we'll make a longer and more complex document. Start the same way: make a new directory named `latex-second`, open a terminal window, and change into the new directory. Then move back to your editor window and open a new file `latex-second.tex` in the `latex-second` directory. Enter this.

```
\documentclass{article}
\usepackage{geometry}
\usepackage{fancyhdr}
\usepackage{amsmath,amsthm,amssymb}
\usepackage{graphicx}
\usepackage{hyperref}
\usepackage{lipsum}

\begin{document}
This is some preamble text that you enter yourself.

Below is a command that will automatically generate seven paragraphs
of text that is commonly used for examples in this field.

\lipsum[1-7]
\end{document}
```

Save that file and in the terminal window run `pdflatex latex-second`. Your system's PDF reader should show this.

If you have used a word processor then you will have noticed the difference between that tool and LaTeX. A word processor moves the text around as you type it. With LaTeX, you describe what you want and then it goes off and figures out how best to do that. For instance, below we will tell LaTeX to make a section and the system handles the font changes, vertical space, etc.

In your editor, change the file `latex-second.tex` to say this.

```
\documentclass{article}
\usepackage{geometry}
\usepackage{fancyhdr}
\usepackage{amsmath,amsthm,amssymb}
\usepackage{graphicx}
\usepackage{hyperref}
\usepackage{lipsum}

\begin{document}
This is some preamble text that you enter yourself.
```

```
\section{Text for the first section}
\lipsum[1]

\subsection{Text for a subsection of the first section}
\lipsum[2-3]

\subsection{Another subsection of the first section}
\lipsum[4-5]

\section{The second section}
\lipsum[6]

\subsection{Title of the first subsection of the second section}
\lipsum[7]
\end{document}
```

Here is the resulting output.

Note that numbering of the sections and subsections is done automatically. How to change how they look is beyond this document, but the point is that the system generates those for you.

With those, we can illustrate cross-references. Change the document's text to this.

```
\begin{document}
This is some preamble text that you enter yourself.

\section{Text for the first section}
\lipsum[1]

\subsection{Text for a subsection of the first section}
\lipsum[2-3]
\label{labelone}

\subsection{Another subsection of the first section}
\lipsum[4-5]
\label{labeltwo}

\section{The second section}
\lipsum[6]

Refer again to \ref{labelone}.
Note also the discussion on page \pageref{labeltwo}

\subsection{Title of the first subsection of the second section}
\lipsum[7]
\end{document}
```

Run `pdflatex latex-second` and look at the PDF file. Notice that the references that we just entered didn't work — they appear as question marks. As LaTeX runs it saves labels to a file. When you run a file with a first-time label it has not yet been saved. The question marks go away when you run `pdflatex latex-second` a second time.

(When you are writing a real document, because you are fixing typos, etc., you will run LaTeX a number of times, so in practice having to rerun the command isn't an issue.)

We'll finish by adding footnotes, a table of contents, and a bibliography. Edit `latex-second.tex` to say this.

```
\documentclass{article}
\usepackage{geometry}
\usepackage{fancyhdr}
\usepackage{amsmath,amsthm,amssymb}
\usepackage{graphicx}
\usepackage{hyperref}
\usepackage{lipsum}

\title{Test document}
\author{Your name \\ \url{you@example.com}}
\date{2009-Oct-12}
\begin{document}
\maketitle
\tableofcontents
\newpage

This is some preamble text that you enter
yourself.\footnote{First footnote.}\footnote{Second footnote.}

\section{Text for the first section}
\lipsum[1]

\subsection{Text for a subsection of the first section}
\lipsum[2-3]
\label{labelone}

\subsection{Another subsection of the first section}
\lipsum[4-5]
\label{labeltwo}

\section{The second section}
\lipsum[6]

Refer again to \ref{labelone}.\cite{ConcreteMath}
Note also the discussion on page \pageref{labeltwo}

\subsection{Title of the first subsection of the second section}
```

```
\lipsum[7]

\begin{thebibliography}{9}
\bibitem{ConcreteMath}
Ronald L. Graham, Donald E. Knuth, and Oren Patashnik,
\textit{Concrete Mathematics},
Addison-Wesley, Reading, MA, 1995.
\end{thebibliography}
\end{document}
```

Run `pdflatex latex-second` (in the `\begin{thebibliography}{9}` line, the `9` tells LaTeX that the widest reference has one digit).

Try changing the margins by altering the second line to `\usepackage[margin=1in]{geometry}`. You can also experiment with the headings by changing the `fancyhdr` line to this.

```
\usepackage{fancyhdr}
\pagestyle{fancy}
\lhead{\today}
\chead{}
\rhead{Test document}
\lfoot{}
\cfoot{\thepage}
\rfoot{}
```

Get math

Many people interested in LaTeX want to include mathematics. The examples below are from *Concrete Mathematics* [**?**]. (We'll stop giving the complete listings of the input files, and we'll stop showing screenshots to instead just give the output directly.)

Add this text, for instance before the bibliography.

```
There are $\binom{2n+1}{n}$ sequences with $n$ occurrences of
$-1$ and $n+1$ occurrences of $+1$, and Raney's lemma
tells us that exactly $1/(2n+1)$ of these sequences have all
partial sums positive.
```

It produces this output.

> There are $\binom{2n+1}{n}$ sequences with n occurrences of -1 and $n+1$ occurrences of $+1$, and Raney's lemma tells us that exactly $1/(2n+1)$ of these sequences have all partial sums positive.

This input

```
Elementary calculus suffices to evaluate $C$ if we are clever enough
to look at the double integral
\begin{equation*}
  C^2
  =\int_{-\infty}^{+\infty} e^{-x^2} \mathrm{d}x
   \int_{-\infty}^{+\infty} e^{-y^2} \mathrm{d}y\;.
\end{equation*}
```

gives this.

> Elementary calculus suffices to evaluate C if we are clever enough to look at the double integral
>
> $$C^2 = \int_{-\infty}^{+\infty} e^{-x^2}\mathrm{d}x \int_{-\infty}^{+\infty} e^{-y^2}\mathrm{d}y\ .$$

And this source

```
Solve the following recurrence for $n,k\geq 0$:
\begin{align*}
  Q_{n,0} &= 1
  \quad Q_{0,k} = [k=0];  \\
  Q_{n,k} &= Q_{n-1,k}+Q_{n-1,k-1}+\binom{n}{k}, \quad\text{for $n,k>0$.}
\end{align*}
```

produces this result.

> Solve the following recurrence for $n, k \geq 0$:
>
> $$Q_{n,0} = 1 \quad Q_{0,k} = [k = 0];$$
> $$Q_{n,k} = Q_{n-1,k} + Q_{n-1,k-1} + \binom{n}{k}, \quad \text{for } n, k > 0.$$

The `\usepackage{ams...` line in our source files allow us to use the American Math Society's packages. For example, the `align*` above is available because we used `amsmath`.

We also have access to the AMS's symbols. A simple example is that we can get $\mathbb{Z}$ with the command `$\mathbb{Z}$`. One more example shows a long arrow, and some other useful commands

```
Therefore
\begin{equation*}
a\equiv b\pmod{m}
\qquad\Longleftrightarrow\qquad
a\equiv b \pmod{p^{m_p}}\quad\text{for all $p$}
\end{equation*}
if the prime factorization of $m$ is $\prod_p p^{m_p}$.
```

that produce this.

> Therefore
>
> $$a \equiv b \pmod{m} \qquad \Longleftrightarrow \qquad a \equiv b \pmod{p^{m_p}} \quad \text{for all } p$$
>
> if the prime factorization of m is $\prod_p p^{m_p}$.

(The *Comprehensive LaTeX Symbols List* [**?**] shows the widely-available symbols.)

The `amsthm` package gives us access to theorem environments, but those go beyond the scope of this document.

Got it?

You now have a feel for LaTeX. To go on, see the tutorial *The Not-So-Short Guide to LaTeX2e* [?] or the Indian TeX Users Group's tutorial [?]. More references are in the LaTeX Document Pointer [?].

If what you've seen seems very different from what you are used to, and you'd like an overview of the advantages of LaTeX, see Why TeX? [?].

When you take up LaTeX for real-life documents, you need to choose a good editing program. In particular, people often use systems where the editor is integrated with other components such as an output view, a spell checker that works with LaTeX, etc. For advice, ask LaTeX users that you know or click around on the Internet, for instance on `comp.text.tex` [?]. (For what it is worth, I use `emacs` with the AUCTeX add-on. On the Macintosh, many people use TeX Shop [?]. A cross-platform editor, developed for the TeX Users Group with the goal of simplicity, is TeXworks [?].)

References

[1] `http://mirror.ctan.org/info/symbols/comprehensive/symbols-letter.pdf`, *Comprehensive LaTeX Symbols List*, Scott Pakin, 2008.

[2] *Concrete Mathematics*, Ronald L. Graham, Donald E. Knuth, and Oren Patashnik, Addison-Wesley, Reading, MA, 1995.

[3] `http://groups.google.com/group/comp.text.tex/topics?gvc=2`, `comp.text.tex` usenet group, group authorship, 2009.

[4] `http://www.ctan.org/tex-archive/info/first-latex-doc/`, *Getting something out of LaTeX* (source for this document), Jim Hefferon, 2009.

[5] `http://www.tug.org.in/tutorials.html`, Indian TeX Users Group's tutorial, Indian TeX Users Group, 2003.

[6] `http://mirror.ctan.org/tex-archive/info/latex-doc-ptr/`, LaTeX Document Pointer, Jim Hefferon, others, 2009.

[7] `http://www.tug.org/mactex/`, MacTeX, Gerben Wierda, Jerome Laurens, Richard Koch, Herb Schulz, Karl Berry, others.

[8] `http://www.miktex.org`, MiKTeX, Christian Schenk, 2009.

[9] `http://mirror.ctan.org/info/lshort`, *The Not-So-Short Guide to LaTeX2e*, Scott Pakin, 2008.

[10] `http://www.tug.org/texlive/`, TeX Live, Sebastian Rahtz, Karl Berry, others.

[11] `http://www.uoregon.edu/~koch/texshop/`, TeX Shop, Richard Koch, 2009.

[12] `http://www.tug.org/texworks/`, TeXworks, Jonathan Kew, 2009.

[13] `http://www.tug.org/TUGboat/Articles/tb22-1-2/tb70heff.pdf`, Why TeX?, Jim Hefferon, *TUGBoat*, vol 22 no 1/2, 2001.

Author

Jim Hefferon
Saint Michael's College
January 13, 2016
ftpmaint@tug.ctan.org