Getting Started with LaTeX

Your First LaTeX Document

Last Updated: January 5, 2026

Introduction

Welcome to LaTeX! This document will teach you the fundamental skills needed to create professional-looking documents. Each section includes code examples and their corresponding output.

Important

After reviewing this guide, open the `first_latex` folder to practice what you've learned. All examples here can be tested in that working directory.

Part 1: Essential Packages and Preamble Setup

Before we start writing content, it's important to understand what packages you need and why. A **package** is an extension that adds functionality to LaTeX.

Understanding the Preamble

The **preamble** is everything between `\documentclass` and `\begin{document}`. It defines your document's style and loads required packages.

Basic Preamble Example

```
\documentclass[11pt]{article}

% Load packages (these add features to LaTeX)
\usepackage[margin=1in]{geometry}        % Page margins
\usepackage{hyperref}                    % Clickable links
\usepackage{graphicx}                    % Insert images
\usepackage{amsmath, amssymb}            % Math equations
\usepackage{array}                       % Advanced tables
\usepackage[style=apa]{biblatex}         % APA citations

\addbibresource{references.bib}          % Link to bibliography file

\begin{document}
  % Your document content goes here
\end{document}
```

Key Packages Explained

- `geometry` — Controls page margins and layout
- `hyperref` — Enables clickable links and URLs
- `graphicx` — Allows you to insert images
- `amsmath, amssymb` — Advanced mathematical typesetting
- `array` — Enhanced table formatting
- `biblatex` — Professional citation and bibliography management (APA/MLA styles)
- `xcolor, tcolorbox` — Colors and colored boxes (for highlighting)

Best Practice: Separate Your Preamble

To keep documents organized, store your preamble in a separate file called `preamble.tex` and import it:

Recommended Structure

```
% In main.tex:
\documentclass[11pt]{article}
\input{preamble}  % Loads preamble.tex

\begin{document}
  Your content here...
\end{document}
```

Part 2: Formatting Text

Making Text Bold

To make text bold, use the `\textbf{}` command.

Code Example

```
This is normal text, but \textbf{this part is bold}.
```

Output: This is normal text, but **this part is bold**.

You can also use `\textit{}` for *italics* and `\texttt{}` for `monospace text`.

Centering Text

Use the `center` environment to center text on the page.

Code Example

```
\begin{center}
    This text is centered on the page.
    \vspace{0.5em}
    {\Large This is large centered text}
\end{center}
```

Output:

This text is centered on the page. This is large centered text

Size commands: Use `\small`, `\Large`, `\huge`, etc. to adjust font size.

Changing Fonts

LaTeX allows you to change fonts in several ways. Here are the most common approaches:

Font Family Commands

Change the font for selected text using these commands:

Font Family Commands

```
% Roman (serif) - default
This is \textbf{normal text}.

% Sans serif
{\sffamily This text is in sans serif font.}

% Typewriter (monospace)
{\ttfamily This text is monospace (code-like).}
```

Output:
This is normal text.
This text is in sans serif font.
`This text is monospace (code-like).`

Larger Font Regions

To change the font for larger sections, use:

Font Family Environments

```
\begin{sloppypar}
\sffamily
This entire paragraph is now in sans serif font.
You can type multiple sentences and paragraphs here,
and they will all use the sans serif style until
the group or environment ends.
\end{sloppypar}

Back to normal font.
```

Output:

This entire paragraph is now in sans serif font. You can type multiple sentences and paragraphs here, and they will all use the sans serif style until the group or environment ends.

Back to normal font.

Combining Font Styles

You can combine font families with bold, italic, and size changes:

Combined Font Styles

```
% Sans serif + bold
{\sffamily\bfseries Important Notice in Sans Serif}

% Typewriter + italic
{\ttfamily\itshape code_with_emphasis}

% Size + font family
{\sffamily\Large Large sans serif text}
```

Output:

Important Notice in Sans Serif

code_with_emphasis

Large sans serif text

Font Family Reference

Command	**Font Style**	**Example**
`\rmfamily` (default)	Roman/Serif	Roman text
`\sffamily`	Sans Serif	Sans serif text
`\ttfamily`	Typewriter	`Typewriter text`

Table 1: Table 3: Font Family Commands

Note: These are the three default fonts in LaTeX. To use advanced fonts like Helvetica, Times, or custom fonts, you may need additional packages (like `fontspec`) or compile with `xelatex` or `lualatex`.

Part 3: Headings and Structure

LaTeX automatically numbers headings and provides organization.

Heading Hierarchy

```
\section{Main Heading}           % Level 1
\subsection{Subheading}          % Level 2
\subsubsection{Sub-subheading}   % Level 3

% Use * to remove numbering:
\section*{Unnumbered Heading}
```

Part 4: Creating Tables

Tables in LaTeX require the `tabular` environment. This is best learned by example.

Simple Table Code

```
\begin{center}
  \begin{tabular}{|c|c|c|}
    \hline
    \textbf{Name} & \textbf{Age} & \textbf{City} \\
    \hline
    Alice & 25 & New York \\
    \hline
    Bob & 30 & Los Angeles \\
    \hline
  \end{tabular}
\end{center}
```

Output:

Name	**Age**	**City**
Alice	25	New York
Bob	30	Los Angeles

Understanding Table Syntax

- `{|c|c|c|}` — Three centered columns with borders
- `&` — Separates columns
- \\\\ — Ends a row
- \\`hline` — Horizontal line
- `c` = centered, `l` = left-aligned, `r` = right-aligned

Adding a Caption to a Table

Use the `table` environment with a caption:

Table with Caption

```
\begin{table}[h]
  \centering
  \begin{tabular}{|c|c|}
    \hline
    Item & Price \\
    \hline
    Coffee & \$3.50 \\
    \hline
  \end{tabular}
  \caption{Table 1: Sample Prices}
  \label{tab:prices}
\end{table}
```

Output:

Item	**Price**
Coffee	$3.50

Table 2: Table 1: Sample Prices

Key points:

- Use `\caption{}` to add a caption
- Use `\label{}` to reference the table later with `\ref{tab:name}`
- Captions appear below tables

Using Online Table Generators or AI

Writing LaTeX table syntax by hand can be tedious. You have two excellent alternatives:

Option 1: Online LaTeX Table Generator

Websites like Tables Generator allow you to:

1. Design your table visually in a GUI
2. Add rows, columns, and formatting
3. Auto-generate the LaTeX code
4. Copy and paste directly into your document

This is the fastest way to create complex tables.

Option 2: Use AI (ChatGPT, Claude, etc.)

Ask ChatGPT to generate LaTeX table code:

```
"Create a LaTeX table with 3 columns for Name, Age,
and City. Include sample data for 5 people. Add a
caption 'Table X: Sample Data' and make headers bold."
```

ChatGPT will generate the complete `table` environment with proper formatting that you can copy directly into your document.

Recommendation: For simple tables, write them manually to learn syntax. For complex tables, use an online generator or AI to save time.

Part 5: Inserting Images

To insert an image, use the `graphicx` package and the `\includegraphics{}` command.

Simple Image Code

```
\begin{center}
  \includegraphics[width=3in]{shrek.jpg}
\end{center}
```

Image Parameters

- `width=3in` — Set image width (3 inches)
- `height=2in` — Set image height
- `scale=0.5` — Scale by 50%

Adding a Caption to a Figure

Use the `figure` environment with a caption:

Figure with Caption

```
\begin{figure}[h]
  \centering
  \includegraphics[width=4in]{shrek.jpg}
  \caption{Figure 1: A sample image}
  \label{fig:shrek}
\end{figure}
```

Key points:

- The `[h]` option tells LaTeX to place the figure **here** (approximately)
- `\caption{}` goes **below** the image for figures
- `\label{}` allows you to reference with `\ref{fig:name}`

Figure Placement Options

The parameter in square brackets `[...]` controls where LaTeX attempts to place the figure:

Figure Placement Parameters

```
\begin{figure}[h]    % Here - at approximately this location
  \centering
  \includegraphics[width=3in]{shrek.jpg}
  \caption{Figure placed here}
\end{figure}

\begin{figure}[t]    % Top of page
  \centering
  \includegraphics[width=3in]{shrek.jpg}
  \caption{Figure at top of page}
\end{figure}

\begin{figure}[b]    % Bottom of page
  \centering
  \includegraphics[width=3in]{shrek.jpg}
  \caption{Figure at bottom of page}
\end{figure}

\begin{figure}[p]    % On a page of floats (figures/tables only)
  \centering
  \includegraphics[width=3in]{shrek.jpg}
  \caption{Figure on dedicated float page}
\end{figure}

\begin{figure}[h!]   % HERE! (forces placement, ignores LaTeX's judgment)
  \centering
  \includegraphics[width=3in]{shrek.jpg}
  \caption{Figure forced to this location}
\end{figure}
```

Placement parameter guide:

- `h` — Here (approximately, LaTeX may move it)
- `t` — Top of the page
- `b` — Bottom of the page
- `p` — Dedicated page for floats
- `!` — Override LaTeX's strictness (use with h, t, b, or p)

Pro tip: Combine options like `[htbp]` to give LaTeX multiple acceptable placements. Use `[h!]` only when absolutely necessary, as it can disrupt document flow.

Image Sizing with `\linewidth`

Use `\linewidth` to scale images relative to the page/column width:

Linewidth Sizing

```
% Full width of the page (100%)
\begin{figure}[h]
  \centering
  \includegraphics[width=\linewidth]{shrek.jpg}
  \caption{Image takes full page width}
\end{figure}

% 80% of page width
\begin{figure}[h]
  \centering
  \includegraphics[width=0.8\linewidth]{shrek.jpg}
  \caption{Image is 80% of page width}
\end{figure}

% 50% of page width (2 images side-by-side)
\begin{figure}[h]
  \centering
  \includegraphics[width=0.45\linewidth]{shrek.jpg}
  \includegraphics[width=0.45\linewidth]{shrek.jpg}
  \caption{Two images at 45% each}
\end{figure}

% Specific measurements
\includegraphics[width=4in]{shrek.jpg}     % 4 inches
\includegraphics[width=10cm]{shrek.jpg}    % 10 centimeters
\includegraphics[scale=0.5]{shrek.jpg}     % Scale to 50%
```

Key sizing commands:

- `\linewidth` — Width of the current line/column
- `0.5\linewidth` — Half the page width
- `width=4in` — Specific width (inches)
- `width=10cm` — Specific width (centimeters)
- `scale=0.5` — Scale by percentage (50%)
- `height=2in` — Set height (width scales proportionally)

Line Breaks and Page Breaks

Control spacing and page layout with these commands:

Spacing Commands

```
% Newline (line break within a paragraph)
This is line one. \\
This is line two on a new line.

% Force small space before next line
Some text \newline
Next line with forced break.

% Vertical spacing (adjust gap between elements)
Some content here.
\vspace{0.5cm}    % Add 0.5 cm of vertical space
More content.

% Page break (new page)
This is the end of the first page.
\newpage
This starts on a new page.

% Optional page break (can be ignored if not needed)
\pagebreak
```

Spacing command guide:

- \\ — Line break (end line, stay in same paragraph)
- `\newline` — Line break (alternative to \\)
- `\vspace{1cm}` — Add vertical space (1 cm in this example)
- `\hspace{2cm}` — Add horizontal space
- `\newpage` — Hard page break (starts new page immediately)
- `\pagebreak` — Soft page break (LaTeX may ignore)
- `\clearpage` — Outputs all pending floats, then starts new page

Practical Example: Multi-Figure Layout

Advanced Figure Example

```
\section{Results}

\begin{figure}[h!]
  \centering
  \includegraphics[width=0.6\linewidth]{shrek.jpg}
  \caption{Figure 1: Main result showing Shrek}
  \label{fig:main_result}
\end{figure}

See Figure \ref{fig:main_result} for the main result.

\vspace{0.5cm}

\begin{figure}[h]
  \centering
  \includegraphics[width=0.35\linewidth]{shrek.jpg}
  \hspace{0.05\linewidth}
  \includegraphics[width=0.35\linewidth]{shrek.jpg}
  \caption{Figure 2: Side-by-side comparison}
  \label{fig:comparison}
\end{figure}

\newpage

\section{Discussion}
Discussion of results continues on the new page...
```

LaTeX excels at typesetting mathematics. This section covers all equation environments and numbering options.

Inline Math

For math within text, use single dollar signs (`$...$`):

Inline Equation Code

```
The equation $E = mc^2$ is Einstein's famous formula.
The golden ratio is $\phi = \frac{1 + \sqrt{5}}{2}$.
```

Output: The equation $E = mc^2$ is Einstein's famous formula. The golden ratio is $\phi = \frac{1+\sqrt{5}}{2}$.

Single-Line Displayed Equations

For equations on their own line (unnumbered), use `\[ ... \]`:

Unnumbered Equation

```
\[
  y = mx + b
\]
```

Output:

$$y = mx + b$$

For a numbered equation, use the `equation` environment:

Numbered Equation

```
\begin{equation}
  \int_0^{\infty} e^{-x^2} \, dx = \frac{\sqrt{\pi}}{2}
  \label{eq:gaussian}
\end{equation}
```

Output:

$$\int_0^{\infty} e^{-x^2} \, dx = \frac{\sqrt{\pi}}{2} \tag{1}$$

You can reference this equation later with `\ref{eq:gaussian}` to get: Equation 1.

Multiple Equations: Using `align`

The `align` environment is used for multiple equations that you want aligned (e.g., at the equals sign).

Numbered Align Environment

Each line gets its own number:

Align with Numbers

```
\begin{align}
  a + b &= c \\
  x + y &= z \\
  p + q &= r
\end{align}
```

Output:

$$a + b = c \tag{2}$$

$$x + y = z \tag{3}$$

$$p + q = r \tag{4}$$

Notice the `&` symbol aligns the equals signs, and \\ separates lines.

Unnumbered Align Environment (`align*`)

Use `align*` to suppress numbering on all lines:

Align without Numbers

```
\begin{align*}
  (a + b)^2 &= a^2 + 2ab + b^2 \\
  (a - b)^2 &= a^2 - 2ab + b^2 \\
  a^2 - b^2 &= (a+b)(a-b)
\end{align*}
```

Output:

$$\begin{aligned}(a + b)^2 &= a^2 + 2ab + b^2 \\ (a - b)^2 &= a^2 - 2ab + b^2 \\ a^2 - b^2 &= (a+b)(a-b)\end{aligned}$$

Selective Numbering with `nonumber`

Use `\nonumber` to suppress numbering on specific lines within `align`:

Align with Selective Numbering

```
\begin{align}
  x^2 + y^2 &= z^2 \\
  a + b &= c \nonumber \\
  p &= q
\end{align}
```

Output:

$$x^2 + y^2 = z^2 \tag{5}$$

$$a + b = c$$

$$p = q \tag{6}$$

Only the first and third equations are numbered. The second uses `\nonumber`.

Summary of Equation Environments

Common Math Symbols and Notation

- Superscripts: `$x2̂$` produces x^2
- Subscripts: `$x_i$` produces x_i
- Fractions: `$\frac{a}{b}$` produces $\frac{a}{b}$
- Greek letters: `$\alpha, \beta, \gamma$` produce α, β, γ
- Integrals: `$\int$` produces $\int$

Environment	**Numbered?**	**Best For**
`$...$`	No	Inline math
`\[...\]`	No	Single equation, displayed
`equation`	Yes	Single equation, numbered
`equation*`	No	Single equation (alternative)
`align`	Yes (each line)	Multiple equations, aligned
`align*`	No	Multiple equations, no numbers

Table 3: Table 3: Equation Environment Comparison

- Summation: `$\sum$` produces $\sum$
- Square root: `$\sqrt{x}$` produces $\sqrt{x}$
- Limits: `$\lim_{x \to 0}$` produces $\lim_{x \to 0}$

Part 6: Theorems, Definitions, and Proofs

Mathematical and scientific documents often require formal statements like theorems, definitions, and proofs. LaTeX provides special environments for these through the `amsthm` package.

Setting Up Theorem Environments

First, add to your preamble:

Preamble Setup

```
\usepackage{amsthm}

% Define theorem environments
\theoremstyle{definition}
\newtheorem{theorem}{Theorem}[section]
\newtheorem{definition}[theorem]{Definition}
\newtheorem{lemma}[theorem]{Lemma}
\newtheorem{corollary}[theorem]{Corollary}

\theoremstyle{remark}
\newtheorem*{remark}{Remark}
```

This creates numbered environments that automatically number within each section.

Using the Theorem Environment

Theorem Code

```
\begin{theorem}
  For any right triangle, the sum of the squares of
  the two shorter sides equals the square of the
  hypotenuse.
  \label{thm:pythagoras}
\end{theorem}
```

Output:

Theorem 0.1. For any right triangle, the sum of the squares of the two shorter sides equals the square of the hypotenuse.

Notice LaTeX automatically adds the label “Theorem 1” and numbers it.

Using the Definition Environment

Definition Code

```
\begin{definition}
  A \textbf{prime number} is a natural number greater
  than 1 that has no positive divisors other than
  1 and itself.
\end{definition}
```

Output:

Definition 0.2. A **prime number** is a natural number greater than 1 that has no positive divisors other than 1 and itself.

Using the Lemma Environment

A lemma is a smaller theorem used to prove larger theorems:

Lemma Code

```
\begin{lemma}
  If $n$ is even, then $n^2$ is even.
\end{lemma}
```

Output:

Lemma 0.3. If n is even, then n^2 is even.

Using the Proof Environment

The `proof` environment creates a formal proof with a QED symbol at the end:

Proof Code

```
\begin{proof}
  Suppose $n$ is even. Then $n = 2k$ for some
  integer $k$. Therefore:
  \[
    n^2 = (2k)^2 = 4k^2 = 2(2k^2)
  \]
  Since $n^2$ is a multiple of 2, $n^2$ is even.
\end{proof}
```

Output:

Proof. Suppose n is even. Then $n = 2k$ for some integer k. Therefore:

$$n^2 = (2k)^2 = 4k^2 = 2(2k^2)$$

Since n^2 is a multiple of 2, n^2 is even. □

Unnumbered Environments

Add an asterisk to create unnumbered versions:

Unnumbered Theorem

```
\begin{theorem*}
  This theorem has no number.
\end{theorem*}
```

Custom Theorem Names

Add optional text in brackets to customize the theorem name:

Custom Theorem Name

```
\begin{theorem}[Pythagorean Theorem]
  In a right triangle, $a^2 + b^2 = c^2$.
\end{theorem}
```

Output:

Theorem 0.4 (Pythagorean Theorem)**.** In a right triangle, $a^2 + b^2 = c^2$.

Using Corollary

A corollary is a consequence of a theorem:

Corollary Code

```
\begin{theorem}
  All angles in a triangle sum to $180°$.
\end{theorem}

\begin{corollary}
  Each angle in an equilateral triangle is $60°$.
\end{corollary}
\end{corollary}
```

Output:

Theorem 0.5. All angles in a triangle sum to 180.

Corollary 0.6. Each angle in an equilateral triangle is 60.

Complete Mathematical Paper Example

Full Example

```
\section{Number Theory}

\begin{definition}
  An integer $n$ is \textbf{perfect} if it equals
  the sum of its proper divisors.
\end{definition}

\begin{example}
  The number 6 is perfect: $6 = 1 + 2 + 3$.
\end{example}

\begin{lemma}
  If $n$ is perfect, then $n > 0$.
\end{lemma}

\begin{proof}
  This follows directly from the definition.
\end{proof}

\begin{theorem}[Euclid-Euler Theorem]
  An even perfect number has the form
  $n = 2^{p-1}(2^p - 1)$ where $2^p - 1$ is prime.
\end{theorem}
```

Quick Theorem Reference

Environment	Purpose
theorem	Main mathematical statement
lemma	Helper lemma (numbered)
corollary	Consequence of a theorem
definition	Define a concept
proof	Formal proof (adds QED)
remark	Non-formal comment

Table 4: Table 4: Theorem Environment Types

Part 7: Hyperlinks

The `hyperref` package enables clickable links.

Hyperlink Code

```
\href{https://www.overleaf.com}{Click here to visit Overleaf}

% Link within document
See Section \ref{sec:intro} for the introduction.
```

Output: Click here to visit Overleaf

Key points:

- `\href{URL}{display text}` creates a clickable link
- `\ref{}` references labeled sections, tables, and figures
- `\label{}` must be placed before the reference

Part 8: Citations and Bibliography (APA/MLA Style)

Professional documents require citations. LaTeX uses BibTeX for bibliography management.

Setting Up BibTeX

First, create a `references.bib` file with your sources:

Example references.bib

```
@article{Einstein1905,
  title={On the Electrodynamics of Moving Bodies},
  author={Einstein, Albert},
  year={1905},
  journal={Annalen der Physik}
}

@book{Knuth1984,
  title={The TeXbook},
  author={Knuth, Donald E.},
  year={1984},
  publisher={Addison-Wesley}
}
```

Using Citations in Your Document

Citation Code

```
% In preamble:
\usepackage[style=apa]{biblatex}
\addbibresource{references.bib}

% In document:
According to \cite{Einstein1905}, the theory of
relativity revolutionized physics.

% At the end of document:
\printbibliography
```

Citation styles available:

- `style=apa` — American Psychological Association
- `style=mla` — Modern Language Association
- `style=chicago` — Chicago Manual of Style
- `style=ieee` — IEEE style (for engineering)

Part 9: Using ChatGPT to Convert Word Documents to LaTeX

If you have a document created in Word or another format, ChatGPT can help convert it to LaTeX code.

Process

1. Copy the text from your Word document
2. Open ChatGPT
3. Paste the text and ask: *"Convert this to LaTeX code with proper formatting, tables, and equations"*

4. Copy the generated LaTeX code
5. Paste it into your `.tex` file
6. Adjust and debug as needed

Important Note

ChatGPT output usually requires adjustments. Always review the generated code and verify:

- All packages are included in the preamble
- Syntax is correct
- Formatting matches your needs

Part 10: Labels and References (`\label` and `\ref`)

Labels and references allow you to automatically number and cross-reference figures, tables, equations, and sections throughout your document.

Creating a Label

Any labeled object (section, figure, table, equation) can be referenced later. Use `\label{}` right after creating the object:

Labeling Examples

```
% Labeling a section
\section{My Section}
\label{sec:mysection}

% Labeling a figure
\begin{figure}[h]
  \centering
  \includegraphics[width=4in]{shrek.jpg}
  \caption{A funny character}
  \label{fig:shrek}
\end{figure}

% Labeling a table
\begin{table}[h]
  \centering
  \begin{tabular}{|c|c|}
    \hline
    A & B \\
    \hline
  \end{tabular}
  \caption{Sample Table}
  \label{tab:sample}
\end{table}

% Labeling an equation
\begin{equation}
  E = mc^2
  \label{eq:einstein}
\end{equation}
```

Referencing Objects

Use `\ref{}` to insert the number of the labeled object:

Reference Examples

```
% Reference a figure
As shown in Figure \ref{fig:shrek}, this is a character.

% Reference a table
Table \ref{tab:sample} displays data.

% Reference an equation
The famous equation \ref{eq:einstein} relates
mass and energy.

% Reference a section
See Section \ref{sec:mysection} for more details.
```

Output Example: As shown in Figure 1, this demonstrates the Shrek character.

Figure 1: Shrek - A Famous Animated Character

Key points:

- Labels must come immediately after the object (caption, equation, section title)
- Use descriptive label names: `fig:`, `tab:`, `eq:`, `sec:` prefixes help organize
- `\ref{}` automatically inserts the correct number
- If references show as “??”, compile twice to update references
- Labels only work with numbered objects (sections, tables with captions, figures with captions, equations)

Part 11: Table of Contents

LaTeX can automatically generate a table of contents from your section headings.

Inserting a Table of Contents

Add this command where you want the table of contents to appear (usually after the title page):

Table of Contents Code

```
\begin{document}

\begin{center}
  {\Large \textbf{My Document}}
\end{center}

\tableofcontents          % Insert TOC here
\newpage                  % Start content on new page

\section{Introduction}
Content here...

\section{Methods}
More content...

\end{document}
```

Important: Compile your document **twice** for the table of contents to appear correctly (first pass collects headings, second pass generates the TOC).

Controlling TOC Depth

By default, all sections and subsections appear. To control which levels are included:

Controlling TOC Depth

```
% Show only sections (not subsections)
\setcounter{tocdepth}{1}
\tableofcontents

% Show sections and subsections
\setcounter{tocdepth}{2}
\tableofcontents
```

Part 12: Title Pages

A professional title page includes the document title, author, date, and institution.

Simple Title Page

Simple Title Page

```
\begin{document}

\begin{titlepage}
  \begin{center}
    \vspace*{1cm}

    {\huge\bfseries My Research Paper}\\
    \vspace{0.5cm}
    {\LARGE A Study of Something Important}\\

    \vspace{2cm}

    {\Large By}\\
    {\Large John Doe}\\

    \vspace{2cm}

    {\large Department of Computer Science}\\
    {\large University Name}\\

    \vspace{2cm}

    {\large January 5, 2026}

    \vfill

  \end{center}
\end{titlepage}

\newpage

\tableofcontents
\newpage

\section{Introduction}
% Your content starts here
```

Using the Built-in `\maketitle` Command

Alternatively, use LaTeX's built-in title command in the preamble:

Built-in Title Page

```
% In preamble:
\title{My Research Paper}
\author{John Doe}
\date{January 5, 2026}

% In document:
\begin{document}

\maketitle    % Creates the title page automatically

\tableofcontents
\newpage

\section{Introduction}
% Your content starts here
```

Part 13: Fancy Headers and Page Numbering

The `fancyhdr` package allows custom headers, footers, and page numbering styles.

Setting Up Fancy Headers

Add to your preamble:

Basic Fancy Header Setup

```
\usepackage{fancyhdr}

\pagestyle{fancy}           % Activate fancy style

% Define header content
\lhead{Left Side}           % Left header
\chead{Center}              % Center header
\rhead{Right Side}          % Right header

% Define footer content
\lfoot{Left Footer}         % Left footer
\cfoot{\thepage}            % Center footer (page number)
\rfoot{Right Footer}        % Right footer

\renewcommand{\headrulewidth}{0.4pt}    % Header line
\renewcommand{\footrulewidth}{0.4pt}    % Footer line
```

Example: Document with Custom Headers

Complete Header Example

```
% In preamble:
\usepackage{fancyhdr}

\pagestyle{fancy}
\lhead{\textit{My Document}}
\chead{}
\rhead{Chapter 1}

\lfoot{}
\cfoot{\thepage}
\rfoot{}

% In document:
\begin{document}

\section{Introduction}
Your content here. Notice the headers
and footers on the pages!

\newpage

\section{Methods}
The page numbers automatically update.

\end{document}
```

Page Numbering Styles

Control how page numbers are displayed:

Page Numbering Styles

```
% Arabic numerals (1, 2, 3, ...)
\pagenumbering{arabic}

% Roman numerals (i, ii, iii, ...)
\pagenumbering{roman}

% Capital Roman numerals (I, II, III, ...)
\pagenumbering{Roman}

% Letters (a, b, c, ...)
\pagenumbering{alph}

% Reset page counter
\setcounter{page}{1}
```

Professional Example: Title Page + TOC + Content

Complete Document Structure

```
\documentclass{article}
\usepackage{fancyhdr}

\pagestyle{fancy}
\rhead{My Paper}
\cfoot{\thepage}

\title{Research on Something Important}
\author{Jane Smith}
\date{\today}

\begin{document}

% Title page with roman numerals
\pagenumbering{roman}
\maketitle

% Table of contents
\newpage
\tableofcontents

% Switch to arabic numerals for main content
\newpage
\pagenumbering{arabic}
\setcounter{page}{1}

\section{Introduction}
Main content starts here...

\section{Methods}
More content...

\end{document}
```

Key points:

- Use roman numerals (i, ii, iii) for front matter (title page, TOC, preface)
- Switch to arabic numerals (1, 2, 3) for main content with `\pagenumbering{arabic}`
- `\thepage` inserts the current page number
- Fancy headers are disabled on the first page of sections by default; use `\fancypagestyle{plain}{}` to override

Command	**Purpose**
`\textbf{text}`	Make text bold
`\textit{text}`	Make text italic
`\texttt{text}`	Monospace text
`\section{Title}`	Create a section
`\includegraphics{file}`	Insert image
`\cite{key}`	Cite a reference
`\href{url}{text}`	Create hyperlink
`begin{table}...\end{table}`	Create table
`$...$`	Inline math
`\[...\]`	Display equation

Table 5: Table 2: LaTeX Quick Reference

Your Next Step: Practice

Now that you've learned the fundamentals:

Action Items

1. Open the `first_latex` folder in your editor
2. Create a new document with:
   - A centered title with `\textbf{}`
   - At least two sections
   - One bold paragraph
   - A simple table with a caption
   - An inline equation (e.g., $E = mc^2$)
   - A hyperlink
3. Compile your document and review the output
4. Experiment with different styling options

Advanced Topic: Customization and Modularity in LaTeX

Notice throughout this document that we use a custom command called `\Part{}` to automatically number sections. This demonstrates one of LaTeX's greatest strengths: **customizability and modularity**. Instead of manually typing "Part 1:", "Part 2:", etc., we defined a reusable command that handles numbering automatically.

How the `\Part` Command Works

In the `preamble.tex` file, we added these lines:

Custom Part Counter and Command

```
% Create a new counter
\newcounter{partcounter}

% Set the starting value
\setcounter{partcounter}{0}

% Define a custom command with auto-incrementing counter
\newcommand{\Part}[1]{%
    \stepcounter{partcounter}%
    \section*{Part \thepartcounter: #1}%
}
```

Breaking this down:

- `\newcounter{partcounter}` — Creates a new counter variable
- `\setcounter{partcounter}{0}` — Initializes the counter to 0
- `\newcommand{\Part}[1]` — Defines a command that accepts 1 argument (the title)
- `\stepcounter{partcounter}` — Increments the counter by 1 before each section
- `\thepartcounter` — Displays the current counter value
- `%` — Ends lines to prevent unwanted spaces

Usage Throughout the Document

Instead of writing:

The Old Way (Not Used Here)

```
\section*{Part 1: Essential Packages and Preamble Setup}
  % content
\section*{Part 2: Formatting Text}
  % content
\section*{Part 3: Headings and Structure}
  % content
% ... and so on, manually updating numbers
```

We use the custom command:

The New Way (Used in This Document)

```
\Part{Essential Packages and Preamble Setup}
  % content
\Part{Formatting Text}
  % content
\Part{Headings and Structure}
  % content
% Numbers update automatically!
```

Benefits of This Approach

- **Automatic numbering:** Add, remove, or reorder sections without manual updates
- **Consistency:** All parts use the same formatting automatically
- **Easy maintenance:** Change the formatting once in the preamble; it updates everywhere
- **Modularity:** The preamble is separate from the document content
- **Reusability:** Copy the preamble to new documents and start using `\Part{}` immediately

Creating Your Own Custom Commands

You can create any custom command you need using `\newcommand`:

Custom Command Examples

```
% Command with no arguments
\newcommand{\mytitle}{My Important Title}

% Command with 1 argument
\newcommand{\highlight}[1]{\textbf{\textcolor{red}{#1}}}

% Command with 2 arguments
\newcommand{\myfraction}[2]{\frac{#1}{#2}}

% Use them in your document:
\mytitle              % Outputs: My Important Title
\highlight{Warning!} % Outputs: Warning! in bold red
\myfraction{a}{b}     % Outputs: a/b as a fraction
```

Creating Custom Environments

For larger blocks of content, define custom environments:

Custom Environment Example

```
% In preamble:
\newenvironment{highlight}
  {\begin{tcolorbox}[colback=yellow!20]}
  {\end{tcolorbox}}

% In document:
\begin{highlight}
This text will be in a yellow highlight box!
\end{highlight}
```

Modular Document Structure

For large projects, organize your document into separate files:

Modular Project Example

```
% main.tex
\documentclass{article}
\input{preamble}

\begin{document}

\maketitle

\tableofcontents
\newpage

% Include separate chapter files
\input{chapters/introduction.tex}
\input{chapters/methods.tex}
\input{chapters/results.tex}
\input{chapters/discussion.tex}

\printbibliography

\end{document}
```

Benefits of modularity:

- Keep each chapter in a separate file for easier editing
- Reuse preambles across multiple projects
- Share standardized preambles with collaborators
- Version control is easier with modular structure
- Compile individual chapters for testing

Key Takeaways

LaTeX's power comes from its ability to be customized:

- Define counters for automatic numbering
- Create custom commands with `\newcommand`
- Build reusable environments with `\newenvironment`
- Separate content into modular files
- Use preambles to maintain consistency across documents
- Share and reuse code through packages and templates

As you become more advanced, you'll find that LaTeX allows you to create professional workflows tailored to your specific needs!

- **Compilation error?** Check for missing closing braces }
- **Image not showing?** Verify the filename and file location
- **Bibliography empty?** Make sure you've run `bibtex` or used `biblatex`
- **Math symbols not working?** Ensure you've included `\usepackage{amssymb}`

Good luck with your LaTeX journey!