

An Introduction to L^AT_EX

Simon Hollis

Department of Computer Science
University of Bristol

Outline of this lecture

- 1 Introduction to \LaTeX ?
- 2 What is \LaTeX ?
- 3 How does it work?
- 4 Producing a document in \LaTeX
- 5 Installing \LaTeX
- 6 Example
- 7 Reference

What is L^AT_EX?

- A typesetting language;
- A *markup* language;
- An extension to the T_EX markup language, written by Donald Knuth;
- A command that runs on a `.tex` input file;
- What I produced this presentation in!



How does it work?

As a markup language, it consists of *tags* (which \LaTeX calls **commands**), interspersed with text. For example, to make something **bold**, use the `\textbf` tag, with the text to bold in . Commands are stand-alone — they do not exist in pairs.

Example:

```
this is how you make some \textbf{bold} text.
```

produces:

this is how you make some **bold** text.

`\textit{}` works for italics, `\textrm{}` for roman, and so on.

`\emph{}` emphasises text regardless of current italicisation.

Input syntax

There are several reserved characters, plus space in \LaTeX .



e.g. `'%$&^#\backslash`

If you wish to use any (except `\`), then you need to escape them by placing a `\` first.

E.g. `\%` is printed by `\%`

`%` by itself starts a comment line

Space in the input file

Spaces also have a different meaning to normal: one space or a thousand has the same effect on the output.

And paragraphs are only started by a full line's space.
e.g.

This does not start
a new paragraph

This does not start a new paragraph.

This does start
a new paragraph.

This does start
a new paragraph.

Environments

As well as commands, \LaTeX also has **environments**.

Environments act in the same way as XML tags, starting a temporary state.

They are begun by

```
\begin{ }
```

and ended by

```
\end{ }
```

The environment's name appears between the two braces.

Environment example

The creation of a bullet-point list involves the combination of both an environment and commands. A list of bullet points is created by using the

`itemize` environment (which may be nested to produce further levels). Each entry in the list is then signalled by being preceded by a `\item` command.

So, a list of three items looks like this:

```
\begin{itemize}
```

```
\item First item.
```

```
\item Second item.
```

```
\item Third item.
```

```
\end{itemize}
```

Which creates:

- First item.
- Second item.
- Third item.

Producing a document in \LaTeX

- There are several steps to producing a document in \LaTeX .
- The good news is that, with a simple template, the *body* of the document can be *plain text* and produce an acceptable result.
- However, to make your document look pretty, or customise the format/fonts or include figures, you will need to master some \LaTeX commands (such as the `\textbf` command we saw earlier).

The document preamble

All documents have a **preamble**, which contains definitions of commands, packages to be used and document options.

It is begun by

```
\documentclass [] {}
```

(which **must** be the first non-comment line of your TeX file)

and ended by

```
\begin {document}
```

The document body

This is where the main body of your text goes, with `\chapter` and `\section` separators.

It is begun by

```
\begin{document}
```

and ended by

```
\end{document}
```

(which should be the last thing in your `.tex` file)

What can I edit \LaTeX in?

Anything! Any text editor will do. Here are just a few possibilities:

- Vim*
- Emacs*
- notepad
- gEdit
- Kile*
- LyX*
- eclipse*
- Word
- ...



Programs with an '*' have special \TeX modes that support auto-colouration, auto-complete, templates etc.

Linux

Use the 'TeXLive' distribution for ease of installation. Also install a .pdf/.eps/.dvi viewer to see the output.

Ubuntu/debian:

- 1 `sudo apt-get install texlive-latex-base
texlive-latex-extras texlive-latex-recommended`
- 2 `sudo apt-get install evince xpdf`

Other distributions:

Search for 'latex' via your package installer or go to
<http://www.tug.org/texlive/acquire.html>

Windows

- 1 Install the **Cygwin** environment, including the 'xpdf' package.;
- 2 Install **MikTeX**. <http://miktex.org/>
- 3 (Optional) Install Adobe Acrobat Reader[®] to see how the outside world will view your output.

Be aware that the full \LaTeX distribution is **LARGE** (hundreds of MB), but many of the most useful (and commonly-used) packages come only in the full/'extras' installation.

To convert your input file to a useable output file, you must run L^AT_EX on your `.tex` input file.

You have a choice: `latex`, which will produce a `.dvi` file out, or `pdflatex`, which produces a `.pdf` out. I recommend always using `pdflatex`.

The syntax is very simple:

```
pdflatex foo.tex
```

will produce `foo.pdf`

You may have to run `pdflatex` twice to get labels and references right.

Grand example



The best way to learn \LaTeX is to get stuck in!

It has a steep learning curve, but once you have mastered the basics your almost there!

Next I will show an example, demonstrating many features.

Some \LaTeX features

If we have time in the example, we will see:

- Features of the preamble
- Chapters, sections, subsections etc.
- Labels and references
- Bullet-points and numbered lists
- Math mode and equation typesetting
- Graphics
- Tables
- Command (re)definition
- Spaces and spacing
- Titles
- Other useful packages

We will also look at the use of citations and bibliographies via the **bibtex** tool, and how to use a **Makefile** to simplify the process.

Some useful online documents:

- <http://ctan.tug.org/tex-archive/info/lshort/english/lshort.pdf>
- <http://www.ctan.org/tex-archive/info/visualFAQ/visualFAQ.pdf>
- <http://www.tex.ac.uk/cgi-bin/texfaq2html?introduction=yes>

Examples on the unit webpage: <https://www.cs.bris.ac.uk/Teaching/Resources/COMS12303/LaTeX.html>

