

A quick guide to L^AT_EX

L^AT_EX is more of a programming language. The text you produce in your input file will look almost completely different from the final result in print. This requires knowledge of the commands that format your input into something that looks like typesetting by an “expert”.

A good book for getting started with L^AT_EX is:

David F. Griffiths and Desmond J. Higham, Learning L^AT_EX. SIAM, Philadelphia, 1996, ISBN 0-89871-383-8

Acknowledgment: This survey is based on the past notes of JHB Deane and S Reich.

The L^AT_EX and PostScript source for this document (outline.tex and outline.ps respectively) can be found at:

<http://www.maths.surrey.ac.uk/personal/st/Mark.Holland>

A collection of L^AT_EX resources on the Internet can be found in:

<http://www.maths.surrey.ac.uk/personal/st/J.Deane/Teach/latex/index.html>

Contents

1	The First Steps	3
2	Basic L^AT_EX	4
2.1	The Absolute Minimum	4
2.2	Plain Text	4
2.3	Type Styles	5
2.4	Environments	6
2.5	Vertical and Horizontal Spacing	7
3	More Basics	7
3.1	Titles for Documents	7

3.2	Sectioning Commands	8
3.3	Bibliography	9
3.4	Troubleshooting	10
4	Typesetting Mathematics	10
4.1	Examples	10
4.2	Equation Environments	10
4.3	Fonts and Indices	12
5	Introduction to tables	13
6	Including figures	13

1 The First Steps

Let's produce a file with name `first.tex` and content:

```
\documentclass{article}
\begin{document}

Hello world. I claim that  $\sqrt{\tan(\pi/5)} = \cos 2$ . I will provide a proof in a short
while once I've learnt \LaTeX.

\end{document}
```

We know well enough that this is very likely to be rubbish. But `LATEX` doesn't care. So we go ahead and turn this text into something printable.

- Step 1. Type the command `latex first.tex` under UNIX. This will produce a file with name `first.dvi` (and some other files we don't care about at this point).
- Step 2. You can view this file by typing `xdvi first.dvi &`. This allows you to view the “translated” content of our file `first.tex`.
- Step 3. If you are satisfied with the outcome, you can send the file to a printer by typing `dvips first.dvi` which produces the postscript file `first.ps` which can be printed via the UNIX command `lpr -P<printer name> first.ps`. Alternatively if you just want to create the postscript file, and not print it, type `dvips first -o`.

You should receive a piece of paper with something on it that looks like this:

Hello world. I claim that $\sqrt{\tan(\pi/5)} = \cos 2$. I will provide a proof in a short while once I've learnt `LATEX`.

The rest of this pamphlet will do its best to get you started with `LATEX`.

2 Basic L^AT_EX

2.1 The Absolute Minimum

Each L^AT_EX file has to contain at least the following three lines. At the very beginning:

```
\documentclass{*****}
```

```
\begin{document}
```

and at the very end:

```
\end{document}
```

The six asterisks can be replaced by one of the following: **article**, **report**, **book**. It tells L^AT_EX what sort of author you are. If you wish to write an article to be submitted to a journal, you should use **article**. If you're writing a thesis or a final year project report, you should replace the six asterisks with **report** etc. L^AT_EX will try to interpret everything it finds between the `\begin{document}` command and the `\end{document}` command into something printable.

From now on, I will present various small chunks of L^AT_EX that live between these very basic commands.

2.2 Plain Text

At first, we deal with plain text. Remember, even in this simplest of all situations, L^AT_EX will modify the appearance of the text as you typed it into your L^AT_EX document. A few rules:

1. The number of spaces between words is immaterial. A “newline” generated by a Return (or Enter) key is thought of as a space.
2. A blank line signifies the end of a paragraph.
3. The following characters have a special meaning in L^AT_EX:

`\` `&` `$` `%` `~` `_` `{` `}` `#` `^`

When you want one of these characters to appear in the output, most of them can be generated by preceding the character with a backslash.

Let us consider the following sample chunk of a L^AT_EX document:

The special
characters `\&`, `\$`, `\%`, `_`, `\{`, `\}`, and `\#` may be
printed by preceding each with a backslash.

It is likely that 50\% of the time you will be frustrated
because you forgot to precede the % symbol by a backslash.
The \% symbol is used to comment out the rest of the line.

L^AT_EX will produce the following output:

The special characters `&`, `$`, `%`, `_`, `{`, `}`, and `#` may be printed by
preceding each with a backslash.

It is likely that 50% of the time you will be frustrated because
you forgot to precede the (something that makes L^AT_EX to ignore the
whole line after %).

2.3 Type Styles

One can control the shape, series, and family of the type. There are four shapes: upright, *italic*, *slanted*, and SMALL CAPS, and two series: medium, **boldface**, and three families: roman, sans serif, and typewriter.

This text was produced using the following lines:

One can control the shape, series, and family of the type.
There are four shapes: `\textup{upright}`, `\textit{italic}`,
`\textsl{slanted}`, and `\textsc{small caps}`, and
two series: `\textmd{medium}`, `\textbf{boldface}`,
and three families: `\textrm{roman}`, `\textsf{sans serif}`,
and `\texttt{typewriter}`.

Note that text whose type is to be changed is enclosed in curly braces after the command. One can also combine features by staggered curly braces.

The declarations

```
\Huge \huge \LARGE \Large \large \normalize  
\small \footnotesize \scriptsize \tiny
```

can be used to change the type size selectively. These declarations, and the words to which they apply, are enclosed in curly braces to limit their scope. A space separates the command from the text. For example

```
{\LARGE Large text} makes ideal {\Large text} for  
shortsighted people.
```

which produces

Large text makes ideal text for shortsighted people.

2.4 Environments

Environments are portions of the document that we want L^AT_EX to treat differently from the main body. They are generally created by enclosing the text between the commands

```
\begin{environment name}  
...  
\end{environment name}.
```

Some of the common non-mathematical environments are list-making environments such as `itemize` and `enumerate`, text centering via `center`, and the two table-making environments `table` and `tabular`. For example

```
\begin{enumerate}  
  
\item Mr Green  
\item football  
\item England  
  
\end{enumerate}
```

will turn into

1. Mr Green
 2. football
 3. England

2.5 Vertical and Horizontal Spacing

The vertical spacing between lines can be altered using the `\vspace` command. For example, the command

```
\vspace{3.5cm}
```

will leave a vertical space of 3.5cm. Horizontal spacing works in a similar way using the `\hspace` command. For example

Get out your rulers and measure these lengths.

```
\vspace{1.0cm}
```

Push right\hspace{1cm}one centimeter.

This results in:

Get out your rulers and measure these lengths.
--

Push right one centimeter.

3 More Basics

3.1 Titles for Documents

A title page can be generated automatically by specifying the title, authors, affiliations, and date. For example, the title page of this document was generated using the commands

```
\title{\textbf{A quick guide to \LaTeX}}  
\author{Mr Fish}  
\date{\today}  
\maketitle
```

One could also give a specific date such as `\date{18/10/2000}`. The command `\today` automatically puts in the date the document was produced.

A table of contents can automatically included by using the `\tableofcontents` command, namely:

```
\title{\textbf{A quick guide to \LaTeX}}
\author{Mr Fish}
\date{\today}
\maketitle
\tableofcontents
```

3.2 Sectioning Commands

We can regard L^AT_EX documents as organized hierarchically into units such as words, sentences, paragraphs and *sections*. The command

```
\section{Introduction}
```

creates a section whose heading is **Introduction**. Each time it encounters the `\section` command, L^AT_EX starts a new section. The section number is generated automatically. We may also give the section a *key*, for example, `intro`, by using

```
\section{Introduction}\label{intro}
```

We can refer to that section later in the document using the `\ref` command

It was shown in Section~\ref{intro} that

The commands `subsection` and `subsubsection` subdivide the document even further and work in the same way as `\section`.

Let us consider an example.

```
\Section{Introduction}\label{intro}
```

This is an empty intro. We go ahead and jump to the conclusions.

```
\section{Conclusions}
```

As indicated in Section~\ref{intro}, we didn't get any results.

```
\subsection{Acknowledgements}
```

We thank the department for its hospitality during the preparation of this pointless paper.

We obtain something like

1 Introduction

This is an empty intro. We go ahead and jump to the conclusions.

2 Conclusions

As indicated in Section 1, we didn't get any results.

2.1 Acknowledgements

We thank the department for its hospitality during the preparation of this pointless paper.

3.3 Bibliography

A bibliography can be created with the `thebibliography` environment. This is similar to the list-making environment discussed before. The command `\bibitem`, whose argument is enclosed in curly braces, precedes each entry. The argument specifies the key by which the entry can be cited, anywhere in the document, using the `\cite` command.

A bibliography is, for example, created by using

```
\begin{thebibliography}{99}
```

```
\bibitem{1a}
```

```
Bill Bryson, \LaTeX\ is fun, Penguin, London, 2001.
```

```
\bibitem{1b}
```

```
Claus Davidson, \LaTeX\ is no fun, Knopf, New York, 1564.
```

```
\end{thebibliography}
```

You should place the environment `\thebibliography` at the end of your document but before `\end{document}`. You can now cite both books in the text. For example

The two most important sources on `\LaTeX\` are `\cite{1a,1b}`.

3.4 Troubleshooting

When you run the `latex` command under UNIX, you might receive error messages and warnings. These are often not very helpful. Just try to get the `latex` command to finish its job by either pressing Return as many times as necessary or, in the worst case, by doing a `^C` (control C). Then try the following:

- If a `dvi` file has been produced, then preview the document. This may make the error apparent.
- Comment out suspicious looking lines by putting a `%` at the beginning of them.
- Insert the `\end{document}` command before (or about) where the error might have occurred. Run `latex` on this abbreviated document and see if it works. Use this strategy to pinpoint the error.
- In cases of real desperation, have a look at a `LATEX` manual and consult its error section.

4 Typesetting Mathematics

4.1 Examples

Mathematical symbols that appear within the text are enclosed between `$` signs. For example, $x = \sin y$ is created by

For example, `$x = \sin y$` is created by

`LATEX` provides special expressions for many functions and expressions such as `\sin`. A list is provided at the end of this document.

4.2 Equation Environments

Equations that are to be displayed on a separate line should be inclosed in `\[...\]`. In order to get a numbered expression, we must use the `equation` environment, which is contained in `\begin{equation}...\end{equation}`. If we include a labeling command, as in `\label{fermat}`, we can refer to the equation by its key `\ref{fermat}` rather than its number. If equations with more than one line have to be displayed, then a more elaborate environment is to be used. This is provided by `\begin{eqnarray}...\end{eqnarray}`.

Given the inequality

$$2 > 1$$

we can conclude that

$$2^2 > 1^2 .$$

The statement (`\ref{eqarr}`) can be generalized to a sequence of inequalities such as

$$\begin{array}{l} 2^1 &>& 1^1 \\ 2^2 &>& 1^2 \\ 2^3 &>& 1^3 \end{array}$$

etc.

which yields:

<p>Given the inequality</p> $2 > 1$ <p>we can conclude that</p> $2^2 > 1^2. \tag{1}$ <p>The statement (1) can be generalized to a sequence of inequalities such as</p> $\begin{array}{l} 2^1 &>& 1^1 \\ 2^2 &>& 1^2 \\ 2^3 &>& 1^3 \end{array}$ <p>etc.</p>

If you don't like to have all equations numbered, you can use the `\begin{eqnarray*}.....\end{eqnarray*}` environment. A `\\` indicates that a new line starts, while `&>&` makes sure that everything is centered about the inequality sign `>`.

4.3 Fonts and Indices

We continue with examples involving fractions, subscripts, and superscripts. The command `\frac` for forming fractions is always followed by two expressions that are enclosed in curly brackets (the numerator and denominator). The characters `_` and `^` produce subscripts and superscripts, respectively. Here is an example

The equation

```
\[
x^2 = \frac{\sqrt{2} + x^{3x}}{\sin x}
\]
```

is clearly a tough one.

as you can see more clearly from

The equation

$$x^2 = \frac{\sqrt{2} + x^{3x}}{\sin x}$$

is clearly a tough one.

You can also include Greek letters as well as calligraphic and many other fonts.

Here is a short list of such: `$_gamma, \Gamma, \mathcal{H}, \mathbf{H}, \mathit{H}$`.

or

Here is a short list of such: $\gamma, \Gamma, \mathcal{H}, \mathbf{H}, H$.

5 Introduction to tables

This is best tackled through an example. The following

```
\begin{tabular}{lcr}
Name & Function & Range\\
\hline
Sine &  $\sin x$  &  $[-1, 1]$ \\
Gaussian &  $\exp -x^2$  &  $(0, 1]$ \\
Unity & 1 &  $[1]$ \\
\hline
\end{tabular}
```

gives

Name	Function	Range
Sine	$\sin x$	$[-1, 1]$
Gaussian	$\exp -x^2$	$(0, 1]$
Unity	1	$[1]$

Notes:

1. `{lcr}` says left-justify the first column, centre the second and right-justify the third.
2. `&` separates the columns, `\\` ends the lines.
3. `\hline` puts a horizontal line right across the table.
4. To get vertical lines right down the table, change `{lcr}` into `{|l|c|r|}` — try it and see.
5. To centre the whole table, enclose the `tabular` environment within `\begin{center}... \end{center}`.

6 Including figures

You may want to include include `eps` (encapsulated PostScript) figures in your document. In order to do this, you will need `\usepackage[dvips]{graphics}` straight after the `\documentclass` line (i.e. the second line in your file). This loads a package capable of dealing with figures for you. Suppose a file called `figure.eps` contains your `eps` figure, then

```

\begin{figure}[htbp]
\begin{center}
\scalebox{0.6}{\includegraphics{figure.eps}}
\caption{A simple figure.}
\label{fig1}
\end{center}
\end{figure}

```

will include the figure, scale, caption and label it. The options [htbp] stand for ‘here’, ‘top of current page’, ‘bottom of current page’ and ‘on a separate page’ respectively. L^AT_EX will try to do the first one first; if it can’t, it will try the second and so on.

With the above code, this is what you get:

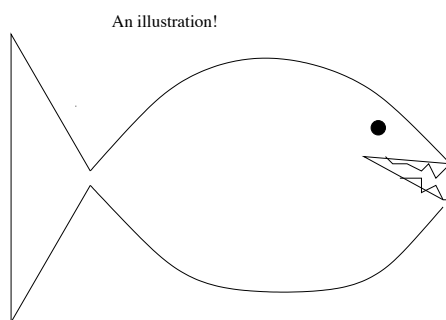


Figure 1: A work of art: Untitled, 2004.

The End

That’s about it; good luck!