

L^AT_EX for CDT chemists

Richard Cox

TECS CDT

2023

Why use L^AT_EX?

From overleaf:

Why use \LaTeX ?

From overleaf:

- ▶ support for typesetting extremely complex mathematics, tables and technical content for the physical sciences

Why use \LaTeX ?

From overleaf:

- ▶ support for typesetting extremely complex mathematics, tables and technical content for the physical sciences
- ▶ facilities for footnotes, cross-referencing and management of bibliographies

Why use L^AT_EX?

From overleaf:

- ▶ support for typesetting extremely complex mathematics, tables and technical content for the physical sciences
- ▶ facilities for footnotes, cross-referencing and management of bibliographies
- ▶ ease of producing complicated, or tedious, document elements such as indexes, glossaries, table of contents, lists of figures

Why use L^AT_EX?

From overleaf:

- ▶ support for typesetting extremely complex mathematics, tables and technical content for the physical sciences
- ▶ facilities for footnotes, cross-referencing and management of bibliographies
- ▶ ease of producing complicated, or tedious, document elements such as indexes, glossaries, table of contents, lists of figures
- ▶ being highly customizable for bespoke document production due to its intrinsic programmability and extensibility through thousands of free add-on packages

Why use L^AT_EX?

From overleaf:

- ▶ support for typesetting extremely complex mathematics, tables and technical content for the physical sciences
- ▶ facilities for footnotes, cross-referencing and management of bibliographies
- ▶ ease of producing complicated, or tedious, document elements such as indexes, glossaries, table of contents, lists of figures
- ▶ being highly customizable for bespoke document production due to its intrinsic programmability and extensibility through thousands of free add-on packages
- ▶ great deal of control over the production of documents which are typeset to extremely high standards

Why use L^AT_EX?

From overleaf:

- ▶ support for typesetting extremely complex mathematics, tables and technical content for the physical sciences
- ▶ facilities for footnotes, cross-referencing and management of bibliographies
- ▶ ease of producing complicated, or tedious, document elements such as indexes, glossaries, table of contents, lists of figures
- ▶ being highly customizable for bespoke document production due to its intrinsic programmability and extensibility through thousands of free add-on packages
- ▶ great deal of control over the production of documents which are typeset to extremely high standards
- ▶ separation of document content from document style

Writing \LaTeX

Unlike other document-producing programs, \LaTeX is written in a plain text (.txt) file – therefore, it is free and its compilers are open-source!

Writing L^AT_EX

Unlike other document-producing programs, L^AT_EX is written in a plain text (.txt) file – therefore, it is free and its compilers are open-source!

Choose a good code editor. You can use:

- ▶ Notepad – any editor will do, right?

Writing L^AT_EX

Unlike other document-producing programs, L^AT_EX is written in a plain text (.txt) file – therefore, it is free and its compilers are open-source!

Choose a good code editor. You can use:

- ▶ Notepad – any editor will do, right?
- ▶ T_EXmaker, T_EXstudio or T_EXworks – L^AT_EX-specialised code editors

Writing L^AT_EX

Unlike other document-producing programs, L^AT_EX is written in a plain text (.txt) file – therefore, it is free and its compilers are open-source!

Choose a good code editor. You can use:

- ▶ Notepad – any editor will do, right?
- ▶ T_EXmaker, T_EXstudio or T_EXworks – L^AT_EX-specialised code editors
- ▶ Visual studio code – all-purpose, lightweight and highly customisable code editor – **recommended for advanced users**

Writing L^AT_EX

Unlike other document-producing programs, L^AT_EX is written in a plain text (.txt) file – therefore, it is free and its compilers are open-source!

Choose a good code editor. You can use:

- ▶ Notepad – any editor will do, right?
- ▶ T_EXmaker, T_EXstudio or T_EXworks – L^AT_EX-specialised code editors
- ▶ Visual studio code – all-purpose, lightweight and highly customisable code editor – **recommended for advanced users**
- ▶ `overleaf.com` – if VSCode is MS word, then overleaf is google docs. All your work is saved to the cloud – **recommended for beginners**

Setting up a \LaTeX document

```
1 \documentclass{article}
2
3 \begin{document}
4     Hello world!
5
6 \end{document}
```

Setting up a \LaTeX document

```
1 \documentclass[a4paper,12pt]{article}
2 \title{LaTeX test}
3 \author{Richard Cox}
4 \date{}
5
6 \begin{document}
7     \maketitle
8
9     \section{Introduction}
10         Hello world!
11
12     \subsection{}
13
14         Some text in a subsection
15
16 \end{document}
17
```

Setting up a L^AT_EX document

```
1 \documentclass[a4paper,12pt,twocolumn]{article}
2 \title{LaTeX test}
3 \author{Richard Cox}
4 \date{}
5
6 \usepackage[left=20mm, right=20mm, top=20mm, bottom=20
   mm]
7 \usepackage{lipsum}
8
9 \begin{document}
10     \maketitle
11
12     \section{Introduction}
13         \lipsum{1}
14
15         \subsection{A subsection}
16
17         \lipsum{1-3}
18
19 \end{document}
20
```


The preamble

Works similarly to a python script:

```
1 import seaborn as sns
2 import pandas as pd
3
4 tips = pd.read_csv("https://milliams.com/courses/
    data_analysis_python/tips.csv")
5 sns.catplot(data=tips, x="day", y="bill_per_person",
    order=["Thur", "Fri", "Sat", "Sun"], kind="box")
```

Think of this as where you do general page set-up, creating commands, etc.

Must have the packages you want to invoke **BEFORE** the code

What's a better place to put it than before `\begin{document}`?

Using packages

- ▶ Similarly to python libraries, packages are snippets of code made by other people that do things
- ▶ Use the command `\usepackage[]{}` to load a package
- ▶ `{}` lets you specify the argument; in this case, the package
- ▶ `[]` lets you include options – detailed in the *package documentation*
- ▶ adding `*` to most commands means they won't be numbered or won't appear in the table of contents

Using packages

- ▶ Similarly to python libraries, packages are snippets of code made by other people that do things
- ▶ Use the command `\usepackage[]{}` to load a package
- ▶ `{}` lets you specify the argument; in this case, the package
- ▶ `[]` lets you include options – detailed in the *package documentation*
- ▶ adding `*` to most commands means they won't be numbered or won't appear in the table of contents

Example:

```
1 \usepackage{mhchem} % Easy chemistry formatting
2 \usepackage{graphicx} % Including images in LaTeX
3 \usepackage[margin=1in]{geometry} % Better page sizing
4
```

The main body

All displayed text is written in an environment:

```
1 \begin{env}  
2  
3     Some text  
4     \some-commands  
5  
6 \end{env}  
7
```

The visual part of the whole document is in the document environment, between `\begin{document}` and `\end{document}`

Useful text commands

- ▶ **Bold text:** `\textbf{}`
- ▶ *Italicised text:* `\textit{}`
- ▶ Underlined text: `\underline{}`
- ▶ Commented text: Some commented text
- ▶ Superscript text: `\textsuperscript{}`
- ▶ Subscript text: `\textsubscript{}`

Useful text commands

- ▶ **Bold text:** `\textbf{}`
- ▶ *Italicised text:* `\textit{}`
- ▶ Underlined text: `\underline{}`
- ▶ Commented text: Some commented text
- ▶ Superscript text: `\textsuperscript{}`
- ▶ Subscript text: `\textsubscript{}`
- ▶ Adding a label to a figure, table, scheme or equation:
`\label{}`
- ▶ Cross-referencing a figure, table, scheme or equation: `\ref{}`

Useful text commands

- ▶ **Bold text:** `\textbf{}`
- ▶ *Italicised text:* `\textit{}`
- ▶ Underlined text: `\underline{}`
- ▶ Commented text: Some commented text
- ▶ Superscript text: `\textsuperscript{}`
- ▶ Subscript text: `\textsubscript{}`
- ▶ Adding a label to a figure, table, scheme or equation:
`\label{}`
- ▶ Cross-referencing a figure, table, scheme or equation: `\ref{}`
- ▶ Linebreak: `\\` or `\linebreak`
- ▶ pagebreak: `\pagebreak`
- ▶ Including other .tex files: `\include{}`

Useful text commands

- ▶ **Bold text:** `\textbf{}`
- ▶ *Italicised text:* `\textit{}`
- ▶ Underlined text: `\underline{}`
- ▶ Commented text: Some commented text
- ▶ Superscript text: `\textsuperscript{}`
- ▶ Subscript text: `\textsubscript{}`
- ▶ Adding a label to a figure, table, scheme or equation:
`\label{}`
- ▶ Cross-referencing a figure, table, scheme or equation: `\ref{}`
- ▶ Linebreak: `\\` or `\linebreak`
- ▶ pagebreak: `\pagebreak`
- ▶ Including other .tex files: `\include{}`
- ▶ Adding a table of contents: `\tableofcontents`
- ▶ Adding to a table of content:
`\addcontentsline{toc}{section}`

Maths in L^AT_EX

- ▶ For an inline maths environment, use `$some maths$`
- ▶ For a displayed maths environment, use `\begin{math}` and `\end{math}`
- ▶ For equations, use `\begin{equation}` and `\end{equation}` (n.b. equations are numbered automatically like figures and tables!)

Maths in L^AT_EX

- ▶ For an inline maths environment, use `$some maths$`
- ▶ For a displayed maths environment, use `\begin{math}` and `\end{math}`
- ▶ For equations, use `\begin{equation}` and `\end{equation}` (n.b. equations are numbered automatically like figures and tables!)

Inside maths environments, treat it like an inline maths calculator:

- ▶ For subscript, use `a_{n}` = a_n
- ▶ For superscript, use `a^{n}` = a^n
- ▶ For fractions, use `\frac{numerator}{denominator}`
- ▶ For greek letters, use `\greek-letter` (e.g. `α` = α)

Making tables

Use the table and tabular environments:

```
1 \begin{table}[float]
2   \begin{tabular}{l }
3     Column 1 & Column 2 & column 3 \\
4     Some more & Some more & More data \\
5   \end{tabular}
6   \caption{}
7   \label{}
8 \end{table}
```

Making tables

Use the table and tabular environments:

```
1 \begin{table}[float]
2   \begin{tabular}{l }
3     Column 1 & Column 2 & column 3 \\
4     Some more & Some more & More data \\
5   \end{tabular}
6   \caption{}
7   \label{}
8 \end{table}
```

Give the number of columns after tabular:

- ▶ left: l
- ▶ right: r
- ▶ center: c
- ▶ custom width: p{xcm}

There are some great online tools to quickly generate table environments from excel, like

<https://www.tablesgenerator.com!>

Using pictures with graphicx

Use the figure environment:

```
1 \usepackage{graphicx}
2 ...
3 \begin{figure}[floats]
4     \centering
5     \includegraphics[sizing]{PATH_TO_PICTURE}
6     \caption{}
7     \label{}
8 \end{figure}
```

Using pictures with graphicx

Use the figure environment:

```
1 \usepackage{graphicx}
2 ...
3 \begin{figure}[floats]
4     \centering
5     \includegraphics[sizing]{PATH_TO_PICTURE}
6     \caption{}
7     \label{}
8 \end{figure}
```

Floats control positioning on the page

- ▶ t = top of page
- ▶ b = bottom of page
- ▶ h = relative in the text
- ▶ H (with float package) = relative in text, overriding \LaTeX placement algorithm

sizing options:

- ▶ width = 0.8\linewidth
- ▶ scale = 0.8

Practice - adding images with graphicx

Draw out the Chan-Lam reaction from the DoE experiment, and add it to the document

Using minipages

Can separate page into sections, and have two figures on the same line

```
1 \usepackage[labelfont=bf]{caption,subcaption}
2 ...
3 \begin{figure}[h]
4     \centering
5
6
7
8     \caption{}
9     \label{}
10 \end{figure}
11
```


Using minipages

```
1 \usepackage[labelfont=bf]{caption,subcaption}
2 ...
3 \begin{figure}[h]
4     \centering
5     \begin{minipage}{0.49\linewidth}
6         \includegraphics{}
7         \subcaption{}
8     \end{minipage}
9     \begin{minipage}{0.49\linewidth}
10        \includegraphics{}
11        \subcaption{}
12    \end{minipage}
13    \caption{}
14    \label{}
15 \end{figure}
16
```

All package documentation can be found on CTAN
(Comprehensive TeX Archive Network) – for specific package
documentation, go to `CTAN.org/package/"package-name"`

Some useful packages

Some useful packages

geometry

A flexible and easy interface to page dimensions

```
1 \usepackage[left=20mm, right=20mm, top=20mm, bottom=20  
   mm]{geometry}  
2  
3 \newgeometry{left=10mm, right=10mm}  
4 \restoregeometry  
5
```

Some useful packages

geometry

A flexible and easy interface to page dimensions

```
1 \usepackage[left=20mm, right=20mm, top=20mm, bottom=20  
   mm]{geometry}  
2  
3 \newgeometry{left=10mm, right=10mm}  
4 \restoregeometry  
5
```

xcolor

Easy driver-independent access to several kinds of colors, tints, shades, tones, and mixes of arbitrary colors by means of color expressions

```
1 \definecolor{weird-red}{rgb}{163, 82, 77}  
2  
3 \textcolor{weird-red}{Some text in weird red}  
4
```

Some useful packages

setspace

Support for setting the spacing between lines in a document

```
1 \doublespacing % double-spaced text
```

```
2
```

Some useful packages

setspace

Support for setting the spacing between lines in a document

```
1 \doublespacing % double-spaced text
```

```
2
```

titlesec

Provides an interface to sectioning commands for selection from various title styles

```
1 \titleformat*{\section}{\normalfont\Large\bfseries}
```

```
2 \titlespacing*{\section}{0pt}{0pt}{0pt}
```

```
3
```

```
4 \titleformat*{\subsection}{\normalfont\large\bfseries}
```

```
5 \titlespacing*{\subsection}{0pt}{0pt}{0pt}
```

```
6
```

Some useful packages

caption and subcaption

Custom captions, and provides support for subcaptions

```
1 \usepackage[labelfont=bf]{caption,subcaption}  
2
```


Some useful packages

caption and subcaption

Custom captions, and provides support for subcaptions

```
1 \usepackage[labelfont=bf]{caption,subcaption}
2
```

wrapfig

Wrapped text around figures

```
1 \begin{wrapfig}[h]
2 ...
3 \end{wrapfig}
4
```

Some useful packages

hyperref

Hyperreferencing within documents and support for external hyperlinks

```
1 \usepackage[hidelinks=true]{hyperref}
2 ...
3 % By using this package, all references and cross-
   references are automatically hyperlinked
4 This link takes you to \href{google.com}{google}!
5
```

Some useful packages

hyperref

Hyperreferencing within documents and support for external hyperlinks

```
1 \usepackage[hidelinks=true]{hyperref}
2 ...
3 % By using this package, all references and cross-
   references are automatically hyperlinked
4 This link takes you to \href{google.com}{google}!
5
```

float

Provides new float modifier “H”

```
1 \usepackage{float}
2 ...
3 \begin{figure}[H]
4 ...
5 \end{figure}
6
```

Some useful packages

pdfpages

Simplifies including multi-page PDF's within a \LaTeX document

```
1 \includepdf{PATH/file.pdf}
```

```
2
```

Some useful chemistry packages

mhchem

Commands for typesetting chemical molecular formulae and equations, hazard and precaution statements, and risk and safety phrases

`\ce{Na2SO4}` = Na_2SO_4

Some useful chemistry packages

mhchem

Commands for typesetting chemical molecular formulae and equations, hazard and precaution statements, and risk and safety phrases

`\ce{Na2SO4}` = Na₂SO₄

natbib

Library of common reference formats (including RSC, ACS provide their own bib-style package), advanced customisation options for citations and references

```
1 \usepackage[super, comma, sort-and-compress]{natbib}
2 ...
3 \bibliographystyle{rsc}
4
```

Some useful chemistry packages

chemstyle

A bundle of packages containing auto-numbering compounds, schemes as well as figures, latin phrases, common chemical SI units and alkyl radicals. Works with mhchem.

- ▶ `\iPr = iPr`
- ▶ `\SI{3}{molar} = 3 mol dm-3`
- ▶ `\invacuo = in vacuo`

```
1 \begin{scheme}[float]
2   \schemeref[TMPn]{a-cool-compound}
3   \includegraphics{chemdraw.eps}
4   \caption
5 \end{scheme}
6
```

Auto-numbering compounds with chemstyle

- ▶ Provided by the chemscheme package in chemstyle

Auto-numbering compounds with chemstyle

- ▶ Provided by the chemscheme package in chemstyle
- ▶ *Must* use \LaTeX -DVI-PS compiler, instead of pdf \LaTeX

Auto-numbering compounds with chemstyle

- ▶ Provided by the `chemscheme` package in `chemstyle`
- ▶ *Must* use \LaTeX -DVI-PS compiler, instead of pdf \LaTeX
- ▶ *Must* use Encapsulated PostScript (`.eps`)

Auto-numbering compounds with chemstyle

- ▶ Provided by the chemscheme package in chemstyle
- ▶ *Must* use \LaTeX -DVI-PS compiler, instead of pdf \LaTeX
- ▶ *Must* use Encapsulated PostScript (.eps)
- ▶ *Must* use graphicx package

Auto-numbering compounds with chemstyle

- ▶ Provided by the chemscheme package in chemstyle
- ▶ *Must* use \LaTeX -DVI-PS compiler, instead of pdf \LaTeX
- ▶ *Must* use Encapsulated PostScript (.eps)
- ▶ *Must* use graphicx package
- ▶ TMP numbers written on .cdxml files are replaced with compound number

Auto-numbering compounds with chemstyle

- ▶ Provided by the chemscheme package in chemstyle
- ▶ *Must* use \LaTeX -DVI-PS compiler, instead of pdf \LaTeX
- ▶ *Must* use Encapsulated PostScript (.eps)
- ▶ *Must* use graphicx package
- ▶ TMP numbers written on .cdxml files are replaced with compound number
- ▶ Schemes drawn in chemdraw are saved as .eps, then included in \LaTeX

Auto-numbering compounds with chemstyle

- ▶ Provided by the chemscheme package in chemstyle
- ▶ *Must* use \LaTeX -DVI-PS compiler, instead of pdf \LaTeX
- ▶ *Must* use Encapsulated PostScript (.eps)
- ▶ *Must* use graphicx package
- ▶ TMP numbers written on .cdxml files are replaced with compound number
- ▶ Schemes drawn in chemdraw are saved as .eps, then included in \LaTeX
- ▶ Compounds are numbered based on the order they appear in a figure in the text

Practice – auto-numbering compounds with chemstyle

```
1 ...  
2 \usepackage{chemstyle}  
3 \usepackage{graphicx}  
4 \usepackage{epstopdf}  
5  
6 \begin{document}  
7 ...  
8 \begin{scheme}[h]  
9     \includegraphics{.eps}  
10    \caption{A reaction}  
11 \end{scheme}  
12 ...  
13 \end{document}  
14
```

Practice – auto-numbering compounds with chemstyle

```
1 ...  
2 \usepackage{chemstyle}  
3 \usepackage{graphicx}  
4 \usepackage{epstopdf}  
5  
6 \begin{document}  
7 ...  
8 \begin{scheme}[h]  
9     \schemeref[TMP1]{starting-material}  
10    \schemeref[TMP2]{product}  
11    \includegraphics{.eps}  
12    \caption{A reaction}  
13 \end{scheme}  
14 ...  
15 \end{document}  
16
```


Compiling with \LaTeX -Dvi-Ps

\LaTeX -dvi-ps can only compile with .eps images - png's and jpeg's will not work!! In overleaf:

- ▶ GUI-based approach
- ▶ Click menu in the top left corner of the GUI
- ▶ Change the compiler from pdf \LaTeX to \LaTeX
- ▶ Terminal-based approach
- ▶ Ensure Mik \TeX is installed
- ▶ *case sensitive!*
- ▶ `cd PATH-TO-.tex-FILE`
- ▶ `latex file.tex`
- ▶ `dvips file.dvi`
- ▶ `ps2pdf file.ps`

Referencing with bibT_EX

Create a new .txt file with the extension .bib

In the .tex file, use \bibliography

Give a bibliography style

Can use different bibliography files for different sections

Use @format to describe the format being referenced

Surround data in {}

Use comma as a delimiter

Example in the .bib file:

```
1 @article{label,  
2   author = {},  
3   title = {},  
4   volume = {},  
5   year = {},  
6   journal = {},  
7   pages = {},  
8 },  
9
```

BibT_EX will sort the formatting to match the chosen bibliography style!

Referencing in text

With the natbib package and super options:

- ▶ `\cite{label}` gives a superscript number, and should be used after the end of the sentence
- ▶ `\citeauthor{label}` gives the author. If there are multiple, it will appear as *Author et al.*

Practice – referencing in text

Set up a .bib file and add some references to the document.
Practice using `\citeauthor` and `\cite`

Some quality-of-life macros

Remove all hyphenation:

```
1 \tolerance=1
2 \emergencystretch=\maxdimen
3 \hyphenpenalty=10000
4 \hbadness=10000
5
```

Adjust paragraph spacing

```
1 \setlength{\parskip}{0.5\baselineskip}%
2 \setlength{\parindent}{0cm}
3
```

Some chemical quality-of-life macros

```
1 \newcommand{\via}{\textit{via }}
2 \newcommand{\snX}[1]{S\textsubscript{N}#1}
3 \newcommand{\insitu}{\textit{in situ}}
4 \newcommand{\invacuo}{\textit{in vacuo}}
5 \newcommand{\nmr}[2]{\textsuperscript{#1}#2 NMR}
6 \newcommand{\degC}{\text{\textcircled{C}}}
7 \newcommand{\Degree}{\text{\textcircled{D}}}
8 \newcommand{\incsch}[1]{\includegraphics[scale
=0.8]{#1}}
9 \newcommand{\Rf}{R\textsubscript{\textit{f}}}
10 \newcommand{\kJmol}{\text{\textcircled{k}J\textit{per}\textit{mole}}}
11 \newcommand{\vmax}{FT-IR $\nu_{\text{\textit{max}}}$ (film)
/\text{\textcircled{s}i\textit{per}\textit{cm}}}
12 \newcommand{\esiplus}[4]{HRMS (Q-TOF) Exact mass
calcd. for [#1$\text{\textcircled{+}}$] [M + #2$\text{\textcircled{+}}$] calcd. #3, found
#4.}
```

Making your own macros

If you find yourself writing something lots of times, maybe consider making a command that will do it simply and quickly

Making your own commands is simple!

- ▶ Use `\newcommand{}{}`
- ▶ First `{}` defines your command
- ▶ Add variables using `#`
- ▶ `\renewcommand{}{}` to re-define a command

Making your own macros

If you find yourself writing something lots of times, maybe consider making a command that will do it simply and quickly

Making your own commands is simple!

- ▶ Use `\newcommand{}{}`
- ▶ First `{}` defines your command
- ▶ Add variables using `#`
- ▶ `\renewcommand{}{}` to re-define a command

Example:

```
1 \newcommand{\hello}[1]{Hello [#1]! How are you? Bye  
  [#1]!}  
2 ...  
3 \hello{John} = Hello John! How are you? Bye John!  
4 \hello{Matthew} = Hello Matthew! How are you? Bye  
  Matthew!  
5 \renewcommand{\hello}[1]{Hi [#1]! How you doinnnn.}  
6
```


Inheritance in L^AT_EX

Inheritance in \LaTeX

- ▶ Object-oriented languages like Java or Python pass information contained in a parent class to a child class

Inheritance in \LaTeX

- ▶ Object-oriented languages like Java or Python pass information contained in a parent class to a child class
- ▶ \LaTeX is a macro-based language, so classes aren't the same, but inheritance can still be passed from a parent `.tex` to a child `.tex`

Inheritance in \LaTeX

- ▶ Object-oriented languages like Java or Python pass information contained in a parent class to a child class
- ▶ \LaTeX is a macro-based language, so classes aren't the same, but inheritance can still be passed from a parent `.tex` to a child `.tex`
- ▶ Use `\include{PATH_TO_FILE/name.tex}`

Inheritance in L^AT_EX

- ▶ Object-oriented languages like Java or Python pass information contained in a parent class to a child class
- ▶ L^AT_EX is a macro-based language, so classes aren't the same, but inheritance can still be passed from a parent `.tex` to a child `.tex`
- ▶ Use `\include{PATH_TO_FILE/name.tex}`
- ▶ Packages and user-defined commands (functions) in the parent file are passed to the child, but child classes pass none of this information to the parent file

Inheritance in L^AT_EX

- ▶ Object-oriented languages like Java or Python pass information contained in a parent class to a child class
- ▶ L^AT_EX is a macro-based language, so classes aren't the same, but inheritance can still be passed from a parent `.tex` to a child `.tex`
- ▶ Use `\include{PATH_TO_FILE/name.tex}`
- ▶ Packages and user-defined commands (functions) in the parent file are passed to the child, but child classes pass none of this information to the parent file

Example:

```
1 \title{Thesis.tex}
2 ...
3 \begin{document}
4 \include{Introduction.tex}
5 \include{Results-and-discussion.tex}
6 \include{Conclusion.tex}
7 \include{Experimental.tex}
8 ...
9
```

- ▶ If you have any questions, ask!
- ▶ Overleaf have some really great tutorials on all things \LaTeX – I often find myself on overleaf when I've forgotten how to do something
- ▶ Read the documentation – normally provides excess information, but you can learn a lot of new (useful) commands that make your life easy!
- ▶ If you have a problem, try google – 99 times in 100, someone else has had the same problem as you. Smart people on StackExchange, overleaf, and \LaTeX forums will provide great solutions, and even smarter people write packages to solve these issues!