Introduction to LaTeX and Overleaf

Frank Würthner

February 2024

Table of Contents

What and Why

LaTeX is a typesetting system used for producing scientific, mathematical, and physical documents of high print quality.

What and Why

LaTeX is a typesetting system used for producing scientific, mathematical, and physical documents of high print quality.

- Content is handled separately from the document's appearance (as opposed to WYSIWYG)

What and Why

LaTeX is a typesetting system used for producing scientific, mathematical, and physical documents of high print quality.

- Content is handled separately from the document's appearance (as opposed to WYSIWYG)
- LaTeX is much better for more complex mathematics, tables and technical content for physical sciences

What and Why

LaTeX is a typesetting system used for producing scientific, mathematical, and physical documents of high print quality.

- Content is handled separately from the document's appearance (as opposed to WYSIWYG)
- LaTeX is much better for more complex mathematics, tables and technical content for physical sciences
- Easy management of footnotes, references and citations

What and Why

LaTeX is a typesetting system used for producing scientific, mathematical, and physical documents of high print quality.

- Content is handled separately from the document's appearance (as opposed to WYSIWYG)
- LaTeX is much better for more complex mathematics, tables and technical content for physical sciences
- Easy management of footnotes, references and citations
- Can automatically accomplish tedious tasks (Table of contents and figure lists)

Overleaf has thousands of templates

Document Structure

- Preamble: Document's setup section

Document Structure

- Preamble: Document's setup section
  - Define class type

```
\documentclass[14pt,
letterpaper]{article}
```

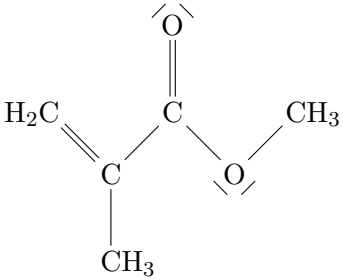
Document Structure

- Preamble: Document's setup section
  - Define class type
  - Add additional parameters

```
\documentclass[14pt,
letterpaper]{article}
\documentclass[14pt,
letterpaper]{article}
```

Document Structure

- Preamble: Document's setup section
  - Define class type
  - Add additional parameters
- Packages: Extends the capabilities of LaTeX with external files



Document Structure

- Preamble: Document's setup section
  - Define class type
  - Add additional parameters
- Packages: Extends the capabilities of LaTeX with external files
- Title, author and date

- `\title{My first LaTeX document}`
- `\author{<Name>}`
- `\date{Feb 2024}`

Document Structure

- Preamble: Document's setup section
  - Define class type
  - Add additional parameters
- Packages: Extends the capabilities of LaTeX with external files
- Title, author and date
- Press `Enter` twice to start a new paragraph or type either `\\` or `\newline`

Fonts and basic Formatting

- We can make text **bold**, *italicised*, or underlined.

Fonts and basic Formatting

- We can make text **bold**, *italicised*, or underlined.
  - `\textbf{bold}`
  - `\textit{italicised}`
  - `\underline{underlined}`

Fonts and basic Formatting

- We can make text **bold**, *italicised*, or underlined.
  - `\textbf{bold}`
  - `\textit{italicised}`
  - `\underline{underlined}`
- Create (numbered) lists with `\begin{itemize}` or `\begin{enumerate}`

Fonts and basic Formatting

- We can make text **bold**, *italicised*, or underlined.
  - `\textbf{bold}`
  - `\textit{italicised}`
  - `\underline{underlined}`
- Create (numbered) lists with `\begin{itemize}` or `\begin{enumerate}`

- Description environment: `\begin{description}`
- Change labels of individual entries: `\item[!] An item`
- Use package `\usepackage{layouts}`, and `\listdiagram` to show LaTeX list layout

Fonts and basic Formatting

- We can make text **bold**, *italicised*, or underlined.
  - `\textbf{bold}`
  - `\textit{italicised}`
  - `\underline{underlined}`
- Create (numbered) lists with `\begin{itemize}` or `\begin{enumerate}`
- Embed maths *inline* with `\(...\)` or `$...$` and in *display math mode* with `\[...\]`

Fonts and basic Formatting

- We can make text **bold**, *italicised*, or underlined.
  - `\textbf{bold}`
  - `\textit{italicised}`
  - `\underline{underlined}`
- Create (numbered) lists with `\begin{itemize}` or `\begin{enumerate}`
- Embed maths *inline* with `\(...\)` or `$...$` and in *display math mode* with `\[...\]`

If $v_1, ..., v_p$ are in $\mathbb{R}^n$ then the set of all linear combinations of $v_1, ..., v_p$ is denoted by $\text{Span}\{v_1, ..., v_p\}$ and is called the **subset of $\mathbb{R}^n$ spanned by** $v_1, ..., v_p$. That is, $\text{Span}\{v_1, ..., v_p\}$ is the collection of all vectors that can be written in the form

$$c_1 v_1 + c_2 v_2 + \cdots + c_p v_p$$

with $c_1, ..., c_p$ scalars.

Chapters and Sections

Larger documents can be split into different parts:

- `\chapter`
- `\section`
- `\subsection`
- `\subsubsection`
- `\paragraph`
- `\subparagraph`

Chapters and Sections

Larger documents can be split into different parts:

- `\chapter`
- `\section`
- `\subsection`
- `\subsubsection`
- `\paragraph`
- `\subparagraph`

All parts in a document can be rendered at the beginning of the document with `\tableofcontents`

Images

1. Use the **Insert Figure** button(), located on the editor toolbar, to insert a figure into **Visual Editor** or **Code Editor**.
2. Copy and paste an image into **Visual Editor** or **Code Editor**.
3. Use **Code Editor** to write LaTeX code that inserts a graphic and places it inside a figure environment.

Images

1. Use the **Insert Figure** button(), located on the editor toolbar, to insert a figure into **Visual Editor** or **Code Editor**.
2. Copy and paste an image into **Visual Editor** or **Code Editor**.
3. Use **Code Editor** to write LaTeX code that inserts a graphic and places it inside a figure environment.

```
\begin{figure}
\centering
\includegraphics[width=0.5\linewidth]
{Vrije_logo}
\label{fig:enter-label}
\end{figure}
```

Images

1. Use the **Insert Figure** button(), located on the editor toolbar, to insert a figure into **Visual Editor** or **Code Editor**.
2. Copy and paste an image into **Visual Editor** or **Code Editor**.
3. Use **Code Editor** to write LaTeX code that inserts a graphic and places it inside a figure environment.

Reference to figures and Tables with
`\ref{fig:enter-label}` or
`\hyperref[fig:enter-label]{text}`

Images

1. Use the **Insert Figure** button(), located on the editor toolbar, to insert a figure into **Visual Editor** or **Code Editor**.
2. Copy and paste an image into **Visual Editor** or **Code Editor**.
3. Use **Code Editor** to write LaTeX code that inserts a graphic and places it inside a figure environment.

Add images to an **Attachments** folder, and set `\graphicspath{{./Attachments/}}`

Reference to figures and Tables with `\ref{fig:enter-label}` or `\hyperref[fig:enter-label]{text}`

Images

- You can wrap text around images
  using the `wrapfigure` environment

Images

- You can wrap text around images using the `wrapfigure` environment

proident elit commodo dolor sint qui non ullamco nisi nisi consectetur amet eu sint sunt tempor laborum eu Lorem deserunt consectetur ullamco reprehenderit ut labore voluptate occaecat deserunt proident enim

Tables

Easily create tables with Tables Generator. Visit Overleaf's website for more detail on complex tables.

```
\begin{table}[]
    \centering
    \begin{tabular}{c|c}
       a & b \\
       c & d
    \end{tabular}
    \label{tab:my_label}
\end{table}
```

a	b
c	d

header1	header2	header3
cell1	cell2	cell3
cell4	cell5	cell6
cell7	cell8	cell9

Paragraphs

- To create centered, left aligned, or right aligned text, use the `center`, `leftflush`, and `rightflush` environments.

Paragraphs

- To create centered, left aligned, or right aligned text, use the `center`, `leftflush`, and `rightflush` environments.
- Toggle the alignment in a specific environment with `\centering`, `\raggedright` and `\raggedleft` respectively.

Paragraphs

- To create centered, left aligned, or right aligned text, use the `center`, `leftflush`, and `rightflush` environments.
- Toggle the alignment in a specific environment with `\centering`, `\raggedright` and `\raggedleft` respectively.
- Every paragraph (other than the first) is automatically indented. Use `\noindent` to unindent a paragraph.