Introduction to LaTeX

M.Sc. in Mathematical Modelling & Scientific Computing, Additional Skills

22nd October 2026

What will LaTeX do for you?

Allow you to write mathematical reports which look well presented. You can easily include:

- mathematical equations (with or without numbers);
- tables;
- pictures, graphs and other figures;
- references;
- a table of contents.

But it is *not* a WYSIWYG environment.

References and webpages

- Overleaf — an easy way to use LaTeX, https://www.overleaf.com/
- A good starting point is "The Not so Short Introduction to LaTeX 2ε" found at https://tobi.oetiker.ch/lshort/lshort.pdf
- LaTeX. A document Preparation System. Leslie Lamport. Addison-Wesley, 1994.
- The LaTeX Companion. Michel Goossens, Frank Mittelbach and Alexander Samarin. Addison-Wesley, 1994 (or 2004 for second edition).
- Lots of information about LaTeX including the OCIAM thesis class is available at http://www.maths.ox.ac.uk/members/it/faqs/latex
- For extra packages try searching at http://www.ctan.org/pkg
- For a Q&A site see TeX–LaTeX Stack Exchange https://tex.stackexchange.com/

Getting started

Let's start a project called `Example 2026` in Overleaf. Such a file starts with a command of the form

```
\documentclass{...}
```

where the main document classes are article, report, book and slides. article should be sufficient for a special topic, you might want to use report or the ociamthesis class for a dissertation. We will begin with a simple article so we begin with the command

```
\documentclass{article}
```

Title, author etc

We can set up the title, author and date using commands of the form

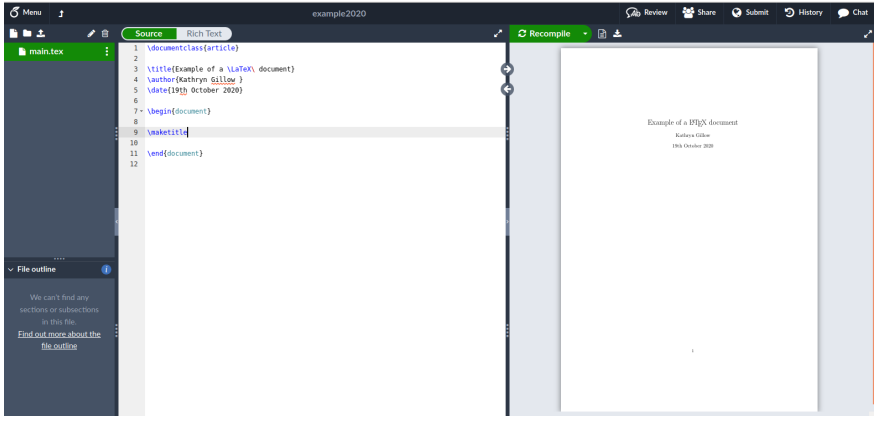
```
\title{Example of a \LaTeX\ document}
\author{Kathryn Gillow}
\date{22nd October 2026}
```

Then we type into our file

```
\begin{document}
\maketitle
\end{document}
```

Compiling and viewing the document

We run pdflatex from within Overleaf (use “Recompile’) and automatically view the pdf.



Text

Next we want to type lots of text (most likely an introduction to the maths we're about to do). So we just type lots of text into our file. Things to remember are:

- it doesn't matter how many spaces you put between words (as long as there is at least one!);
- it doesn't matter where you break the lines;
- anything which comes after a completely empty line starts a new paragraph.

Sectioning

You probably want to split up your report into sections to make it more readable. To do this you can use commands of the form

```
\section{Introduction}
\subsection{Finite Difference Methods}
\subsubsection{Finite Difference Methods for the Heat
Equation}
```

If you want to add a table of contents then you can type

```
\tableofcontents
```

after the `\maketitle` command.

Changing the layout of the page

The default font size is 10pt with single spaced text. However, all submitted reports for MMSC require a minimum font size of 12pt with well-spaced text.

```
\documentclass[12pt]{article}
```

To adjust the spacing of the text we adjust the "baselineskip" by adding a line of the form

```
\baselineskip=20pt plus1pt
```

just below the `\begin{document}` command.

Changing the layout of the page

We see that with a 12pt font and "well-spaced" text we don't actually get much on the page. There is a huge gap at both the top and the bottom and the margins are quite large. However, it is fairly easy to adjust the margins, we can simply add the following lines after the `\documentclass` command

```
\setlength{\topmargin}{0.0in}
\setlength{\oddsidemargin}{0.33in}
\setlength{\textheight}{9.0in}
\setlength{\textwidth}{6.0in}
```

Symbols

Most symbols will just appear in your LaTeX if you just type them in, e.g. ( ) : [ ] . !

However, some symbols, most notably

$ & % # _ { } ~ ^ \

do not simply print, they make LaTeX do something different instead. Some of these can be used by adding a backslash as a prefix, e.g.

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

The others can be printed in mathematical equations or as accents using special commands.

Note: anything following a typed % symbol is interpreted as a comment so it's really important to put the backslash in front if you want a % (and anything following it) to appear.

Maths

Using the package `amssymb` will give you, by default, access to all the standard mathematical symbols. Your LaTeX file can access this package if you use the command

```
\usepackage{amssymb}
```

at the beginning of the file, just after the `\documentclass` command.

Inline maths

You can put mathematics in with your text by putting it between two $ symbols. Greek letters are printed by typing their name with a \ in front, superscripts are subscripts are produced by putting them between { and } following either _ or ^.

So for example

```
The equation $\alpha_{1} = \beta^{2}$ tells us
```

produces

The equation $\alpha_1 = \beta^2$ tells us

Equations

Often you don't want your maths to be in your text as it makes it harder to read so you want to have an equation on a separate line. You can achieve this either by putting it between `\[` and `\]` or between `\begin{displaymath}` and `\end{displaymath}` commands. So either of

```
\[
\alpha_{1} = \beta^{2}
\]
```

or

```
\begin{displaymath}
\alpha_{1} = \beta^{2}
\end{displaymath}
```

produces

$$\alpha_1 = \beta^2$$

Numbered equations

The `displaymath` and `\[, \]` environments do not produce numbered equations which can sometimes be useful. To get a single numbered equation we can use the `equation` environment instead. For example

```
\begin{equation}
\alpha_{1} = \beta^{2}
\end{equation}
```

produces

$$\alpha_1 = \beta^2 \tag{1}$$

Referring to numbered equations

We might then want to refer back to this numbered equation and we can do this by giving it a label. The text

```
\begin{equation}
\label{alpha_beta_eqn}
\alpha_{1} = \beta^{2}
\end{equation}
Equation~(\ref{alpha_beta_eqn}) explains the relationship
between $\alpha_1$ and $\beta$.
```

produces

$$\alpha_1 = \beta^2 \tag{2}$$

Equation (2) explains the relationship between α_1 and β.

Referring to numbered equations cont'd

An alternative command for referring to equations is `eqref` which requires the `amsmath` package. We add the command

```
\usepackage{amsmath}
```

just after the `documentclass` command and then we can refer to the equation using

```
Equation~\eqref{alpha_beta_eqn} explains the relationship
between $\alpha_1$ and $\beta$.
```

which produces

Equation (2) explains the relationship between α_1 and β.

The `eqref` command has the advantage of putting parentheses around the equation number for us.

Referring to numbered equations cont'd

We can use the cleveref package to reference more easily, for example

```
\Cref{alpha_beta_eqn} explains the relationship
between $\alpha_1$ and $\beta$.
```

which produces

Equation (2) explains the relationship between α_1 and β.

Using the package hyperref also allows for clickable links. You need to add this before the cleveref package.

Multiple equations

Often you want to produce a list of several equations neatly lined up. This can be achieved using the `eqnarray` environment, for example

```
\begin{eqnarray}
\beta & = & 1 \\
\alpha_1 & = & \beta^2 \\
\alpha_2 & = & \beta^3
\end{eqnarray}
```

produces

$$\beta = 1 \tag{3}$$
$$\alpha_1 = \beta^2 \tag{4}$$
$$\alpha_2 = \beta^3 \tag{5}$$

The `align` environment works similarly.

Multiple equations cont'd

Basically this produces an array with three columns of which

- the left-hand column is right-justified
- the middle column is centered
- the right-hand column is left-justified

The start of a new column on every line is marked with an `&`. A new line is preceded by the `\\` symbol.

By default all the equations are numbered. To avoid numbering a particular equation use `\nonumber`. To avoid numbering all equations use `\begin{eqnarray*}` and `\end{eqnarray*}` instead.

More maths

How do we write integrals, sums, fractions, partial derivatives, trigonometric functions etc?

- `\int_{1}^{\infty} x^{-1} {\rm d} x`
- `\sum_{n=0}^{s} n`
- `\frac{3}{2x}`
- `\frac{\partial u}{\partial x}`
- `\sin \cos \tan \log \exp \max`

$$\int_1^{\infty} x^{-1}\mathrm{d}x,\ \sum_{n=0}^{s} n,\ \frac{3}{2x},\ \frac{\partial u}{\partial x},\ \sin,\ \cos,\ \tan,\ \log,\ \exp,\ \max$$

Arrays (vectors and matrices etc)

To produce a matrix (think of a vector as a matrix with only one column) you can use the array environment, e.g.

```
\begin{eqnarray}
A & = & \left(
\begin{array}{c c c}
1 & 2 & 3 \\
4 & 5 & 6
\end{array}
\right)
\end{eqnarray}
```

gives

$$A = \left(\begin{array}{c c c} 1 & 2 & 3 \\ 4 & 5 & 6 \end{array} \right) \tag{6}$$

Arrays (vectors and matrices etc)

Alternatively you can use the `pmatrix` environment, e.g.

```
\begin{eqnarray}
A & = &
\begin{pmatrix}
1 & 2 & 3 \\
4 & 5 & 6
\end{pmatrix}
\end{eqnarray}
```

gives

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \tag{7}$$

More arrays

Alternatively try commands of the form

```
\begin{eqnarray*}
|x| & = & \left\{
\begin{array}{r l}
-x & x<0 \\
x & x \ge 0
\end{array}
\right.
\end{eqnarray*}
```

to get

$$|x| = \begin{cases} -x & x < 0 \\ x & x \geq 0 \end{cases}$$

Tables

```
\begin{table}[ht]
\centering
\begin{tabular}{c c c c }
\hline
mesh & triangles & nodes & current \\
\hline
1 & 32 & 25 & 1.270 \\
2 & 94 & 59 & 1.131 \\
3 & 201 & 116 & 1.066 \\
4 & 372 & 208 & 1.034 \\
5 & 527 & 288 & 1.019 \\
\hline
\end{tabular}
\caption{Triangulations produced by FEM.}
\label{disctab}
\end{table}
In Table \ref{disctab} (or \Cref{disctab}) we see some numbers.
```

Tables cont'd

mesh	triangles	nodes	current
1	32	25	1.270
2	94	59	1.131
3	201	116	1.066
4	372	208	1.034
5	527	288	1.019

Table: Triangulations produced by FEM.

In Table 1 (or Table 1) we see some numbers.

Tables cont'd

To add vertical lines in the table use a command of the form

```
\begin{tabular}{|c |c |c |c |}
```

It is also possible to add thick lines with a command of the form

```
\Xhline{5\arrayrulewidth}
```

although you will need to use the package `makecell` in order to use this type of command.

Figures

To insert an eps, pdf or jpeg file if you are using Overleaf you will need to upload the figure file using the upload button at the top left. Then you can use the package `epsfig` and commands of the form

```
\begin{figure}[ht]
\centering
\includegraphics[width=6cm]{sphere}
\caption{A sphere.}
\label{spherefig}
\end{figure}

In Figure \ref{spherefig} we see a sphere.
```

Figures cont'd

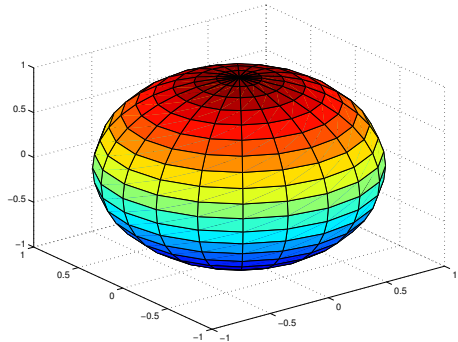


Figure: A sphere.

In Figure 1 we see a sphere.

Theorems

Using the package `amsthm` it is possible to add theorems (and proofs) to your LaTeX document.

To do this add a line of the form

```
\newtheorem*{thm_pic}{Theorem: Picard}
```

Then the statement of the theorem and the proof should go between commands of the form

```
\begin{thm_pic}
\end{thm_pic}
\begin{proof}
\end{proof}
```

Theorems cont'd

This gives something of the form:

Theorem: Picard
state the theorem here

Proof.
prove the theorem here □

Theorems cont'd

For named and numbered theorems use

```
\newtheorem{thm_pic}{Theorem}
\begin{thm_pic}[Picard]
state the theorem here
\end{thm_pic}
```

to get

Theorem (Picard)

state the theorem here

Inserting code

The package `listings` allows you to insert code into your LaTeX document. Although it is not necessary to include your code in special topics or case studies, you may sometimes like to add code in an appendix. After the `\usepackage{...}` command you can set the coding language you will use, as well as other options, e.g.

```
\lstset{language=Python,frame=single}
```

and this will format the code nicely and put a box round it. To insert the code you just need a command of the form

```
\lstinputlisting{mynewt.py}
```

Citations

Either use `thebibliography` environment *or* BibTeX.

- BibTeX has the advantage that you maintain a file containing all the references you ever use so you only need to type them in once.
- There are lots of different style files for use with BibTeX so you can easily format all your references in one way. With `thebibliography` environment you have to do the formatting.
- The different style files also allow you to order your references in different ways. With `thebibliography` environment they appear in the order in which they are typed.

Whichever method is used, a citation command takes the form

```
For an introduction to finite difference methods see \cite{label},
or, more specifically, \cite[Chapter 2]{label}.
```

Citations using thebibliography

Add the references section to the table of contents with a command of the form

```
\addcontentsline{toc}{section}{References}
```

Entries in the bibliography are normally entered just before the \end{document} command and take the form

```
\begin{thebibliography}{99}
\bibitem{morton} K.W. Morton and D.F. Mayers. Numerical Solution of
Partial Differential Equations. Cambridge University Press, 1994.
\bibitem{houston} P. Houston, J. Mackenzie, E. S\"{u}li and G. Warnecke.
A posteriori error analysis for numerical approximations of Friedrichs
systems. \textit{Numerische Mathematik}, \textbf{82}:409--432, 1999.
\end{thebibliography}
```

Other things

- List of figures
- Appendices
- Lists (numbered using `enumerate` or bullet points using `itemize`)
- Other type faces (bold, italic ...)
- Lots more mathematical symbols

Take a look at "The Not So Short Introduction to LaTeX 2_ε".

Other things to remember

- Use a spellchecker.
- Read what you have written to check it makes sense.
- Punctuate equations — put a full stop at the end if it represents the end of a sentence.
- When referring to specific equations, sections, tables, figures etc. use a capital letter for the word “equation” etc. E.g. “In Section 4.1 we see that . . . “