LaTeX Crash Course
GRACE Transferable Skills

Dr. Anthea Alberto Dr. Ina Serif

16.05.2023
09:00-13:00

Intro

Welcome!

Agenda - 16.05.2023

09:00-09:30	Introductory round
09:30-09:45	Pros & cons of using LaTeX
09:45-10:45	Structuring documents, syntax etc.
10:45-11:15	Break
11:15-11:45	Typography
11:45-12:15	Exercise
12:15-12:30	Discussing the exercise
12:30-13:00	Typesetting

Materials

You can find all the materials on our GitHub repository.

Not everything is there yet, but you'll have full access to the slides, exercises and solutions by the end of the course.

About Us - Anthea Alberto

- MA in Political Science from University of Zurich
- PhD on political communication and coalition governments from Humboldt-Universität zu Berlin
- Focus on quantitative text analysis
- Currently research navigator/research support for RISE at Uni Basel

About Us - Ina Serif

- Studied History, German and Italian at the University Freiburg (D) (Staatsexamen)
- PhD in Medieval History about manuscript transmission from University Freiburg (D)
- Interested in making life and work easier by using computers
- Currently assistant for Digital and Premodern History at the University Basel

Introductory Round

Please tell us a bit about yourself:

- Name
- Field of study
- Special requirements from your field (equations, specific typography etc.)

What is LaTeX?

- In simple terms: it's a typesetting software
- Main difference to Word et al.: what you see is not what you get
- LaTeX uses a plain text editor that later gets compiled into a nicely typeset PDF
- LaTeX operates with *commands* and *environments* - things you will learn about today

Where and how can I use LaTeX?

This course uses Overleaf, an online LaTeX editor, as its main teaching tool.

The benefit of Overleaf is that you don't have to install anything, you just need to make an account.

Plus, the free version is usually more than enough, particularly for working on solo projects.

The drawbacks are the online requirement and the fact that some features require a subscription.

Where and how can I use LaTeX?

Many editors exist that allow working with LaTeX while being offline.

I personally use TeXstudio and MiKTeX as package manager.

Important: there rarely is “one true way” of doing things in LaTeX. This course aims to get you started and teach you good practices.

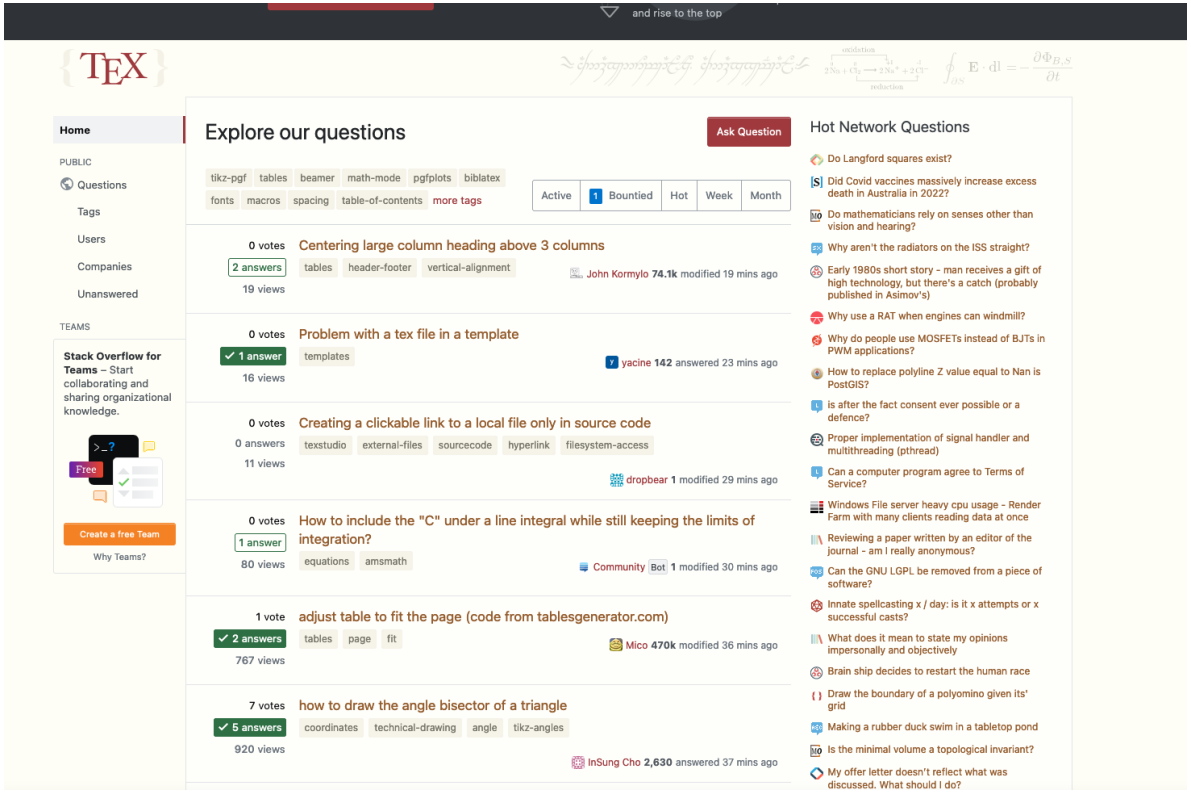
Why you should (not) learn to use LaTeX: a non-comprehensive list

Pro

LaTeX...

- ...focuses on **content**, not layout.
- ...makes **bibliographies** easier to handle.
- ...has many **libraries and packages** that make life easier.
- ...is customizable and flexible.
- ...has a **big user base** and there are many tutorials.
- ...makes it easier to include **mathematical notation.**

Big user base



Screenshot taken on March 29th 2023, 15:35

Contra

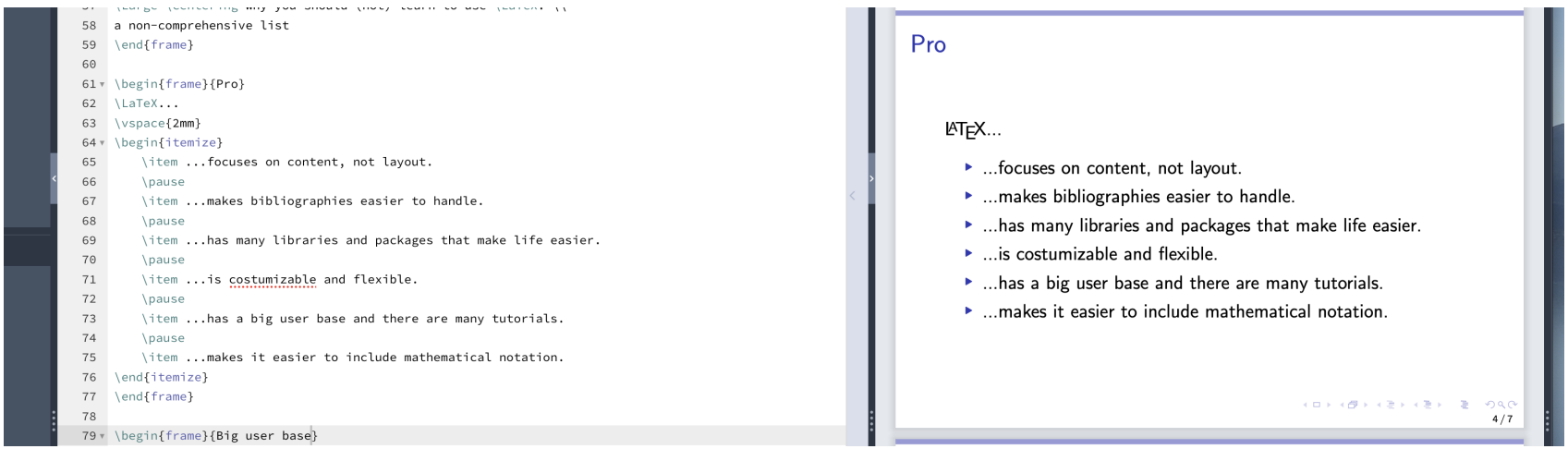
We would not be teaching this course if we thought you *shouldn't* actually learn LaTeX.

Therefore, the next section is not necessarily about arguments *against* learning it, but rather some *caveats*; or things to keep in mind.

Contra

- There is a **steep learning curve** and **potential for frustration**, particularly at the beginning.
- What you see is **not** what you get (unlike Word et al.)
- "More options" sometimes comes at the expense of efficiency.
- **Version control** is not always straightforward.
- Obstacles to working on a document collaboratively/simultaneously.
- Potential co-authors might not be familiar with LaTeX.

WYSINWYG

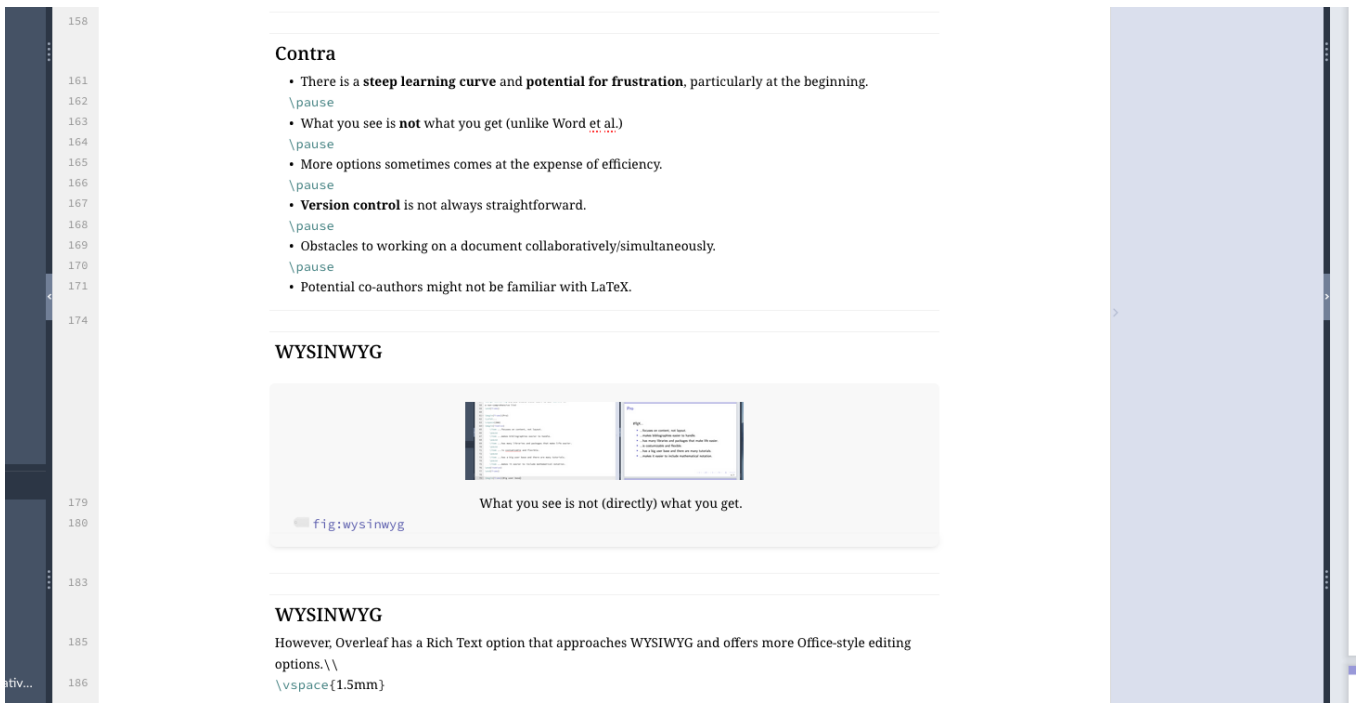


```
58 a non-comprehensive list
59 \end{frame}
60
61 \begin{frame}{Pro}
62 \LaTeX...
63 \vspace{2mm}
64 \begin{itemize}
65     \item ...focuses on content, not layout.
66     \pause
67     \item ...makes bibliographies easier to handle.
68     \pause
69     \item ...has many libraries and packages that make life easier.
70     \pause
71     \item ...is costumizable and flexible.
72     \pause
73     \item ...has a big user base and there are many tutorials.
74     \pause
75     \item ...makes it easier to include mathematical notation.
76 \end{itemize}
77 \end{frame}
78
79 \begin{frame}{Big user base}
```


What you see is not (directly) what you get.

WYSINWYG

However, Overleaf has a Rich Text option that approaches WYSIWYG and offers more Office-style editing options.



Version Control

- For Overleaf free plan: only last 24 hours available.
- Syncing with git, GitHub or Dropbox is possible for premium users.
- For other editors (e.g. TeXstudio), versioning is possible via Dropbox, GitHub etc.
- TeXstudio has its own version control system/git integration, but it needs to be set up first.

Working collaboratively

- Limited options on Overleaf's free plan: only one collaborator per project
- For other editors: similar problems as with version control; no straightforward option to work on a document simultaneously
- Possibility to use external software to facilitate collaboration

Structuring LaTeX documents

Structure of a LaTeX document

preambel and body

preambel defines global options for layout, font, bibliography asf.

body contains the actual text

minimal example:

```
\documentclass[10pt,a4paper]{article}        ⟵ preambel
\begin{document}
Some meaningful text.                        ⟵ body
\end{document}
```

Structure of a LaTeX document

Some meaningful text.

Structure of a LaTeX document

preambel, body and top matter/title page

minimal example:

```
\documentclass[10pt,a4paper]{article}        ⟵ preambel
\begin{document}
\title{Title of the document}                ⟵ topmatter
\author{Arthur Dent}
\date{\today}
\maketitle
Some meaningful text.                        ⟵ body
\end{document}
```

Structure of a LaTeX document

Title of the document

Arthur Dent

April 25, 2023

Some meaningful text.

Structuring LaTeX documents

chapters, sections, paragraphs as common document structures

```
\chapter{Chapter name}
\section{Header name on level 1}
\subsection{Header name on level 2}
\subsubsection{Header name on level 3}
\paragraph{Paragraph name}
\subparagraph{Subparagraph name}
```

Structuring LaTeX documents

Note: All structural elements will be numbered automatically:

1 A section of the document

Some meaningful text.

1.1 A subsection

1.1.1 Followed by a subsubsection

More meaningful text.

1.2 Another subsection

A very interesting fact.

2 Back to section level

And even more meaningful text.

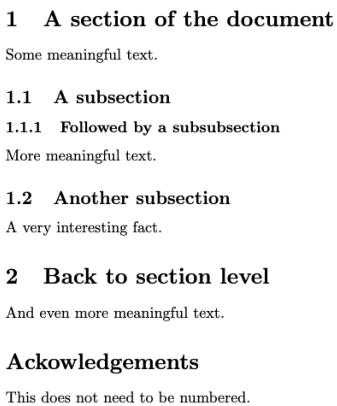
You can leave subsections unnumbered by using `\setcounter`:

```
\setcounter{secnumdepth}{1}
```

Structuring LaTeX documents

You can leave certain chapters/sections unnumbered, using "*".
This will also not include the section into the table of contents:

```
\section*{Acknowledgements}
```



1 A section of the document

Some meaningful text.

1.1 A subsection

1.1.1 Followed by a subsubsection

More meaningful text.

1.2 Another subsection

A very interesting fact.

2 Back to section level

And even more meaningful text.

Ackowledgements

This does not need to be numbered.

Structuring LaTeX documents

As with all default settings in LaTeX:
You can change the behaviour of everything, e.g. vertical space between headers and text, but think twice if it's necessary.

You can insert comments that explain modifications, or just to make a note to yourself, or to keep a paragraph in a document, but without typesetting it, using "%".

```
This is a very good thought.  % look up missing
reference!
```

Syntax, commands, environments and packages

How to

- write commands
- change default behaviour
- use environments
- use packages

Basic syntax and commands

Basic syntax:

```
\ + command
\ + command + {mandatory argument}
\ + command + [optional argument], + {mandatory argument}
```

Basic syntax and commands

Basic syntax:

inserting page break:

`\newpage`

changing font:

`\textit{italic text}` ⟶ *italic text*

inserting a section:

`\section{Section title}` ⟶ **1 Section title**

defining document class:

`\documentclass[10pt,a4paper]{article}` ⟶
a document with default settings according to class "article", with changes regarding font size and paper format

Environments

environments are used to apply commands to a defined section of a document

use existing or define new environments

environments start with \begin{name} and end with \end{name}

```
\begin{itemize}
  \item Item 1
  \item Item 2
\end{itemize}
```

$\longrightarrow$

- Item 1
- Item 2

Packages

packages contain additional LaTeX commands to change style features or to modify existing ones

most LaTeX distributions, such as MiKTeX, include a package manager

packages can also be installed via the command line

packages are called in the preambel:

```
\usepackage[english]{babel}
```

Packages

packages are called in the preambel:

```
\usepackage[paper=letterpaper,
marginparwidth=3in, % Length of section titles
marginparsep=-3in, % Space between titles and text
margin=1in, % 1 inch margins
includemp] % includes the margin notes
{geometry}
```

Typography primer

- font style and size
- quotes, citations, footnotes
- in-text referencing

Font style and size

font style:

```
\underline{This} is a \emph{text} with a \textbf{lot}
of different \textsc{styles}.
```

<u>This</u> is a *text* with a **lot** of different STYLES.

Font style and size

font size:

```
{\tiny This} is a {\Huge text} with a {\Large lot} of
different {\LARGE sizes}.
```

{\tiny This} is a {\Huge text} with a {\Large lot} of different {\LARGE sizes}.

quotes, citations and footnotes

```
\begin{quote}
  quoted text.\footnote{footnote text}
\end{quote}
```

> This is a longer quote from a scientific article that I would like to cite in its whole beauty, including the reference.[1]

[1] Adams, Douglas: The Hitchhiker's Guide to the Galaxy. London 1979, p. 42.

quotes, citations and footnotes

How to use your reference manager (Zotero, Citavi, . . .) with LaTeX to create a bibliography will be covered next session – stay tuned!

in-text referencing

you can mark structural, textual or graphic elements in a document and reference to it

you can label a section or a figure:

`\section{Great section title}\label{great}`

(figures next session!)

As discussed in `\ref{great}`, I will now ...

As discussed in chapter 2, I will now ...

Exercise


Astro Joke of the Week @JokesAstro · 23. Apr.
I didn't do a LaTeX joke in a while.
Beautiful Typesetting with LaTeX
Overfull \hbox (9.895pt too wide)
O RLY?
14
133
1.288
40.235
Diesen Thread anzeigen

documentclass

At the beginning of each LaTeX file, you have to specify what *type* of document you want to create.

This happens with the *documentclass* command.

documentclass also lets you set additional options for your document.

These slides uses a very simple form of document class: `\documentclass{beamer}`, where what's in the curly brackets indicates the document type.

Frequently used arguments for `\documentclass{}`:

- article
- beamer
- report
- book or scrbook
- letter

More options

In addition to type, you can add font size, paper size and format and more to costumize your document.

This is done via square brackets before the curly ones.

For example:

```
\documentclass[12pt,a4paper,oneside]{scrbook}
```

Standard options (taken from texblog) include:

- Font size (10pt, 11pt, 12pt...)
- Paper size and format (a4paper, letterpaper...)
- Multiple columns (onecolumn, twocolumn)
- Title page behavior (titlepage, notitlepage)
- Draft mode (draft)

Most of these options and costumizations can also be done with packages like *geometry*, as you've seen in an earlier slide.

Example:

```
\usepackage[a4paper,top=2cm,bottom=2cm,left=3cm,
right=3cm,marginparwidth=1.75cm]{geometry}
```

Themes

For presentations like this one, you can use a *theme* for your slides.

This presentation uses the *Szeged* theme; you call it by putting `\mode<presentation> {\usetheme{Szeged}}` into the preamble.

You can find a gallery of themes here. There are many more options, and you can of course make your own.

Language options

Set the language: either within documentclass, or with the *babel* package.

These slides use `\usepackage[english]{babel}`.

You can use multiple languages within the same document, just separate them with a comma.

Language options

Babel supports various languages (documentation here and here), but if you require a non-Latin alphabet (like Arabic, Hebrew, Japanese, Mandarin...) the situation is unfortunately a bit more complicated.

You can find an intro on using the polyglossia package with Overleaf here.