Your Institution

My Awesome Research Notebook

A Subtitle for My Brilliant Work

Your Name

your.email@example.com

18th July 2025

Contents

Chapter 1

Welcome to Your Elegant Notebook

Welcome to the Elegant Notebook template! We have designed this template to help you create stunning and professional documents with ease. Whether you are taking notes, writing a research paper, or compiling a report, our goal is to make your work shine.

This template is built on a foundation of consistent styling, thoughtful typography, and a powerful theming system.

> **Key Features at a Glance**
>
> Here is a glimpse of what you can do:
>
> - Craft a unique look with our one-step **font and colour theme switcher**.
>
> - Organise your thoughts with beautifully designed **custom boxes** for notes and examples.
>
> - Showcase your work with elegant support for **mathematics**, **code listings**, and **bibliographies**.
>
> - Focus on your content, knowing the design is polished and professional.

Let's get started and bring your ideas to life!

Chapter 2

Getting Started in Three Simple Steps

Getting started is straightforward. Just follow these three steps:

1. **Personalise Your Document**: Open `config/metadata.tex` and add your title, author, and other details.

2. **Choose Your Style**: In this file (`elegant-notebook.tex`), find the `usetheme` command. Pick your favourite font and colour theme to match your taste. See Chapter 4 for a full list of themes!

3. **Compile Your Notebook**: Run one of the provided scripts to compile your document.

2.1 Compiling Your Document

For a quick preview without bibliography processing, use:

```
1  ./simple-compile.sh
```

For a full compilation with bibliography support (this requires the `biber` engine), run:

```
1  ./compile.sh
```

And that's it! You are ready to start writing.

Chapter 3

Showcase of Core Features

This template includes a rich set of features to make your content look its best. Let's explore some of them.

3.1 Effortless Typography and Text

The template provides clean, readable typography right out of the box. You can easily format your text:

- Normal text is optimised for readability.

- **Bold text** for giving emphasis.

- *Italic text* is perfect for definitions or foreign terms.

- `Monospace text` for technical terms.

- Sans-serif text for headings and UI elements.

- Of course, you can use coloured text and highlighted text, too!

    Hyperlinks, like this one to the LaTeX Project, are styled to fit seamlessly.

3.2 Engaging Custom Boxes

Highlight key information and provide clear examples with our custom box environments.

3.2.1 Note Boxes

Use a 'notetbox' to draw attention to important points, warnings, or helpful tips.

> **Note**
>
> This is a note box. It is perfect for highlighting information that your reader should not miss. Note boxes support multiple paragraphs and are styled using your theme's main colour.

3.2.2 Example Boxes

The 'exampletbox' is designed to showcase examples, from mathematical proofs to code snippets.

> **Example**
>
> This is an example box, styled with your theme's accent colour.
> You can include anything you like, such as a famous equation: $E = mc^2$.

3.3 Beautiful Code Listings

Whether you are writing a tutorial or documenting your code, our syntax highlighting provides excellent support. You can use `inline code` for small snippets. For larger blocks:

```python
1  def factorial(n):
2      """Calculate the factorial of a non-negative integer."""
3      if n < 0:
4          raise ValueError("Factorial is not defined for negative numbers")
5      elif n == 0 or n == 1:
6          return 1
7      else:
8          return n * factorial(n - 1)
9
10 # Example usage:
11 result = factorial(5)
12 print(f"The factorial of 5 is {result}")
```

Listing 3.1: A simple Python function

3.4 Seamless Mathematics

The template provides excellent support for mathematical notation, both inline and in display environments.

Inline equations like $\sum_{i=1}^{n} x_i^2 = x_1^2 + \cdots + x_n^2$ fit perfectly within your text. For more complex expressions, use the 'align' environment:

$$f(x) = (x - 1)^2 \tag{3.1}$$
$$f'(x) = 2(x - 1) \tag{3.2}$$

> **Note**
>
> Notice how mathematical expressions are rendered clearly in both inline and display forms. The consistent styling ensures your document remains professional and easy to read.

3.5 Professional Tables and Figures

Present your data and diagrams with elegance.

3.5.1 Tables

Tables are styled using the `booktabs` package for a clean, professional look without vertical lines.

Table 3.1: Sample Experimental Results

Metric	Value	Uncertainty
Mean	0.12	±0.03
Std. Dev.	1.05	±0.02
Skewness	0.05	±0.01

3.5.2 Figures and Diagrams

The template includes full support for TikZ, allowing you to create stunning, customised diagrams.
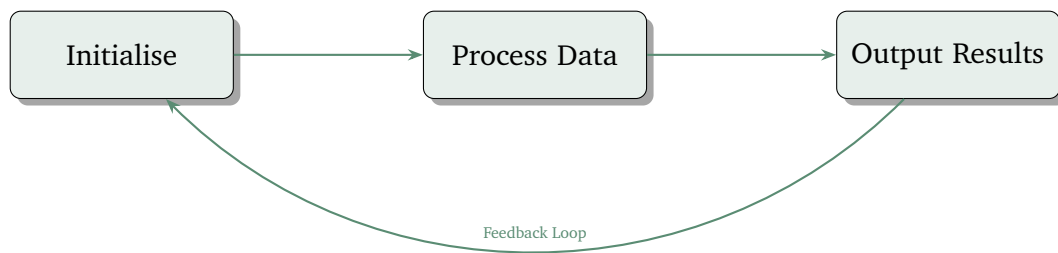
Figure 3.1: A styled TikZ diagram showing a simple workflow with a feedback loop.

3.6 Citations and Bibliography

The template uses BibLaTeX for easy and powerful reference management. You can cite sources from your bibliography, such as [1] or [2].

> **Note**
>
> To use citations, ensure you have the `biber` bibliography processor installed and run the full compilation with `./compile.sh`.

Simply add your references to the `references.bib` file. Here is an example entry:

```
1  @article{Smith2020,
2    author  = {Smith, John and Johnson, Sarah},
3    title   = {Recent Advances in Document Preparation Systems},
4    journal = {Journal of Documentation},
5    year    = {2020},
6    volume  = {76},
7    number  = {3},
8    pages   = {710--725},
9    doi     = {10.1000/example.doi}
10 }
```

Listing 3.2: Example reference entry in references.bib

Chapter 4

Theming: Your Document, Your Style

The powerful theming system is one of the standout features of this template. You can effortlessly change the entire look and feel of your document with a single command.

4.1 One-Step Theme Switching

To change your theme, simply edit the `usetheme` command at the top of this file. It takes two arguments: the font theme and the colour theme.

```
1  % The first parameter is the font theme, the second is the colour theme.
2  \usetheme{font-theme}{colour-theme}
3
4  % Example: Modern fonts with the Oxford Blue theme
5  \usetheme{modern}{oxford-blue}
6
7  % Example: Elegant fonts with the Vintage Rose theme
8  \usetheme{elegant}{vintage-rose}
```

Listing 4.1: Using the theme switcher command

4.2 Available Font Themes

We have curated seven professional font pairings to suit any document. Your current theme is **modern**.

- **modern**: A contemporary look with Charter and Bera Sans (the default).

- **classic**: A traditional academic style with Palatino and TeX Gyre Heros.

- **elegant**: A refined style with EB Garamond and Source Sans Pro.

- academic: The cohesive Libertinus family, perfect for scholarly papers.

- scientific: STIX Two and Cabin, optimised for scientific and mathematical content.

- **professional**: An enhanced Palatino clone with Source Sans Pro for a smart, business look.

- **contemporary**: Modern Fira Sans with matching mathematics support for a clean, unified feel.

4.3 Available Colour Themes

Choose from over 17 carefully designed colour themes.

- **Cambridge Family**: **cambridge-blue** (default), **cambridge-classic**, **cambridge-forest**, **cambridge-sunset**, **cambridge-river**

- **Oxford Family**: **oxford-blue**, **oxford-mist**, **oxford-monochrome**

- **Sky Family**: summer-skies, aurora-skies, twilight-skies, midnight-skies

- **Other Themes**: vintage-rose, emerald, royal-purple, arctic, autumn

4.4 Creating Your Own Colour Theme

Feeling creative? You can easily define your own colour theme in `config/colours.tex`.

> **Note**
>
> To add your own theme, find the "CUSTOM THEME" section in `config/colours.tex` and follow the template pattern provided.
>
> - Choose a unique theme name (e.g., `my-awesome-theme`).
>
> - Define all the required colours.
>
> - Activate it with `usetheme{font-theme}{my-awesome-theme}`.

Here is an example of what a custom theme definition looks like:

```
1  % Add this block to config/colours.tex
2  \ifdefstring{\activetheme}{my-custom-theme}{%
3      \definecolor{maincolor}{RGB}{76,40,130}       % Royal Purple
4      \definecolor{accentcolor}{RGB}{212,175,55}    % Gold
5      \definecolor{highlightcolor}{RGB}{220,20,60}  % Crimson
6      \definecolor{codebg}{RGB}{245,240,255}        % Light purple background
7      \definecolor{codefg}{RGB}{50,30,80}           % Dark purple foreground
8      \definecolor{softgray}{RGB}{200,190,210}      % Purple-tinted grey
9      \definecolor{titlepagebg}{RGB}{240,235,250}   % Soft purple background
10     \definecolor{inlinecodebg}{RGB}{235,230,250}  % Richer purple for
            inline code
11     \definecolor{inlinecodefg}{RGB}{90,50,150}    % Deeper royal purple for
            inline code
12 }{}
13
14 % Then, in elegant-notebook.tex:
15 \usetheme{modern}{my-custom-theme}
```

Listing 4.2: Custom theme definition in config/colours.tex

Chapter 5

Advanced Customisation

For those who want to dig deeper, the template offers even more ways to tailor your document.

5.1 Creating Custom Box Types

You can design your own box styles in `config/settings.tex`. For example, you could create a 'warning-box' for critical alerts.

```
1  \newtcolorbox{warningbox}{
2      enhanced,
3      colback=red!10,
4      colframe=red!80!black,
5      fonttitle=\sffamily\bfseries,
6      coltitle=white,
7      title=Warning,
8      left=2mm, right=2mm, top=2mm, bottom=2mm,
9      arc=2mm,
10     drop shadow,
11     breakable
12 }
```

Listing 5.1: Creating a custom box type in config/settings.tex

5.2 Understanding the Project Structure

The template is organised into a clear and logical file structure, making it easy to find what you need.

```
1  elegant-notebook.tex    # Main document (this file)
2  references.bib          # Bibliography file
3  README.md               # Project documentation
4  compile.sh              # Full compilation script
5  simple-compile.sh       # Basic compilation script
6  test-themes.tex         # Theme preview tool
7  config/                 # Configuration files
8      colours.tex         # Colour theme definitions
9      metadata.tex        # Document metadata (titles, author, etc.)
10     settings.tex        # Core package imports and style settings
11     theme.tex           # The theme switching system
12     title-page.tex      # The title page layout
```

Listing 5.2: Project directory structure

Chapter 6

Conclusion and Best Practices

We hope you enjoy using the Elegant Notebook template. It is designed to be a powerful yet flexible tool that gets out of your way, so you can focus on what matters most: your content.

> **Best Practices for Best Results**
>
> Here are a few final tips for getting the best results:
>
> - Keep your document organised with a clear hierarchy of chapters and sections.
>
> - Use the provided environments ('notetbox', 'exampletbox', etc.) consistently for visual coherence.
>
> - Choose a font and colour theme that matches the tone and subject matter of your work.
>
> - Always use the `./compile.sh` script for the final version to ensure all references and links are correct.
>
> - For collaborative projects, we highly recommend using a version control system like Git.

Happy writing!

References

[1]   John Smith and Sarah Johnson. 'Recent Advances in Document Preparation Systems'. In: *Journal of Documentation* 76.3 (2020), pp. 710–725. DOI: `10.1000/example.doi`.

[2]   Leslie Lamport. *LaTeX: A Document Preparation System*. 2nd. Reading, Massachusetts: Addison-Wesley, 1994.