Computational Techniques

UPES Dehradun

LaTeX Handbook

Prepared by:

Nitesh Kumar

Email: nitesh.kumar@ddn.upes.ac.in

Website: https://niteshchandra039.github.io/

October 23, 2024

School of Engineering

University of Petroleum and Energy Studies
Dehradun

Contents

© 2024 Nitesh Kumar. All rights reserved.

© 2024 Nitesh Kumar. All rights reserved.

1 Introduction to LaTeX

LaTeX is a powerful typesetting system extensively used in academia, especially for scientific documents that involve complex mathematical equations, figures, and references. It allows users to focus on the content while managing the formatting and layout efficiently. Unlike WYSIWYG (what you see is what you get) editors like Microsoft Word, LaTeX operates using plain text markup, which means you define structure and style using commands.

Key features of LaTeX include:

- Precise control over document formatting.
- Easy management of bibliographies, references, and citations.
- Automatic numbering and cross-referencing.
- Superior handling of mathematical formulas.

This document will guide you through the basics of LaTeX and demonstrate how to create well-structured documents with high-quality formatting.

2 Getting Started with LaTeX

2.1 Installing LaTeX

LaTeX is available on most platforms:

1. **Windows**: Use **MikTeX** or **TeX Live**.
2. **Mac**: Install **MacTeX**.
3. **Linux**: Install via package managers, e.g., `sudo apt-get install texlive-full`.

Popular editors:

- **TeXworks** (included with MikTeX).
- **Overleaf** (online collaborative LaTeX editor).
- **Texmaker** or **VS Code** with LaTeX plugins.

2.2 First LaTeX Document

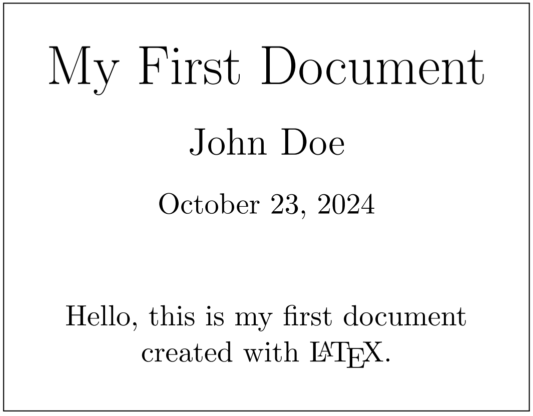
A typical LaTeX document contains a preamble and a body. Below is an example of a basic document:

© 2024 Nitesh Kumar. All rights reserved.

LaTeX Code:

```latex
\documentclass{article}
\usepackage[utf8]{inputenc}

\title{My First Document}
\author{John Doe}
\date{\today}

\begin{document}
\maketitle

Hello, this is my first document created with \LaTeX.
\end{document}

```

Output:



My First Document

John Doe

October 23, 2024

Hello, this is my first document created with LaTeX.

To compile this, run `pdflatex` and a PDF will be generated.

3 The Preamble and Body of a LaTeX Document

A LaTeX document consists of two main parts: the **preamble** and the **body**.

3.1 The Preamble

The preamble is the part of the document before the `\begin{document}` command. It is used to set up the overall structure and formatting of the document. Key components of the preamble include:

- `\documentclass{...}`: This command defines the type of document you are writing (e.g., `article`, `report`, `book`, etc.). You can also pass options to modify the appearance of the document, such as font size or paper size:

```latex
    \documentclass[12pt, a4paper]{article}

```

- `\usepackage{...}`: This command imports additional packages to enhance the functionality of your document. For example, to support UTF-8 character encoding or to add mathematical capabilities:

```latex
    \usepackage[utf8]{inputenc}
    \usepackage{amsmath}

```

- Title information commands:
  - `\title{...}`: Sets the document title.
  - `\author{...}`: Sets the author's name.
  - `\date{...}`: Sets the date. You can use `\today` to automatically insert the current date.

© 2024 Nitesh Kumar. All rights reserved.

These settings are later used when the \maketitle command is called in the body of the document.

3.2 The Body

The body of the document begins after the \begin{document} command. This is where the actual content of your document is written. You can include sections, text, lists, tables, figures, equations, and other elements. Here is an example of a simple document body:

```latex
\begin{document}
\maketitle

This is the body of the document. You can add sections like this:
\section{Introduction}
This is an introduction to my document.

You can also include mathematical equations, figures, and tables here.
\end{document}
```

The body ends with the \end{document} command, which signals the end of the document.

4 Document Structure

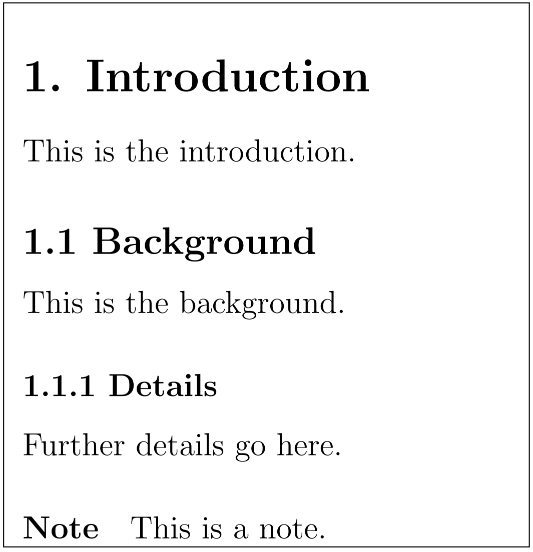
4.1 Basic Structure

A LaTeX document is organized using sections, subsections, and paragraphs. Here's a quick example:

Output:

LaTeX Code:

```latex
\section{Introduction}
This is the introduction.

\subsection{Background}
This is the background.

\subsubsection{Details}
Further details go here.

\paragraph{Note} This is a note.

```



1. Introduction

This is the introduction.

1.1 Background

This is the background.

1.1.1 Details

Further details go here.

Note This is a note.

4.2 Lists

Unordered List:

```latex
\begin{itemize}
    \item First item
    \item Second item
\end{itemize}
```

Output:

- First item
- Second item

Ordered List:

```latex
\begin{enumerate}
    \item First item
    \item Second item
\end{enumerate}
```

Output:

1. First item
2. Second item

5 Mathematical Typesetting

5.1 Inline Math

Inline math is simple to include. For example, the equation of a line can be written as follows:

```latex
The equation of a line is $y = mx + c$.
```

Output:

The equation of a line is $y = mx + c$.

5.2 Displayed Equations

For more complex math that needs its own line, use displayed math:

```latex

$$ E = mc^2 $$
```

Output:

$$E = mc^2$$

© 2024 Nitesh Kumar. All rights reserved.

5.3 Complex Equations

Integrals can be written as:

```
$$ \int_a^b f(x) dx $$
```

Output:

$$\int_a^b f(x)dx$$

For more advanced math, use the `amsmath` package:

```
\documentclass{article}
\usepackage{amsmath}
% other packages in preamble

\begin{document}

% Your code


\end{document}
```

The `amsmath` package allows for advanced mathematical formatting. After including this package, you can use environments like `align`, `gather`, and more.

Complex Equations Using the `align` Environment

`align`: The `align` environment is used for aligning equations at the equal sign or other relation symbols:

LaTeX Code:

```
\begin{align}
    \int_0^{\infty} e^{-x} \, dx &= 1 \\
    \frac{d}{dx}(x^2) &= 2x \\
    \lim_{x \to 0} \frac{\sin x}{x} &= 1 \\
    e^{i\pi} + 1 &= 0
\end{align}
```

Output:

$$\begin{aligned}
\int_0^{\infty} e^{-x} \, dx &= 1 && (1) \\
\frac{d}{dx}(x^2) &= 2x && (2) \\
\lim_{x \to 0} \frac{\sin x}{x} &= 1 && (3) \\
e^{i\pi} + 1 &= 0 && (4)
\end{aligned}$$

© 2024 Nitesh Kumar. All rights reserved.

Complex Equations Using the `gather` Environment

The following equations include integrals, differentiation, and other mathematical symbols:

```
\begin{gather}
    \int_{a}^{b} f(x) \, dx = F(b) - F(a) \\
    \frac{d^2y}{dx^2} + p\frac{dy}{dx} + qy = 0 \\
    \sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6} \\
    \sqrt{a^2 + b^2} = c
\end{gather}
```

Output:

$$\int_{a}^{b} f(x) \, dx = F(b) - F(a) \tag{5}$$

$$\frac{d^2y}{dx^2} + p\frac{dy}{dx} + qy = 0 \tag{6}$$

$$\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6} \tag{7}$$

$$\sqrt{a^2 + b^2} = c \tag{8}$$

6 Figures and Tables

6.1 Inserting Figures

To include images in your LaTeX document, you need to use the `graphicx` package, which provides commands for handling graphics and images.

6.1.1 Include the `graphicx` Package

First, ensure you have the following line in the preamble of your document:

```
\usepackage{graphicx}
```

6.1.2 Inserting a Figure

To insert a figure, you use the `figure` environment. Below is the basic syntax:

```
\begin{figure}[h!]
    \centering
    \includegraphics[width=0.75\textwidth]{image.png}
    \caption{Sample Image}
    \label{fig:image1}
\end{figure}
```

- **Figure Environment**: The `figure` environment is a floating container for figures, which allows LaTeX to place the figure at an optimal location in the document. The

© 2024 Nitesh Kumar. All rights reserved.

optional argument [h!] suggests that LaTeX should place the figure "here," but it can be overridden to maintain document flow.

- \centering: This command centers the figure within the figure environment.
- \includegraphics: This command is used to include the actual image file. The width parameter can be specified as a relative value (e.g., 0.75\textwidth to make the image three quarter the width of the text area) or as an absolute dimension.
- \caption: This command provides a caption for the figure that appears below the image, helping to explain or describe it.
- \label: This command creates a reference label for the figure, allowing you to refer to it elsewhere in your document using \ref{fig:image1}.

Output:

Figure 1: UPES

We can refer Figure 1 anywhere in the document using it's label.

Important Notes

- Make sure that the image file (e.g., image.png) is in the same directory as your .tex file for it to be displayed correctly.
- Adjust the width parameter as needed to fit your document layout.
- The figure environment allows LaTeX to manage the placement of the image, which might not always be exactly where you placed the code, depending on the surrounding content. Use placement options like [h!], [t], [b], or combinations to suggest preferred placement.

6.2 Tables

Tables can be created using the tabular environment, which provides a flexible way to arrange data in rows and columns. The structure of a table is defined using a combination of alignment specifiers, formatting commands, and optional features.

© 2024 Nitesh Kumar. All rights reserved.

6.2.1 Basic Structure

The basic structure of a table consists of the following components: - The `table` environment, which allows for the placement of the table in a floating manner. - The `tabular` environment, which defines the actual content of the table.

Here is an example of a simple table:

```
\begin{table}[h!]
    \centering
    \begin{tabular}{|c|c|c|}
        \hline
        A & B & C \\
        \hline
        1 & 2 & 3 \\
        4 & 5 & 6 \\
        \hline
    \end{tabular}
    \caption{Sample Table}
    \label{tab:table1}
\end{table}
```

Output:

A	B	C
1	2	3
4	5	6

Table 1: Sample Table

6.2.2 Components of the Table Example

- `table`[h!]: This environment wraps the table and allows it to float in the document. The optional argument `[h!]` suggests placing the table "here" if possible.
- \\`centering`: Centers the table on the page.
- `tabular`{|c|c|c|}: This command defines the table structure. The | character adds vertical lines between the columns, and `c` denotes center alignment for each column. You can also use `l` for left alignment and `r` for right alignment.
- \\`hline`: Inserts a horizontal line in the table, creating a clear separation between rows.
- `A & B & C` \\\\: This line specifies the first row of the table. The ampersand `&` separates the columns, and the double backslash \\\\ indicates the end of the row.
- \\`caption`{`Sample Table`}: Provides a caption for the table that appears above or below the table, depending on the document class and settings.
- \\`label`{`tab:table1`}: This command creates a reference label for the table, allowing you to refer to it elsewhere in the document using \\`ref`{`tab:table1`}.

© 2024 Nitesh Kumar. All rights reserved.

6.2.3 Customizing Tables

You can customize tables in several ways:

1. **Changing Column Widths**: You can adjust the width of columns using the `pwidth` specifier instead of `c`, `l`, or `r`. For example, `p3cm` sets a fixed width for a column:

```
   \begin{tabular}{|p{3cm}|p{3cm}|p{3cm}|}
   \hline
   Long text in a column & Another column & More text \\
   \hline
   \end{tabular}

```

   Output:

Long text in a column	Another column	More text

   Table 2: My Caption

2. **Merging Cells**: To merge cells horizontally, you can use the `\multirow` or `\multicolumn` commands from the `multirow` package. Example of merging two columns:

```
\usepackage{multirow} % IN PREAMBLE (BEFORE \begin{document})

   \begin{table}[!h]
       \centering
          \begin{tabular}{|c|c|}
           \hline
           \multicolumn{2}{|c|}{Merged Cell} \\
           \hline
           1 & 2 \\
           3 & 4 \\
           \hline
           \end{tabular}
       \caption{Merged cells table}
       \label{tab:my_label_2}
   \end{table}

```

   Output:

Merged Cell	
1	2
3	4

   Table 3: Merged cells table

3. **Adding Borders and Color**: You can enhance the appearance of tables using the `booktabs` package, which provides commands like `\toprule`, `\midrule`, and `\bottomrule` for cleaner horizontal lines:

© 2024 Nitesh Kumar. All rights reserved.

```latex
\usepackage{booktabs} % IN PREAMBLE (BEFORE \begin{document})

    \begin{table}[!h]
        \centering
        \caption{Tables using booktabs.}
        \begin{tabular}{ccc}
            \toprule
            A & B & C \\
            \midrule
            1 & 2 & 3 \\
            4 & 5 & 6 \\
            \bottomrule
            \end{tabular}
        \label{tab:my_label_3}
    \end{table}


```

Output:

Table 4: Tables using booktabs.

A	B	C
1	2	3
4	5	6

6.2.4 Example of a More Complex Table

Here's a more complex example that includes merged cells and custom widths:

```latex
\begin{table}[h!]
\centering
    \begin{tabular}{|p{4cm}|p{4cm}|c|}
    \hline
    \multicolumn{2}{|c|}{Combined Columns} & Single Column \\
    \hline
    Item 1 & Item 2 & Item 3 \\
    \hline
    \end{tabular}
\caption{Complex Table Example}
\label{tab:complex_table}
\end{table}
```

Output:

Combined Columns		Single Column
Item 1	Item 2	Item 3

Table 5: Complex Table Example

© 2024 Nitesh Kumar. All rights reserved.

7 Cross-referencing and Bibliography

LaTeX provides powerful tools for cross-referencing and managing bibliographies. These features are particularly useful in larger documents like academic papers, theses, or reports, where you often need to refer to figures, tables, sections, or external references.

7.1 Cross-referencing

Cross-referencing in LaTeX allows you to refer to sections, figures, tables, equations, and more, without hardcoding specific numbers. This way, if your document structure changes, all references update automatically.

To set up a cross-reference, you use the `\label` command to mark a specific element, and then use `\ref` or `\pageref` to refer back to that element.

7.1.1 Cross-referencing Sections

For referencing sections, you can place the `\label` command immediately after the section heading. Here's an example:

```latex
\section{Introduction}\label{sec:intro}

This is the introduction to the paper.

\section{Methodology}\label{sec:method}

As discussed in Section \ref{sec:intro}, the problem is defined ...
```

Expected Output
As discussed in Section 1, the problem is defined...

Here, `\ref{sec:intro}` automatically inserts the section number ("1" in this case) into the text. If the order of sections changes, the reference will update to reflect the new numbering.

7.1.2 Cross-referencing Figures and Tables

Cross-referencing is also helpful for figures and tables. Here's an example for referencing a figure:

```latex
\begin{figure}[h!]
    \centering
    \includegraphics[width=0.5\textwidth]{example-image}
    \caption{An example image.}
    \label{fig:image1}
\end{figure}

As shown in Figure \ref{fig:image1}, the result is clear.
```

Expected Output
As shown in Figure 1, the result is clear.

In this case, the \label command inside the figure environment allows you to reference it using \ref{fig:image1}. The output will insert the figure number (”1” in this case) automatically.

You can similarly cross-reference tables by labeling them within the table environment:

```
\begin{table}[h!]
    \centering
    \begin{tabular}{|c|c|}
        \hline
        Item & Description \\
        \hline
        A & Example A \\
        B & Example B \\
        \hline
    \end{tabular}
    \caption{Example Table.}
    \label{tab:example}
\end{table}

Table \ref{tab:example} shows the details of items.
```

Expected Output
Table 1 shows the details of items.

7.1.3 Cross-referencing Pages

To refer to the page where an element appears, use the \pageref command. This is useful for long documents where you want to direct readers to the exact page of a figure, table, or section:

```
Figure \ref{fig:image1} is found on page \pageref{fig:image1}.
```

Expected Output
Figure 1 is found on page 2.

This command inserts the page number of the referenced element.

7.2 Bibliography

LaTeX is widely used in academic writing due to its excellent citation and bibliography management. LaTeX works with tools like BibTeX and BibLaTeX to handle references efficiently. BibTeX allows you to manage and format bibliographic data separately, while BibLaTeX provides more flexibility and modern features.

7.2.1 Basic Bibliography Using BibTeX

To use BibTeX, create a ‘.bib‘ file containing your references. In the main LaTeX file, include the following commands to generate the bibliography:

```
\bibliographystyle{plain}
\bibliography{references}
```

© 2024 Nitesh Kumar. All rights reserved.

Here, `plain` is the style of the bibliography, and `references` is the name of your bibliography file (e.g., `references.bib`).

Your `.bib` file might look like this:

```
@book{lamport1994latex,
  title={LaTeX: A Document Preparation System},
  author={Lamport, Leslie},
  year={1994},
  publisher={Addison-Wesley}
}

@article{knuth1984texbook,
  title={The TeXbook},
  author={Knuth, Donald},
  journal={Computers \& Typesetting},
  volume={A},
  year={1984},
  publisher={Addison-Wesley}
}
```

When compiling your document, BibTeX automatically formats and adds the references at the end of your document. Citations can be added using the `\cite` command:

```
According to \cite{lamport1994latex}, \LaTeX{} is a powerful tool for document preparation.
```

Expected Output

According to [1], LaTeX is a powerful tool for document preparation.

At the end of your document, BibTeX generates the bibliography:

References

[1] Lamport, Leslie. LaTeX: A Document Preparation System. Addison-Wesley, 1994.

[2] Knuth, Donald. The TeXbook. Addison-Wesley, 1984.

7.2.2 Bibliography Using BibLaTeX

BibLaTeX is an advanced package for managing citations and bibliographies. To use it, load the package and specify the backend (e.g., biber):

```
\usepackage[backend=biber,style=numeric]{biblatex}
\addbibresource{references.bib}
```

This setup allows for more flexible citation styles, including numeric, alphabetic, author-year, and more. You can then cite sources using the `\cite` command just like in BibTeX:

```
\cite{knuth1984texbook}
```

At the end of the document, print the bibliography using:

```
\printbibliography
```

© 2024 Nitesh Kumar. All rights reserved.

7.2.3 Citation Styles

Both BibTeX and BibLaTeX offer various citation styles. Common ones include:

- `plain`: Simple numbered style.
- `alpha`: Citations are based on authors' initials and publication year.
- `ieeetr`: IEEE citation style, commonly used in technical and engineering fields.
- `apalike`: APA-style citations, widely used in social sciences.

For example, to use the APA-like citation style:

```
\bibliographystyle{apalike}
```

Or, with BibLaTeX:

```
\usepackage[style=apa]{biblatex}
```

These commands will automatically format your citations and bibliography according to the selected style.

8 Customizing LaTeX Documents

Customization in LaTeX allows you to modify the appearance of your document to meet various formatting requirements. In this section, we'll cover some essential aspects of page layout, fonts, and text styles, all of which can be easily adjusted to suit your needs.

8.1 Page Layout

The page layout in a LaTeX document, such as paper size, margins, and orientation, can be customized using the `geometry` package. This package provides flexibility in adjusting the dimensions of the page to fit specific formatting needs.

To change the paper size and margins, you can specify options directly when loading the `geometry` package. Here's an example for setting A4 paper size and 1-inch margins:

```
\usepackage[a4paper, margin=1in]{geometry}
```

8.1.1 Customizing Margins

You can also specify custom margins for different sides of the page. For example, to set a 2-inch top margin, 1-inch bottom margin, 1.5-inch left margin, and 1-inch right margin, use:

```
\usepackage[top=2in, bottom=1in, left=1.5in, right=1in]{geometry}
```

© 2024 Nitesh Kumar. All rights reserved.

8.1.2 Changing Paper Size and Orientation

To change the paper size to legal (8.5 x 14 inches) and make it landscape oriented, you can modify the options as follows:

```latex
\usepackage[legalpaper, landscape, margin=1in]{geometry}
```

Output: The page will be oriented horizontally (landscape) on legal-sized paper with 1-inch margins.

These changes will apply globally across the entire document unless you specify otherwise.

8.2 Font and Style

Font customization in LATEX is managed by various packages, such as `fontenc` for encoding and `inputenc` for character sets. Changing fonts and text styles can improve readability and give your document a personalized look.

8.2.1 Font Encoding

Using the `fontenc` package ensures that fonts are properly encoded. For example, to enable T1 encoding, which allows for proper hyphenation and accented characters, use:

```latex
\usepackage[T1]{fontenc}
```

T1 encoding is essential when working with European languages or documents requiring accented characters.

8.2.2 Changing Fonts

You can change the font family to one of LATEX's default font families, such as `serif`, `sans-serif`, or `monospace`, using the following commands:

```latex
\renewcommand{\familydefault}{\sfdefault}  % Sans-serif as
   default
```

To use a specific font, such as the popular `Times New Roman`, you can load the corresponding package:

```latex
\usepackage{times}  % Times New Roman
```

Output:
The entire document's font will switch to Times New Roman.
Other common font packages include:

1. `helvet` for Helvetica (sans-serif)
2. `courier` for Courier (monospace)

8.2.3 Text Styles

In LATEX, text styles such as bold, italic, and underlined text are easily applied using the following commands:

1. **Bold Text:** Use `\textbf{...}` to make text bold.

© 2024 Nitesh Kumar. All rights reserved.

```latex
This is \textbf{bold text}.
```

Output:

This is **bold text**.

2. **Italic Text:** Use `\textit{...}` to italicize text.

```latex
  This is \textit{italic text}.
```

Output:

This is *italic text*.

3. **Underlined Text:** While LaTeX doesn't have a direct underline command, you can use the `ulem` package to underline text:

```latex
  \usepackage{ulem}
  This is \uline{underlined text}.
```

Output:

This is underlined text.

Alternatively, for a simpler underlining solution, you can use the `underline` command from standard LaTeX:

```latex
  This is \underline{underlined text}.
```

8.2.4 Customizing Font Sizes

Font size can be adjusted globally or locally within the document. To set the font size for the entire document, modify the document class as follows:

```latex
\documentclass[12pt]{article}
```

This will set the default font size to 12 points.
For local font size adjustments, use the following commands within the document:

- `\tiny`: Very small text
- `\scriptsize`: Smaller than small
- `\footnotesize`: Slightly larger than `scriptsize`
- `\small`: Small text
- `\large`: Slightly larger text
- `\Large`, `\LARGE`: Progressively larger text
- `\huge`, `\Huge`: Very large text

Example:

```
This is \tiny{tiny text}, and this is \Huge{huge text}.
```

Output:

This is tiny text, and this is huge text.

By combining these commands, you can customize the look and feel of your document, ensuring it matches specific formatting guidelines or personal preferences.

8.3 Color and Highlighting

In LaTeX, the `xcolor` package is commonly used to apply colors to text and other elements. You can highlight important sections, change font colors, and even define your own custom colors.

To load the `xcolor` package:

```
\usepackage{xcolor}
```

8.3.1 Changing Text Color

To change the color of specific text, use the `\textcolor` command. Here's an example that sets the text color to red:

```
This is \textcolor{red}{red text}.
```

Output:

This is red text.

8.3.2 Highlighting Text

You can also highlight text with a background color using the `\colorbox` command:

```
\colorbox{yellow}{This text is highlighted in yellow.}
```

Output:

This text is highlighted in yellow.

Customizing LaTeX documents provides a great deal of flexibility in terms of page layout, fonts, text styles, and colors. Using the `geometry` package, you can control the page dimensions and margins. Text appearance can be easily managed with font encodings, font family selection, and local style adjustments like bold, italics, and color. Together, these tools allow you to craft a professional and visually appealing document.

9 Error Handling and Debugging

When working with LaTeX, errors can arise during the compilation process. Understanding common error messages and how to resolve them is essential for smooth document preparation. LaTeX editors, such as Overleaf, TeXShop, or TeXworks, provide detailed logs that can help you trace and fix errors.

In this section, we'll look at common errors, their causes, and strategies for debugging.

© 2024 Nitesh Kumar. All rights reserved.

9.1 Common LaTeX Errors

Here are some of the most common errors you might encounter when compiling a LaTeX document:

9.1.1 Missing or Mismatched Braces

One of the most frequent errors is missing or unmatched braces (i.e., {...}). Every opening brace { must have a corresponding closing brace }.

Example Error:

```latex
This is an \textbf{example of missing brace.
```

The error message may look like this:

```
! LaTeX Error: \textbf on input line 1 ended by \end{document}.
```

Solution: Ensure that every { has a matching }. The correct syntax is:

```latex
This is an \textbf{example of correct brace}.
```

If you're dealing with nested braces, carefully check that each pair is properly closed.

9.1.2 Undefined References

Undefined references occur when you try to reference a section, figure, table, or citation that has not been labeled correctly or is missing entirely. You will see a warning like this during compilation:

```
LaTeX Warning: There were undefined references.
```

Example Error:

```latex
As shown in Figure \ref{fig:missing}, the results are clear.
```

If no figure with the label `fig:missing` exists, you'll get an error.

Solution: Ensure that you have labeled the element you're referencing. For example:

```latex
\begin{figure}[h!]
    \includegraphics[width=0.5\textwidth]{example-image}
    \caption{An example image.}
    \label{fig:image1}
\end{figure}

As shown in Figure \ref{fig:image1}, the results are clear.
```

Also, make sure to run multiple compilation steps (e.g., in Overleaf, press ”Recompile” twice) to resolve cross-references.

© 2024 Nitesh Kumar. All rights reserved.

9.1.3 Package Errors

Using incorrect or incompatible packages can lead to compilation errors. This happens if a required package is missing from your LaTeX installation or if two packages conflict with each other.

Example Error: If you try to load a non-existent package:

```
\usepackage{nonexistent}
```

You will see an error message like this:

```
! LaTeX Error: File ‘nonexistent.sty’ not found.
```

Solution: Ensure that the package you’re trying to use is installed or available in your LaTeX distribution. For example, replace it with a valid package:

```
\usepackage{graphicx}  % A valid package
```

For package conflicts, try commenting out one of the conflicting packages or look for a compatible alternative.

9.2 Debugging Tips

Here are some strategies to debug your LaTeX documents effectively:

9.2.1 Read the Log File

Most LaTeX editors provide a detailed log file that lists all warnings and errors encountered during compilation. This log can help pinpoint the exact line where the error occurred. The log will often include the following types of messages:

- **Error messages**: These are critical and stop the compilation.
- **Warnings**: These indicate potential issues but do not stop the compilation.
- **Overfull/Underfull boxes**: These warn about text that overflows or does not properly fit in the margins.

To view the log, look for the ”Log” or ”Compiler” section in your editor. In Overleaf, for instance, the log is displayed in a separate window after compilation.

9.2.2 Isolate the Problem

If you’re facing a complex issue and cannot locate the source of the error, try commenting out large sections of your document. You can use the % symbol to comment out lines of text or code temporarily:

```
%\section{Introduction}
%This section is commented out to isolate the problem.
```

Once you’ve isolated the error, you can start uncommenting sections one by one to find the problematic code.

© 2024 Nitesh Kumar. All rights reserved.

9.2.3 Check for Typos in Labels

Typographical errors in labels are a common cause of undefined references. Double-check that your labels are spelled correctly and match the references exactly. LaTeX is case-sensitive, so {`fig:image1`} and {`fig:Image1`} will be treated as different labels.

9.2.4 Run Multiple Compilation Passes

When using cross-references or bibliographies (especially with BibTeX or BibLaTeX), LaTeX often requires multiple compilation passes to resolve all references. You may need to run:

1. `pdflatex`
2. `bibtex`
3. `pdflatex` (twice more)

This ensures that all citations and references are updated correctly.

9.3 Warnings

While warnings do not stop compilation, they can indicate formatting problems or overlooked issues. Some common warnings include:

9.3.1 Overfull or Underfull Boxes

These occur when text exceeds the margins (overfull) or does not fill the available space properly (underfull). The message may look like this:

```
Overfull \hbox (5.0pt too wide) in paragraph at lines 22--23
```

Solution: Adjust the text, font size, or use the `\sloppy` command to relax the formatting rules.

9.3.2 Undefined Citations

If a citation is not defined in the bibliography, you'll see a warning like:

```
LaTeX Warning: Citation ‘key’ on page 3 undefined.
```

Solution: Ensure that the citation key matches the reference in your `.bib` file.

9.4 Tools for Error-Free LaTeX

Here are some tools and techniques that can help you avoid and fix errors in LaTeX documents:

© 2024 Nitesh Kumar. All rights reserved.

9.4.1 Online Editors

Using online editors like Overleaf can make error handling easier since they offer real-time error messages and logs. Overleaf, for example, highlights errors and warnings as you type, making it easier to identify issues immediately.

9.4.2 lacheck and chktex

These are command-line tools designed to check LaTeX documents for common errors and potential formatting issues. `lacheck` checks the syntax of your document, while `chktex` focuses on typographical issues.

Error handling and debugging are crucial aspects of working with LaTeX. By understanding common errors, reading logs carefully, and using debugging strategies, you can efficiently resolve issues and ensure smooth compilation. With the right tools and techniques, error-free LaTeX documents are easy to achieve.

10 Title Page and Its Customization in LaTeX

The title page is the first page of a LaTeX document, serving as the cover for your work. It typically includes the title of the document, the author's name, the institution, the date, and sometimes additional information like the course name or the supervisor's name. Customizing the title page can help create a professional and polished look for your document.

10.1 Basic Title Page

To create a basic title page in LaTeX, you can use the `\title`, `\author`, and `\date` commands, followed by the `\maketitle` command. Here's a simple example:

```latex
\documentclass{article}

\title{The Title of Your Document}
\author{Your Name}
\date{\today}  % Automatically inserts today's date

\begin{document}

\maketitle  % Generates the title page

\end{document}
```

10.2 Customizing the Title Page

- **Changing Fonts and Sizes:** You can customize the font size and style of the title, author, and date by using font commands. For example:

```latex
\title{\huge \textbf{The Title of Your Document}}
\author{\Large Your Name}

```

© 2024 Nitesh Kumar. All rights reserved.

- **Adding a Logo:** If you want to include a logo (e.g., your institution's logo), you can use the `graphicx` package:

```latex
\usepackage{graphicx}

\title{\includegraphics[width=0.5\textwidth]{logo.png}\\[1em] \Huge \textbf{The Title of Your Document}}

```

- **Customizing Layout:** To further customize the layout of the title page, you can create your own title page using the `titlepage` environment. This allows more flexibility in positioning elements:

```latex
\begin{titlepage}
    \centering
    \vspace*{2cm}  % Adds vertical space
    {\Huge \textbf{The Title of Your Document}}\\[1.5cm]
    {\Large Your Name}\\
    {\large Institution Name}\\
    {\large \today}\\[2cm]
    \includegraphics[width=0.3\textwidth]{logo.png}\\[1cm]
    {\large Course Name}\\
    {\large Supervisor Name}
    \vfill
\end{titlepage}

```

- **Using Packages:** You can also explore packages like `titling` or `fancyhdr` for more advanced customization of the title page and headers/footers.

10.3 Example of a Customized Title Page

Here's a complete example with a customized title page:

```latex
\documentclass{article}
\usepackage{graphicx}

\title{\huge \textbf{The Title of Your Document}}
\author{\Large Your Name}
\date{\today}

\begin{document}

\begin{titlepage}
    \centering
    \vspace*{2cm}
    \includegraphics[width=0.3\textwidth]{logo.png}\\[1.5cm]
    {\Huge \textbf{The Title of Your Document}}\\[1.5cm]
    {\Large Your Name}\\[0.5cm]
    {\large Institution Name}\\[1.5cm]
    {\large \today}\\[2cm]
    {\large Course Name}\\
```

© 2024 Nitesh Kumar. All rights reserved.

```
    {\large Supervisor Name}
    \vfill
\end{titlepage}

\end{document}
```

Customizing the title page in LaTeX is straightforward, allowing you to create a visually appealing introduction to your document. With simple commands and environments, you can adjust the layout, include graphics, and modify text styles to match your preferences or institutional requirements.

Further Resources

- Overleaf LaTeXDocumentation: https://www.overleaf.com/learn/latex/Main_Page
- CTAN (Comprehensive TeX Archive Network): https://ctan.org/

© 2024 Nitesh Kumar. All rights reserved.