Practical LaTeX

For Academic and Scientific Documents

A practical guide to writing books, theses, question papers, and research articles

Rajesh Kumar

About This Book

Purpose. This book is a practical guide to building academic and scientific documents in LaTeX. It is not an encyclopedia. It explains the project decisions that make books, theses, question papers, research articles, and Hindi manuscripts easier to write, revise, compile, and reuse.

How to read it. Start with the first two chapters if you are new to LaTeX project structure. Then jump directly to the document type you need: book, thesis, question paper, article, tables and figures, or build repair.

Conventions. Commands appear as `\input` or `\includegraphics`. Package names appear as `geometry`, `fontspec`, or `tabularx`. Generated files belong in `gar/`. Figures use labels such as `fig:geometry-visual`; equations use `eq:...`; tables use `tab:...`.

Principle

The best LaTeX file is not the cleverest file. It is the file that can be opened six months later, understood quickly, compiled reliably, and reused safely.

Chapter	Main use
1	Project shape, packages, fonts, page geometry, and Hindi manuscripts.
2	Long documents: books and theses.
3	Shorter academic documents: question papers and research articles.
4	Tables, figures, labels, and references.
5	Builds, reuse, and error repair.

Contents

CHAPTER 1

Project and Preamble

A serious LaTeX document is a small project. The main file controls the document class, packages, page design, fonts, hyperlinks, and the order of included files. Chapter files should carry the writing. Generated files should stay out of the manuscript folders.

Compile `main.tex` First

Compile the main file, not an individual chapter file. The main file knows the document class, packages, paths, index setup, and output folder.

Engine	Use it when
`pdflatex`	The document uses standard pdfLaTeX packages such as `fontenc`, `inputenc`, and `lmodern`. It is reliable for most English academic documents.
`xelatex`	The document needs system fonts through `fontspec`, Unicode text, or scripts such as Hindi/Devanagari.
`lualatex`	The document needs system fonts like XeLaTeX, with Lua-based extensions or advanced font control.

On macOS or Linux, run these commands from the project directory:

```
mkdir -p gar
pdflatex -interaction=nonstopmode -halt-on-error \
  -output-directory=gar main.tex
makeindex gar/main.idx -o gar/main.ind -t gar/main.ilg
pdflatex -interaction=nonstopmode -halt-on-error \
  -output-directory=gar main.tex
cp gar/main.pdf main.pdf
```

On Windows Command Prompt, use backslashes for files inside `gar`:

```
if not exist gar mkdir gar
pdflatex -interaction=nonstopmode -halt-on-error ^
  -output-directory=gar main.tex
makeindex gar\main.idx -o gar\main.ind -t gar\main.ilg
pdflatex -interaction=nonstopmode -halt-on-error ^
  -output-directory=gar main.tex
copy gar\main.pdf main.pdf
```

If the source uses `fontspec`, replace `pdflatex` with `xelatex` or `lualatex`. The option `-output-directory=gar` keeps auxiliary files such as `.aux`, `.toc`, `.idx`, and `.log` inside `gar/`.

A Project You Can Reopen

Figure 1.1 is the basic habit. The file `main.tex` should read like the source table of contents. It loads the preamble, starts the document, and includes the major

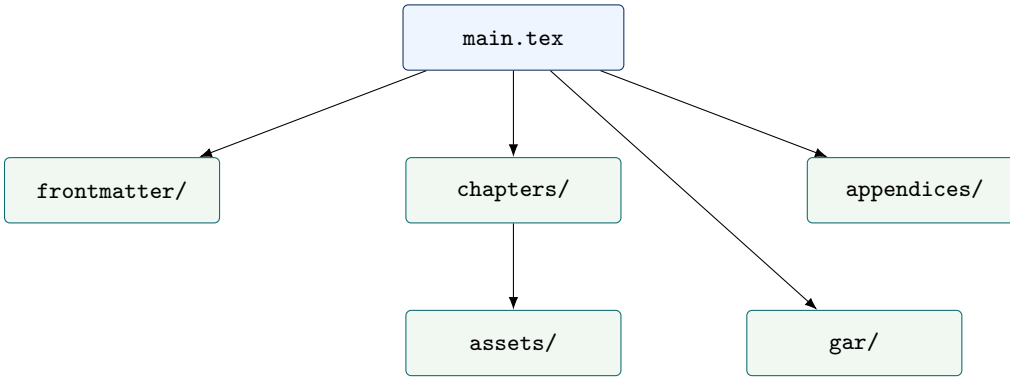

Figure 1.1: A readable project separates control, writing, assets, and generated output.

files in order. The writing folders then stay small enough to browse.

Location	Role
`main.tex`	Class, packages, page layout, document order, index setup, and metadata.
`frontmatter/`	Title, preface, acknowledgements, notation, or a compact opening note.
`chapters/`	Main explanations, one coherent topic per file.
`appendices/`	Reference material that supports the book but should not interrupt the chapters.
`gar/`	Auxiliary files and build output such as `.aux`, `.toc`, `.idx`, and `.log`.

Commands Used Throughout

The book should not re-explain the same command every time it appears. Use this compact command map once, then apply it in context.

Command	Use
`\documentclass`	Selects the broad document type, such as book, article, report, or thesis class.
`\usepackage`	Loads extra capabilities before the document begins.
`\input`	Inserts another source file exactly where the command appears.
`\chapter` and `\section`	Build the visible hierarchy of the document.
`\includegraphics`	Places an image, normally inside a figure environment.
`\label` and `\ref`	Connect numbered items to cross-references that update after recompilation.
`\index`	Marks a word or topic for the generated index.

Packages by Job

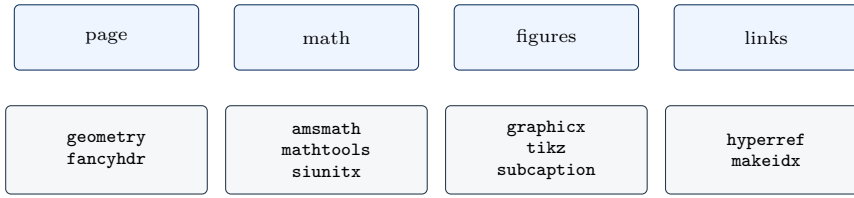

Figure 1.2: Package choices are clearer when each package has a visible job.

`geometry` sets page size and margins. `fontspec` selects system fonts when using XeLaTeX or LuaLaTeX. `amsmath` and `mathtools` improve equations. `siunitx` gives consistent scientific units. `graphicx` inserts images. `subcaption` groups related images. `booktabs` and `tabularx` make readable tables. `tikz` draws diagrams. `hyperref` creates links and PDF metadata.

Page and Font Design

```
\usepackage[
  paperwidth=6in,
  paperheight=9in,
  inner=0.65in,
  outer=0.50in,
  top=0.58in,
  bottom=0.68in
]{geometry}
```

For English documents compiled with pdfLaTeX, `lmodern` is usually enough. For a manuscript that must use a system font, use XeLaTeX or LuaLaTeX with `fontspec`:

```
\usepackage{fontspec}
\IfFontExistsTF{Times New Roman}
  {\setmainfont{Times New Roman}}
  {\setmainfont{TeX Gyre Termes}}
```

Hindi Manuscripts

A Hindi manuscript needs a Unicode engine and a Devanagari-capable OpenType font. The source in `print-hindi/manuscript/book.tex` uses the right pattern: `book`, A5 paper, two-sided margins, `fontspec`, Devanagari font selection, Hindi labels, designed chapter heads, and modular frontmatter/chapter/backmatter files.

```
\setmainfont{Kohinoor Devanagari}[Script=Devanagari]
\setsansfont{Kohinoor Devanagari}[Script=Devanagari]
\linespread{1.16}
\setlength{\parindent}{1.15em}
```

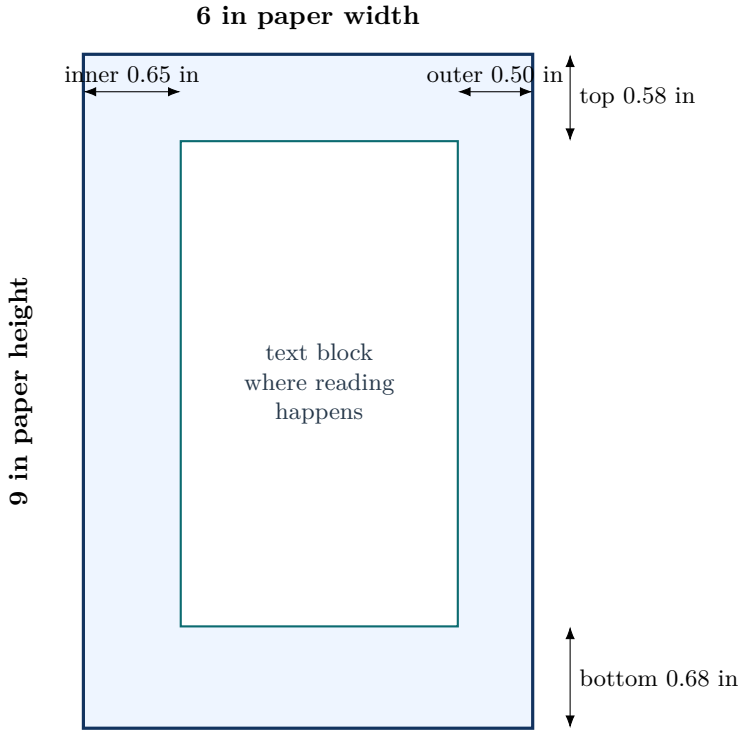

Figure 1.3: The paper is the outer rectangle; the readable text block is inside the margins.

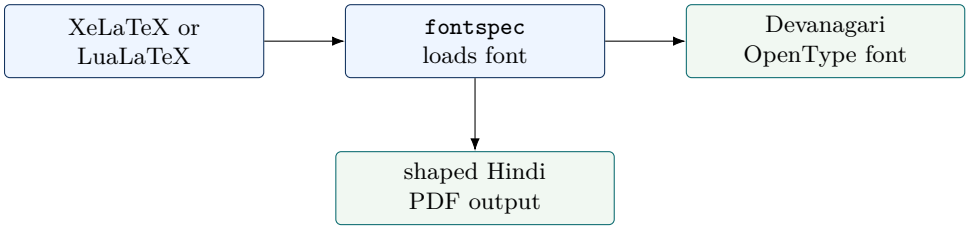

Figure 1.4: Hindi changes the engine and font setup, but not the modular project discipline.

CHAPTER 2

Books and Theses

Books and theses share the same engineering problem: they are long, formal, and revised many times. Their source should make the document order obvious and should keep formal pages separate from the main argument.

The Long-Document Flow

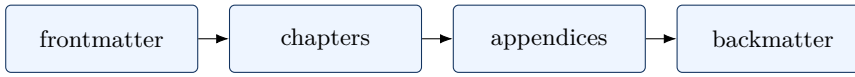

Figure 2.1: Books and theses are easiest to maintain when the source follows the reader's path.

For a book, frontmatter usually means title pages, copyright, preface, and contents. For a thesis, it may also include declarations, certificates, acknowledgements, abstracts, lists of figures, and lists of tables. These formal pages should not be mixed into chapter files.

One Source Pattern

```
\frontmatter
\input{frontmatter/title}
\input{frontmatter/preface}
\tableofcontents

\mainmatter
\chapter{Introduction}
\input{chapters/introduction}

\appendix
\input{appendices/supporting-material}

\backmatter
\printbibliography
\printindex
```

This pattern serves both books and theses. The difference is in the required formal pages, bibliography style, line spacing, fonts, and institutional rules.

Choosing the Document Class

The document class is the first structural decision. It controls the default page style, title behavior, sectioning commands, and whether commands such as `\chapter` exist.

Class	Good use
`book`	Textbooks, monographs, long manuals, and documents with chapters, frontmatter, appendices, and backmatter.
`report`	Theses, project reports, lab reports, and technical documents that need chapters but less book-style frontmatter.
`article`	Papers, question papers, assignments, notes, and short documents without chapters.
`memoir`	Highly customized books where one class should control chapters, headers, captions, and layout.

Do not choose a class by habit. Choose it by the structure the reader will see. If the document needs chapters, use `book` or `report`. If it only needs sections, use `article`.

Book Decisions

A book needs a strong table of contents, consistent chapter openings, useful running heads, and often an index. Packages such as `fancyhdr`, `titlesec`, and `makeidx` support these jobs. Use them to clarify navigation, not to overdecorate the page.

An index is especially useful in a LaTeX manual because readers often return to a command or package name. Mark only useful terms. An index full of obvious words is harder to use than no index.

Chapter Files in a Book

One chapter file should contain one chapter's prose, figures, tables, and local labels. Keep global design outside chapter files. A chapter should not load packages, change the page geometry, or redefine the whole heading style.

```
% main.tex
\chapter{Fourier Series}
\input{chapters/fourier-series}

% chapters/fourier-series.tex
\section{Periodic Functions}
Text, equations, figures, and tables for this chapter.
```

Good chapter files begin with the reader's problem, then introduce notation, examples, figures, and exercises. A folder full of small chapter files is easier to revise than one large manuscript file with thousands of lines.

Thesis Decisions

A thesis must satisfy institutional rules. That usually affects page size, margins, font, line spacing, title pages, certificate pages, bibliography style, and annexures. Use `fontspec` only when a system font is required; otherwise pdfLaTeX with `lmodern` is simpler.

Use `biblatex` with `biber` when the bibliography needs modern sorting and citation control. Use `pdfpages` when signed certificates, published papers, or official reports must be included as PDF pages.

Typical Thesis Order

Most thesis problems come from putting formal pages in the wrong place. A practical order is:

1. Title page, certificate, declaration, acknowledgement, and abstract.
2. Table of contents, list of figures, and list of tables.
3. Introduction, literature review, method, results, discussion, and conclusion.
4. Appendices, bibliography, and annexures.

Use Roman page numbers for formal frontmatter when the institution requires it. Start Arabic numbering at Chapter 1. Keep signed certificates and scanned approvals as external PDFs, then include them with `pdfpages` if the final submission requires them.

```
\includepdf[pages=-]{signed-certificate.pdf}
```

Principle

A thesis should feel like one sustained argument. Modular source files help the author; continuous prose helps the examiner.

CHAPTER 3

Question Papers and Articles

Question papers and research articles are shorter than books and theses, but they still need discipline. A question paper must be scanned quickly. A research article must present a compact scholarly argument.

Choose the Form

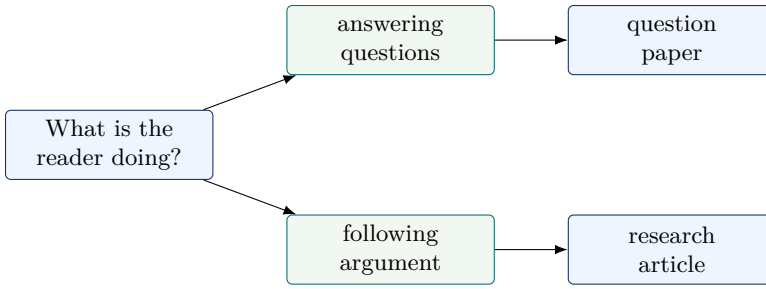

Figure 3.1: The document form follows the reader's task.

Question Papers

The `article` class is usually enough. A question paper needs a header, duration, marks, clear instructions, and questions whose marks are easy to see.

```
\documentclass[12pt]{article}
\usepackage[margin=0.75in]{geometry}
\usepackage{amsmath,array,fancyhdr}
```

Question	Marks
Derive the time period of a simple harmonic oscillator.	5
Explain boundary conditions for a particle in a one-dimensional box.	3
State two assumptions used in the ideal gas model.	2

Use displayed equations when the student must inspect the structure. Keep decoration minimal; an examination page should print cleanly and guide the eye quickly.

Question Paper Anatomy

A question paper is a reading-speed document. Students should find the paper code, time, marks, instructions, and question marks without searching.

Part	What it should contain
Header	Course code, course title, exam name, date, time, and maximum marks.
Instructions	Allowed choices, calculator rules, attempt rules, and assumptions.
Question body	Numbered questions with subparts and visible marks.
Marks column	Right-aligned marks so the examiner and student can scan quickly.
Footer	Page number, paper code, or continuation note when needed.

For physics and mathematics papers, keep variables readable and avoid squeezing long derivations into one line. If the question contains a large expression, display it:

```
\[
  \psi_n(x)=\sqrt{\frac{2}{L}}\sin\left(\frac{n\pi x}{L}\right),
  \qquad 0<x<L.
\]
```

Use `enumerate` when question numbers matter. Use a table only when you need a stable marks column.

Research Articles

A research article needs title, authors, abstract, sections, equations, figures, tables, and references. It should move quickly from problem to method to result.

```
\documentclass[11pt]{article}
\usepackage{amsmath,amssymb,graphicx,booktabs}
\usepackage[colorlinks=true]{hyperref}
```

Label only equations that are cited later. A label is a promise that the text will return to that item.

```
\begin{equation}\label{eq:energy}
  E_n = \frac{n^2h^2}{8mL^2}.
\end{equation}
Equation~\eqref{eq:energy} shows the quadratic dependence on \(n\).
```

Use BibTeX for simple articles and `biblatex` when citation rules are stricter. In both cases, keep reference data in a `.bib` file.

Article Structure

Most scientific articles can be planned with a simple movement:

Section	Job
Abstract	State the problem, method, main result, and significance in compressed form.
Introduction	Explain the gap, context, and exact question the paper answers.
Method or Theory	Define assumptions, equations, apparatus, data, and procedure.
Results	Present figures, tables, estimates, and comparisons.
Discussion	Interpret the results and name limitations.
Conclusion	State what has been established and what remains open.

The abstract should be written last, even though it appears first. It is a map of the finished argument, not a promise made before the work is clear.

Bibliography Basics

Keep bibliography entries in a separate `.bib` file. The source remains readable, and the same references can be reused in articles, theses, and books.

```
% references.bib
@book{knuth1984texbook,
  author    = {Donald E. Knuth},
  title     = {The TeXbook},
  year      = {1984},
  publisher = {Addison-Wesley}
}

% main.tex
\bibliographystyle{plain}
\bibliography{references}
```

Use `biblatex` when you need stronger control over citation style, sorting, author-year rules, or multilingual bibliographies.

CHAPTER 4

Tables, Figures, and Labels

Tables and figures should explain the document. Labels and references keep those explanations stable when the document changes.

Tables

`tabularx` is useful when one column contains text and must stretch to the line width. `booktabs` gives clean horizontal rules.

```
\begin{tabularx}{\linewidth}{@{}lX@{}}
\toprule
Command & Use \\
\midrule
\input & Insert another source file. \\
\includegraphics & Place an image. \\
\bottomrule
\end{tabularx}
```

Command	Use
`\input`	Insert another source file at the current location.
`\includegraphics`	Place an image, usually inside a figure with a caption and label.
`\label`	Create a reference target.
`\ref`	Print the number of a labelled target.

Equations for Science

Physicists and mathematicians need equations that can be read, referenced, and checked line by line. Use `align` for derivations and related laws; use `cases` for definitions that change by condition.

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0}, \qquad \nabla \cdot \mathbf{B} = 0, \tag{4.1}$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}, \qquad \nabla \times \mathbf{B} = \mu_0 \mathbf{J} + \mu_0 \varepsilon_0 \frac{\partial \mathbf{E}}{\partial t}. \tag{4.2}$$

Equation (4.1) shows the usual pattern: label only the equations the text actually discusses.

```
\usepackage{amsthm}
\newtheorem{theorem}{Theorem}[chapter]
\newtheorem{definition}[theorem]{Definition}

\usepackage{siunitx}
\sisetup{per-mode=symbol}
The acceleration is \qty{9.81}{\metre\per\second\squared}.
```

For mathematics-heavy writing, add theorem environments with `amsthm`. For laboratory, physics, and engineering writing, use `siunitx` so numbers and units stay consistent.

Figures

A figure should have a purpose, a caption, and a label. Put the label after the caption so LaTeX attaches the correct number.

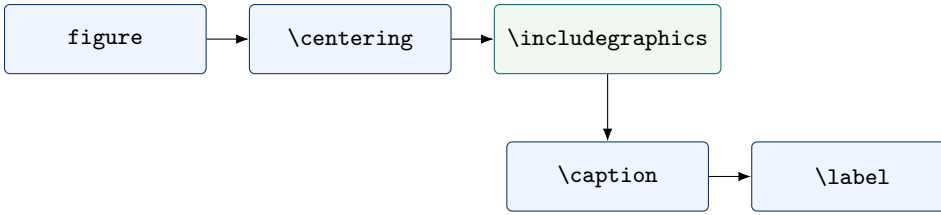

Figure 4.1: A figure float combines image content with a caption and a label.

```
\begin{figure}[htbp]
  \centering
  \includegraphics[width=0.75\linewidth]{experiment}
  \caption{Experimental arrangement.}
  \label{fig:experiment}
\end{figure}
```

Two or Three Images in One Column

Stack related images when the reader should compare stages, views, or panels from top to bottom. Keep one caption for the whole group unless each panel needs a separate explanation.

```
\begin{figure}[htbp]
  \centering
  \begin{minipage}{0.72\linewidth}
    \centering
    \includegraphics[width=\linewidth]{setup-front}

    \vspace{0.5em}
    \includegraphics[width=\linewidth]{setup-side}

    \vspace{0.5em}
    \includegraphics[width=\linewidth]{data-plot}
  \end{minipage}
  \caption{Experimental setup shown from two views with the measured result below.}
  \label{fig:three-image-column}
\end{figure}
```

Use `subcaption` only when the panels need their own labels such as (a), (b), and (c).

Reference Discipline

Labels are part of the document's structure. A label should be unique, meaningful, and placed after the numbered thing it names. Put figure labels after captions, table labels after captions, and equation labels inside the equation environment.

Prefix	Example use
`ch:`	Chapters, such as `\label {ch:methods}`.
`sec:`	Sections, such as `\label {sec:experiment}`.
`eq:`	Equations, such as `\label {eq:schrodinger}`.
`fig:`	Figures, such as `\label {fig:apparatus}`.
`tab:`	Tables, such as `\label {tab:measurements}`.

Use `\eqref` for equations because it prints parentheses in the usual mathematical style. Use `\ref` for chapters, sections, figures, and tables. After adding or moving labels, compile more than once so numbers, contents, and links settle.

Tables for Data

Tables should carry exact values. If the reader needs a trend, use a figure. If the reader needs a number, use a table.

Trial	Voltage	Current	Note
1	1.0	0.12	Stable reading after warm-up.
2	2.0	0.24	Linear response continues.
3	3.0	0.37	Slight deviation at higher setting.

Align numerical columns to the right. Keep long comments in a text column. Do not use vertical rules unless the table is genuinely hard to read without them.

CHAPTER 5

Build, Reuse, and Repair

Compilation is part of writing. The build should be repeatable, generated files should stay separate, and errors should be fixed from the first real cause.

Build Pipeline

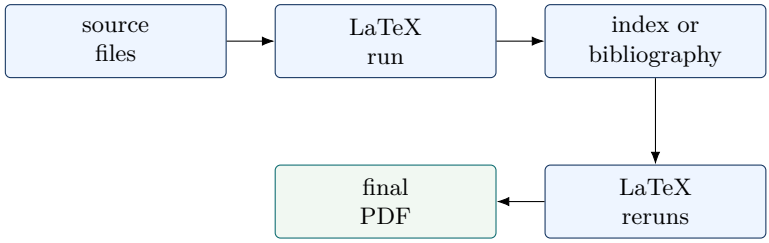

Figure 5.1: Multiple LaTeX runs are normal because references, contents, and indexes settle in stages.

Use the startup commands or `run.sh` to keep auxiliary files in `gar/`. This project's script also copies `gar/main.pdf` to `main.pdf`, so the final PDF appears in the project directory while build files stay separate.

What the Generated Files Mean

Generated files are not junk. They are LaTeX's memory between runs. Keeping them in `gar/` makes the project clean while preserving the information LaTeX needs.

File	Meaning
`.aux`	Cross-reference data, citation requests, labels, and write-back information.
`.toc`	Table-of-contents entries collected from chapters and sections.
`.idx`	Raw index entries written by `\index` commands.
`.ind`	Formatted index created by `makeindex`.
`.log`	The most important diagnostic file; read it when a build fails.
`.out`	Bookmark and outline information for linked PDFs.

When references look wrong, rebuild. When the same error repeats, read the log from the first error line, not from the last warning.

Reuse Without Repeating Mistakes

Reuse stable decisions: page geometry, package loading, heading style, table style, build script, and folder structure. Do not reuse document-specific prose, unfinished formal pages, or old workaround code whose purpose is unclear.

If a design grows, move reusable definitions into a style file such as `bookstyle.sty`. Keep chapter text out of style files.

A Small Style File

When the preamble becomes long, move stable design decisions into a style file. This keeps `main.tex` readable while still making the design reusable.

```
% bookstyle.sty
\ProvidesPackage{bookstyle}
\RequirePackage{xcolor}
\RequirePackage{booktabs}
\RequirePackage{tabularx}

\definecolor{chapterblue}{HTML}{12335F}
\newcommand{\sourcefile}[1]{\texttt{\detokenize{#1}}}
```

Then load it from the main file:

```
\usepackage{bookstyle}
```

Only stable formatting belongs in a style file. Do not hide chapter prose, theorem statements, or project-specific content there.

Repair Table

Message	First place to look
`Undefined control sequence`	Misspelled command or missing package.
`File not found`	Wrong path, wrong extension, or compiling from the wrong folder.
`Missing \$ inserted`	Math syntax used in text mode or an unmatched dollar sign.
`Label(s) may have changed`	Recompile so references and page numbers can settle.
`Overfull hbox`	Long URLs, code, tables, or unbreakable words.

Read the log from the first error. Later errors may only be consequences.

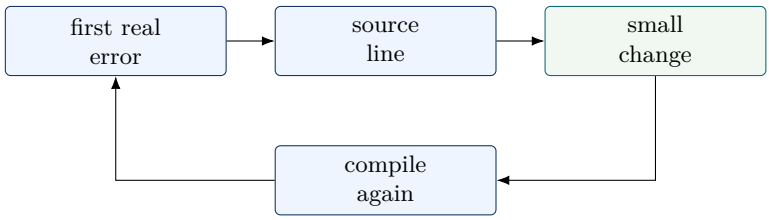

Figure 5.2: Fix one cause at a time; repeated rebuilding is part of the editing loop.

APPENDIX A

Package Reference

Package	What it is normally used for
`geometry`	Page size, margins, header space, footer space, and printable text area.
`fontenc`	Font encoding for pdfLaTeX output.
`inputenc`	Input character handling for pdfLaTeX source files.
`lmodern`	Latin Modern fonts for clean pdfLaTeX documents.
`fontspec`	System-font selection with XeLaTeX or LuaLaTeX.
`amsmath`	Alignments, equation environments, and mathematical structure.
`amssymb`	Additional mathematical symbols.
`mathtools`	Improvements and extensions to `amsmath`.
`amsthm`	Theorem, lemma, definition, and proof environments.
`siunitx`	Consistent scientific units, numbers, and measured quantities.
`bm`	Bold mathematical symbols such as vectors and tensors.
`graphicx`	Image insertion, scaling, rotation, and clipping.
`subcaption`	Multiple related images with panel captions.
`xcolor`	Named colors, custom colors, and optional color support inside tables.
`tikz`	Diagrams drawn directly in LaTeX.
`booktabs`	Professional table rules.
`tabularx`	Tables with automatically stretching text columns.
`array`	Better column definitions in tables.
`enumitem`	Control of list spacing and labels.
`listings`	Source-code display with line breaking and syntax highlighting.
`tcolorbox`	Framed teaching boxes, callouts, examples, and designed text panels.
`fancyhdr`	Running headers and footers.
`titlesec`	Chapter and section heading design.
`titletoc`	Table-of-contents design.
`caption`	Caption font, label, and separator control.
`microtype`	Subtle spacing improvements for more polished paragraphs.
`hyperref`	Clickable links, PDF metadata, bookmarks, and reference targets.
`makeidx`	Index creation.
`biblatex`	Modern bibliography management, usually with `biber`.
`pdfpages`	Inclusion of external PDF pages.

Further Reading

Use these references when the short explanations in this book are not enough.
***The LaTeX Companion*, Frank Mittelbach and Michel Goossens.** Best for broad LaTeX reference, package choices, and practical explanations.

***The TeXbook*, Donald E. Knuth.** Best for understanding paragraphs, boxes, glue, and page building.

AMS *User's Guide for the amsmath Package*. Best for professional equation layouts and mathematical display structures.

Thomas F. Sturm, *The tcolorbox Package Manual*. Best for designed boxes, breakable examples, and framed teaching material.

Index