A Beginner's Gentle Introduction to LaTeX

AI Assistant (Gemini)

May 13, 2025

Abstract

This document serves as a very basic introduction to LaTeX, a powerful document preparation system. It covers the essential elements you need to get started, including document structure, sections, text formatting, lists, basic mathematical expressions, and tables. For each feature, both the LaTeX code and a representation of the rendered output are provided. The goal is to create a simple, compilable document that helps beginners understand the fundamentals of LaTeX. This version includes a clickable Table of Contents and improved content flow without unnecessary page breaks before sections.

Contents

1 What is LaTeX?

LaTeX (pronounced "LAH-tek" or "LAY-tek") is a high-quality typesetting system; it includes features designed for the production of technical and scientific documentation. LaTeX is not a "What You See Is What You Get" (WYSIWYG) word processor like Microsoft Word. Instead, you write your document using plain text and special LaTeX commands, and then compile this source file to produce a professionally formatted document, typically a PDF.

Why use LaTeX?

- **Professional Quality:** LaTeX produces beautiful documents, especially those with complex mathematical formulas and scientific notations.

- **Focus on Content:** You focus on the content, and LaTeX takes care of the formatting.

- **Structured Documents:** It's excellent for large, structured documents like theses, books, and articles, with automatic numbering of sections, figures, tables, and managing bibliographies.

- **Stability:** It's very stable and widely used in academia.

This document will guide you through the very basics.

2 Document Structure: The Basics

Every LaTeX document has a basic structure. Let's look at the minimal components of a LaTeX file.

2.1 The Preamble

The part of your `.tex` file before the `\begin{document}` command is called the preamble. Here, you define the type of document you are creating (the "document class") and can load "packages" that add new features or customize the document.

LaTeX Code (Preamble Example):

```
\documentclass[11pt]{article}
% \usepackage{somepackage} % Packages are loaded here
% \usepackage{hyperref}    % Example of loading hyperref
```

Explanation:

- `\documentclass[11pt]{article}`: This is usually the first line.

  - `article` is the document class, suitable for articles, reports, and shorter documents. Other classes include `report`, `book`, `letter`, `beamer` (for presentations).
  - `[11pt]` is an option that sets the base font size to 11 points. Default is 10pt. You can also use `12pt`.

- `% \usepackage{somepackage}`: Lines starting with `%` are comments. The `\usepackage` command loads external packages. For this guide, we are using `geometry` (for margins) and `hyperref` (for clickable links), which are standard packages. Many other packages exist, like `amsmath` for advanced math, or `graphicx` for including images.

2.2 The Document Body

The actual content of your document goes between \begin{document} and \end{document}.
  LaTeX Code (Document Body):

```
\begin{document}

% Your title, author, and date commands are often in the preamble
% but \maketitle goes inside the document environment.
\maketitle

Hello, LaTeX world! % This is your content.

\end{document}
```

  Explanation:

- \begin{document}: Marks the start of your document's content.

- \maketitle: If you've defined \title, \author, and \date in the preamble, this command will display them.

- Content like "Hello, LaTeX world!" is just typed as is. Paragraphs are separated by one or more blank lines in your source code.

- \end{document}: Marks the end of the document. Anything after this line is ignored by the compiler.

3   Sections and Subsections

LaTeX makes it easy to structure your document with chapters (in book or report classes), sections, subsections, and more. LaTeX automatically numbers these for you and includes them in the Table of Contents (if you use the \tableofcontents command, as we did in this document).
  Commands for Sectioning:

- \section{Section Title}

- \subsection{Subsection Title}

- \subsubsection{Subsubsection Title}

- \paragraph{Paragraph Title} (less common for numbering, more for structure)

- \subparagraph{Subparagraph Title}

  LaTeX Code Example:

```
\section{My First Big Topic}
This is the introduction to my first big topic.

\subsection{An Interesting Sub-Topic}
Here, I discuss a sub-topic in more detail.

\subsubsection{A Finer Point}
And here is an even more specific detail.
```

  Rendered Output (Illustrative - numbering depends on prior sections):

My First Big Topic (Example Rendering)

This is the introduction to my first big topic.

An Interesting Sub-Topic (Example Rendering)

Here, I discuss a sub-topic in more detail.

A Finer Point (Example Rendering)

And here is an even more specific detail.

(Note: In your actual document, these would be numbered, e.g., "1 My First Big Topic", "1.1 An Interesting Sub-Topic", etc., and these titles would appear in the clickable Table of Contents.)

4 Text Formatting

LaTeX provides simple commands for basic text formatting.

4.1 Bold, Italics, and Typewriter Font

LaTeX Code:

```
This is \textbf{bold text}.
This is \textit{italicized text}.
This is \texttt{typewriter (monospace) text}.
You can also use \textsc{small caps text}.
```

Rendered Output:
This is **bold text**.
This is *italicized text*.
This is `typewriter (monospace) text`.
You can also use SMALL CAPS TEXT.

4.2 Emphasizing Text

The \emph{} command is often preferred for emphasis, as it intelligently switches between italics and upright text if nested.

LaTeX Code:

```
This is \emph{emphasized text}.
And this is \textbf{bold and \emph{emphasized text within bold}}.
```

Rendered Output:
This is *emphasized text*.
And this is **bold and *emphasized text within bold***.

4.3 Font Sizes

While the base font size is set in \documentclass, you can change font sizes for portions of text: \tiny tiny text

\scriptsize scriptsize text

\footnotesize footnotesize text

\small small text

`\normalsize` normal text (this is normal)

`\large` large text

`\Large` Larger text

`\LARGE` LARGE text

`\huge` huge text

`\Huge` Huge text

LaTeX Code Example:

```
This is normal text. {\large This is large text.} This is normal again.
```

Rendered Output:

This is normal text. This is large text. This is normal again.

5 Creating Lists

LaTeX supports two main types of lists: unordered (itemized) and ordered (enumerated). Lists are created using "environments".

5.1 Itemized (Unordered) Lists

Use the `itemize` environment. Each item starts with `\item`.

LaTeX Code:

```
\begin{itemize}
    \item First bullet point.
    \item Second bullet point.
    \item Another point, which can be quite long and will wrap
      around automatically.
\end{itemize}
```

Rendered Output:

- First bullet point.

- Second bullet point.

- Another point, which can be quite long and will wrap around automatically.

5.2 Enumerated (Ordered) Lists

Use the `enumerate` environment. Each item also starts with `\item`, and LaTeX handles the numbering.

LaTeX Code:

```
\begin{enumerate}
    \item The first step.
    \item The second step.
    \item The final action.
\end{enumerate}
```

Rendered Output:

1. The first step.

2. The second step.

3. The final action.

5.3 Nested Lists

You can also nest lists within each other. LATEX will automatically change the bullet style or numbering scheme for nested lists.

LATEX Code:

```
\begin{itemize}
    \item Main item A.
    \begin{enumerate}
        \item Sub-item 1 (numbered).
        \item Sub-item 2 (numbered).
        \begin{itemize}
            \item Deeper bullet point.
        \end{itemize}
    \end{enumerate}
    \item Main item B.
\end{itemize}
```

Rendered Output:

- Main item A.

    1. Sub-item 1 (numbered).
    2. Sub-item 2 (numbered).
        - Deeper bullet point.

- Main item B.

6 Mathematical Expressions

One of LATEX's greatest strengths is its ability to typeset mathematical expressions beautifully.

6.1 Inline Mathematics

For math expressions that appear within a line of text, enclose them in dollar signs ($ ... $).

LATEX Code:

```
The famous Pythagorean theorem is $a^2 + b^2 = c^2$.
We can also write variables like $x$, $y$, and $z$.
Greek letters are common: $\alpha, \beta, \gamma, \Delta, \Omega$.
For simple fractions, use $\frac{numerator}{denominator}$, e.g., $\frac{1}{2}$.
Subscripts are $x_i$ and superscripts are $x^2$.
```

Rendered Output (as it would appear in your text):

The famous Pythagorean theorem is $a^2 + b^2 = c^2$. We can also write variables like x, y, and z. Greek letters are common: $\alpha, \beta, \gamma, \Delta, \Omega$. For simple fractions, use $\frac{numerator}{denominator}$, e.g., $\frac{1}{2}$. Subscripts are x_i and superscripts are x^2.

6.2 Displayed Mathematics

For equations that should be set apart from the text on their own line and centered, you can use double dollar signs ($$ ... $$) or, more recommended for standard LATEX, square brackets (\[... \]).

LATEX Code (using \[... \]):

```
\[ E = mc^2 \]
\[ \sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6} \]
\[ \int_{0}^{\infty} e^{-x} dx = 1 \]
```

Rendered Output (equations will be centered on their own lines):

$$E = mc^2$$

$$\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$$

$$\int_{0}^{\infty} e^{-x} dx = 1$$

Common Mathematical Constructs:

- **Exponents and Subscripts:** Use ^ for exponents (e.g., `x^2` for x^2) and _ for subscripts (e.g., `a_1` for a_1). If the exponent or subscript is more than one character, enclose it in curly braces: `x^2y` for x^{2y}, `a_ij` for a_{ij}.

- **Fractions:** `\frac{numerator}{denominator}` (e.g., `\frac{3}{4}` for $\frac{3}{4}$).

- **Square Roots:** `\sqrt{expression}` (e.g., `\sqrt{x}` for $\sqrt{x}$). For n-th roots: `\sqrt[n]{expression}` (e.g., `\sqrt[3]{x}` for $\sqrt[3]{x}$).

- **Sums and Integrals:** `\sum` for $\sum$ and `\int` for $\int$. Limits are specified using subscripts and superscripts: `\sum_i=0^N` for $\sum_{i=0}^{N}$.

- **Greek Letters:** `\alpha` (α), `\beta` (β), `\Gamma` (Γ), `\delta` (δ), etc.

- **Mathematical Symbols:** Many symbols are available: $\leq$ (`\le`), $\geq$ (`\ge`), $\neq$ (`\neq`), $\times$ (`\times`), $\cdot$ (`\cdot`), $\pm$ (`\pm`), ∞ (`\infty`), etc.

For more advanced mathematical typesetting, the `amsmath` package is highly recommended, but the basics shown here work without it.

7 Creating Tables

Tables are created using the `tabular` environment. It can seem a bit complex at first, but simple tables are straightforward.

Basic Structure:

```
\begin{tabular}{column_specifiers}
    % Table content goes here
\end{tabular}
```

Column Specifiers:

- `l`: a left-aligned column.

- `c`: a center-aligned column.

- `r`: a right-aligned column.

- `|`: inserts a vertical line between columns.

Example: `{l|c|r}` would create a three-column table with the first column left-aligned, the second centered, and the third right-aligned, with vertical lines separating them and around them if specified at ends.

Table Content:

- &: separates cell content within a row.

- \\: ends a row.

- \hline: inserts a horizontal line.

LaTeX Code for a Simple Table:

```
\begin{center} % Often good to center the table on the page
\begin{tabular}{|l|c|r|}
    \hline % Horizontal line at the top
    \textbf{Name} & \textbf{Age} & \textbf{City} \\ % Header row
    \hline % Horizontal line after header
    Alice & 30 & New York \\
    Bob & 24 & London \\
    Charlie & 35 & Paris \\
    \hline % Horizontal line at the bottom
\end{tabular}
\end{center}
```

Rendered Output:

Name	Age	City
Alice	30	New York
Bob	24	London
Charlie	35	Paris

More complex tables might involve merging cells or using other packages like `booktabs` for more professional-looking tables without vertical lines, but this covers the basics.

8 Comments in LaTeX

You've seen them throughout this document: comments. A percent sign (%) tells LaTeX to ignore the rest of the line. This is useful for adding notes to yourself or others reading your code, or for temporarily "commenting out" parts of your document.

LaTeX Code:

```
This text will be processed by LaTeX.
% This is a comment and will be ignored.
And this text will also be processed.
% \section{Temporary Section} % This section is commented out.
```

Explanation: In the example above, "This is a comment and will be ignored." and the commented-out section command will not appear in the final PDF and will not affect the document's formatting.

9 Compiling Your LaTeX Document

To turn your `.tex` source file into a viewable document (usually a PDF), you need to "compile" it.

1. **Write your LaTeX code** in a plain text file with a `.tex` extension (e.g., `mydocument.tex`).

2. **Use a LaTeX compiler.** The most common command for this is `pdflatex`. If your document is named `mydocument.tex`, you would run `pdflatex mydocument.tex` in a terminal or command prompt.

3. **Multiple Compilations:** For complex documents with cross-references (like the Table of Contents made clickable by `hyperref`), table of contents, bibliographies, etc., you might need to run the compiler multiple times (e.g., `pdflatex -> bibtex -> pdflatex -> pdflatex`). For just TOC and `hyperref` links, two runs of `pdflatex` are usually sufficient.

4. **LaTeX Editors:** Many specialized text editors and Integrated Development Environments (IDEs) for LaTeX can simplify this process with a "compile" button.

   - **Online Editors:** Overleaf is very popular and requires no local installation.
   - **Desktop Installations:**
     - TeX Live (cross-platform: Windows, macOS, Linux)
     - MiKTeX (primarily Windows, but also available for macOS and Linux)
     - MacTeX (a version of TeX Live for macOS)

   These distributions come with compilers and often an editor like TeXworks. Other editors like VS Code (with LaTeX Workshop extension), TeXstudio, or Sublime Text (with LaTeXTools) are also excellent choices.

The compilation process generates several auxiliary files (`.aux`, `.log`, `.toc`, etc.) besides the final PDF. The `.log` file is particularly important for diagnosing errors if your document doesn't compile.

10   Conclusion

This document has covered the very basic elements of LaTeX. You've learned how to:

- Structure a document with a preamble and body.

- Create sections and subsections (which are now linked from the Table of Contents).

- Apply basic text formatting.

- Make ordered and unordered lists.

- Write inline and displayed mathematical expressions.

- Construct simple tables.

LaTeX has a steeper learning curve than WYSIWYG editors, but its power, flexibility, and the quality of the output make it an invaluable tool for many, especially in academic and technical fields. The key is to start simple, practice, and gradually explore more features and packages as you need them.

Happy LaTeXing!

References and Further Reading

Here are some excellent resources to continue your LaTeX journey:

References

[1] Overleaf. (n.d.). *Learn LaTeX in 30 minutes*. Retrieved May 13, 2025, from `https://www.overleaf.com/learn/latex/Learn_LaTeX_in_30_minutes`

[2] Overleaf. (n.d.). *Documentation*. Retrieved May 13, 2025, from `https://www.overleaf.com/learn`

[3] The LaTeX Project. (n.d.). *LaTeX - A document preparation system*. Retrieved May 13, 2025, from `https://www.latex-project.org/`

[4] CTAN. (n.d.). *The Comprehensive TeX Archive Network*. (This is where LaTeX packages are hosted). Retrieved May 13, 2025, from `https://www.ctan.org/`

[5] Wikibooks. (n.d.). *LaTeX*. Retrieved May 13, 2025, from `https://en.wikibooks.org/wiki/LaTeX`

[6] Scott Pakin. (2021). *The Comprehensive LaTeX Symbol List*. (A very useful guide for finding specific symbols). Available on CTAN: `https://ctan.org/pkg/symbols-a4`