An Interactive Introduction to LaTeX

Part 1: The Basics

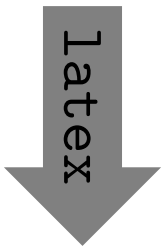
Why LaTeX?

- It makes beautiful documents
  - Especially mathematics
- It was created by scientists, for scientists
  - A large and active community
- It is powerful — you can extend it
  - Packages for papers, presentations, spreadsheets, ...

How does it work?

- You write your document in `plain text` with `commands` that describe its structure and meaning.
- The `latex` program processes your text and commands to produce a beautifully formatted document.

```
The rain in Spain falls \emph{mainly} on the plain.
```



The rain in Spain falls *mainly* on the plain.

More examples of commands and their output. . .

```
\begin{itemize}
\item Tea
\item Milk
\item Biscuits
\end{itemize}
```

- Tea
- Milk
- Biscuits

```
\begin{figure}
\includegraphics{gerbil}
\end{figure}
```

```
\begin{equation}
\alpha + \beta + 1
\end{equation}
```

$$\alpha + \beta + 1 \qquad (1)$$

Image license: CC0

Attitude adjustment

- Use commands to describe ‘what it is’, not ‘how it looks’.
- Focus on your content.
- Let LaTeX do its job.

Getting started

- A minimal LaTeX document:

```
\documentclass{article}
\begin{document}
Hello World! % your content goes here...
\end{document}
```

- Commands start with a *backslash* \ .
- Every document starts with a `\documentclass` command.
- The *argument* in curly braces { } tells LaTeX what kind of document we are creating: an `article`.
- A percent sign % starts a *comment* — LaTeX will ignore the rest of the line.

Getting started with **Overleaf**

- Overleaf is a website for writing documents in LaTeX.
- It 'compiles' your LaTeX automatically to show you the results.

- As we go through the following slides, try out the examples by typing them into the example document on Overleaf.
- **No really, you should try them out as we go!**

Typesetting Text

- Type your text between `\begin{document}` and `\end{document}`.
- For the most part, you can just type your text normally.

```
Words are separated by one or more
spaces.

Paragraphs are separated by one
or more blank lines.
```

Words are separated by one or more spaces.

Paragraphs are separated by one or more blank lines.

- Space in the source file is collapsed in the output.

```
The   rain       in Spain
falls mainly on the plain.
```

The rain in Spain falls mainly on the plain.

Typesetting Text: Caveats

- Quotation marks are a bit tricky:
  use a backtick  on the left and an apostrophe  on the right.

`Single quotes: `text'.`	Single quotes: ‘text’.
`Double quotes: ``text''.`	Double quotes: “text”.

- Some common characters have special meanings in LaTeX:

   percent sign
   hash (pound / sharp) sign
   ampersand
   dollar sign

- If you just type these, you'll get an error. If you want one to appear in the output, you have to *escape* it by preceding it with a backslash.

`\$\%\&\#!`	$%&#!

Handling Errors

- LaTeX can get confused when it is trying to compile your document. If it does, it stops with an error, which you must fix before it will produce any output.
- For example, if you misspell `\emph` as `\meph`, LaTeX will stop with an "undefined control sequence" error, because "meph" is not one of the commands it knows.

Advice on Errors

1. Don't panic! Errors happen.
2. Fix them as soon as they arise — if what you just typed caused an error, you can start your debugging there.
3. If there are multiple errors, start with the first one — the cause may even be above it.

Typesetting Mathematics: Dollar Signs

- Why are dollar signs  special? We use them to mark mathematics in text.

```
% not so good:
Let a and b be distinct positive
integers, and let c = a - b + 1.

% much better:
Let $a$ and $b$ be distinct positive
integers, and let $c = a - b + 1$.
```

Let a and b be distinct positive integers, and let c = a - b + 1.

Let a and b be distinct positive integers, and let $c = a - b + 1$.

- Always use dollar signs in pairs — one to begin the mathematics, and one to end it.
- LaTeX handles spacing automatically; it ignores your spaces.

```
Let $y=mx+b$ be \ldots

Let $y = m x + b$ be \ldots
```

Let $y = mx + b$ be ...

Let $y = mx + b$ be ...

Typesetting Mathematics: Notation

- Use caret ^ for superscripts and underscore _ for subscripts.

```
$y = c_2 x^2 + c_1 x + c_0$
```

$y = c_2 x^2 + c_1 x + c_0$

- Use curly braces { } to group superscripts and subscripts.

```
$F_n = F_n-1 + F_n-2$      % oops!

$F_n = F_{n-1} + F_{n-2}$ % ok!
```

$F_n = F_n - 1 + F_n - 2$

$F_n = F_{n-1} + F_{n-2}$

- There are commands for Greek letters and common notation.

```
$\mu = A e^{Q/RT}$

$\Omega = \sum_{k=1}^{n} \omega_k$
```

$\mu = A e^{Q/RT}$

$\Omega = \sum_{k=1}^{n} \omega_k$

Typesetting Mathematics: Displayed Equations

- If it's big and scary, *display* it on its own line using `\begin{equation}` and `\end{equation}`.

```
The roots of a quadratic equation
are given by
\begin{equation}
x = \frac{-b \pm \sqrt{b^2 - 4ac}}
         {2a}
\end{equation}
where $a$, $b$ and $c$ are \ldots
```

The roots of a quadratic equation are given by

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (2)$$

where a, b and c are . . .

Caution: LaTeX mostly ignores your spaces in mathematics, but it can't handle blank lines in equations — don't put blank lines in your mathematics.

Interlude: Environments

- `equation` is an *environment* — a context.
- A command can produce different output in different contexts.

```
We can write
$ \Omega = \sum_{k=1}^{n} \omega_k $
in text, or we can write
\begin{equation}
  \Omega = \sum_{k=1}^{n} \omega_k
\end{equation}
to display it.
```

We can write $\Omega = \sum_{k=1}^{n} \omega_k$ in text, or we can write

$$\Omega = \sum_{k=1}^{n} \omega_k \tag{3}$$

to display it.

- Note how the Σ is bigger in the `equation` environment, and how the subscripts and superscripts change position, even though we used the same commands.

In fact, we could have written `$...$` as `\begin{math}...\end{math}`.

Interlude: Environments

- The `\begin` and `\end` commands are used to create many different environments.
- The `itemize` and `enumerate` environments generate lists.

```
\begin{itemize} % for bullet points
\item Biscuits
\item Tea
\end{itemize}

\begin{enumerate} % for numbers
\item Biscuits
\item Tea
\end{enumerate}
```

- Biscuits
- Tea

1. Biscuits
2. Tea

Interlude: Packages

- All of the commands and environments we've used so far are built into LaTeX.
- *Packages* are libraries of extra commands and environments. There are thousands of freely available packages.
- We have to load each of the packages we want to use with a `\usepackage` command in the *preamble*.
- Example: `amsmath` from the American Mathematical Society.

```
\documentclass{article}
\usepackage{amsmath} % preamble
\begin{document}
% now we can use commands from amsmath here...
\end{document}
```

Typesetting Mathematics: Examples with amsmath

- Use `equation*` ("equation-star") for unnumbered equations.

```
\begin{equation*}
  \Omega = \sum_{k=1}^{n} \omega_k
\end{equation*}
```

$$\Omega = \sum_{k=1}^{n} \omega_k$$

- LaTeX treats adjacent letters as variables multiplied together, which is not always what you want. amsmath defines commands for many common mathematical operators.

```
\begin{equation*} % bad!
 min_{x,y} (1-x)^2 + 100(y-x^2)^2
\end{equation*}
\begin{equation*} % good!
\min_{x,y}{(1-x)^2 + 100(y-x^2)^2}
\end{equation*}
```

$$min_{x,y}(1-x)^2+100(y-x^2)^2$$

$$\min_{x,y}(1-x)^2+100(y-x^2)^2$$

- You can use `\operatorname` for others.

```
\begin{equation*}
\beta_i =
\frac{\operatorname{Cov}(R_i, R_m)}
     {\operatorname{Var}(R_m)}
\end{equation*}
```

$$\beta_i = \frac{\operatorname{Cov}(R_i, R_m)}{\operatorname{Var}(R_m)}$$

Typesetting Mathematics: Examples with amsmath

- Align a sequence of equations at the equals sign

$$\begin{aligned}(x+1)^3 &= (x+1)(x+1)(x+1)\\ &= (x+1)(x^2+2x+1)\\ &= x^3+3x^2+3x+1\end{aligned}$$

with the `align*` environment.

```
\begin{align*}
(x+1)^3 &= (x+1)(x+1)(x+1) \\
        &= (x+1)(x^2 + 2x + 1) \\
        &= x^3 + 3x^2 + 3x + 1
\end{align*}
```

- An ampersand `&` separates the left column (before the $=$) from the right column (after the $=$).
- A double backslash `\\` starts a new line.

End of Part 1

- Congrats! You've already learned how to ...
  - Typeset text in LaTeX.
  - Use lots of different commands.
  - Handle errors when they arise.
  - Typeset some beautiful mathematics.
  - Use several different environments.
  - Load packages.
- That's amazing!
- In Part 2, we'll see how to use LaTeX to write structured documents with sections, cross references, figures, tables and bibliographies. See you then!