

## 3 Introduction to L<sup>A</sup>T<sub>E</sub>X

---

L<sup>A</sup>T<sub>E</sub>X is a macro package which enables authors to typeset and print their work with the highest typographical quality, using a predefined, professional layout. Since its introduction, it has been periodically updated and revised, like all software products. For many years now the version number has been fixed at 2 $\epsilon$ . In an effort to re-establish a genuine, improved standard, the L<sup>A</sup>T<sub>E</sub>X3 Project was set up in 1989. And the beta version of L<sup>A</sup>T<sub>E</sub>X3 has just been released. Throughout this tutorial by L<sup>A</sup>T<sub>E</sub>X, we mean L<sup>A</sup>T<sub>E</sub>X 2 $\epsilon$ .

### 3.1 Basics

As explained in the previous chapter, L<sup>A</sup>T<sub>E</sub>X is quite different from WYSIWYG (what you see is what you get) approach which most modern word processors such as MS Word or Corel WordPerfect follow. With these applications, authors specify the document layout interactively while typing text into the computer. Along the way they can see on the screen how the final work will look like when it is printed.

When using L<sup>A</sup>T<sub>E</sub>X it is normally not possible to see the final output while typing the text. But the final output can be previewed on the screen after processing the file with L<sup>A</sup>T<sub>E</sub>X.

Following is the method to create a L<sup>A</sup>T<sub>E</sub>X document.

- (1) Type in the text with necessary commands.
- (2) Compile the text with L<sup>A</sup>T<sub>E</sub>X engine.
- (3) After successful compilation of the document output can be previewed on the screen.

### 3.2 L<sup>A</sup>T<sub>E</sub>X Input Files

The input for L<sup>A</sup>T<sub>E</sub>X is a plain ASCII text file. You can create it with any text editor. It contains the text of the document as well as the commands which tells L<sup>A</sup>T<sub>E</sub>X how to typeset the text. The commands start with a ‘\’ (backslash character).

Eg: \bf, \it, etc.

#### 3.2.1 L<sup>A</sup>T<sub>E</sub>X input file structure

When L<sup>A</sup>T<sub>E</sub>X 2 $\epsilon$  process an input file, it requires us to follow a certain structure. Thus every input file must start with the command

```
\documentclass{class}
```

When all the set up work is done, you start the body of the text with the command

```
\begin{document}
```

Now you enter the text mixed with some useful L<sup>A</sup>T<sub>E</sub>X commands. At the end of the document you add the following command

```
\end{document}
```

which tells L<sup>A</sup>T<sub>E</sub>X the end of the file. Any thing which follows this command will be ignored by L<sup>A</sup>T<sub>E</sub>X.

### 3.2.2 Preamble

The first command in any  $\LaTeX$  file normally determines the global processing format for the entire document. The syntax for this command:

```
\documentclass[options]{class}
```

The possible values of class, of which one and only one may be given, are: article, book, report or letter.

The options available allow various modifications to be made to the formatting, like selecting font size — 10pt, 11pt, 12pt, specifying paper size — letterpaper, legalpaper, executivepaper, page formats — onecolumn, twocolumn etc.

The standard  $\LaTeX$  class used for ordinary documents is the report class. The article class is generally used for shorter documents than the report class. book class is for real books and letter class for formatting letters.

Preamble is the portion between `\documentclass` and `\begin{document}`. This can contain package loading command like `\usepackage{packagename}`. Any number of `\usepackage` command can be issued or alternatively you can give the packagename as a comma separated list in a single `\usepackage` command.

Preamble can also contain the header/footer style chosen, the command for which take the following form:

```
\pagestyle{style option}
```

The style options available are empty (header and footer empty), plain (page number in the footer alone, no header), headings (chapter heading in odd header and section heading in even header, no footers), myheadings (user defined text in odd and even headers, no footers). You can also define your own custom headers and footers with fancied text, boxes, graphic elements, etc.

A typical preamble of a  $\LaTeX$  document will look like:

```
\documentclass[a4paper,11pt,twocolumn]{article}
\usepackage{amsmath,times}
\pagestyle{headings}
\begin{document}
```

## 3.3 The Document

A  $\LaTeX$  document has broadly three parts *viz.*, frontmatter, mainmatter and backmatter.

### 3.3.1 Frontmatter

As the name specifies the frontmatter of an article has the title of the article, its authors, affiliations and an optional date which can be produced with the following commands.

```
\title{Title text}
\author{Author names and addresses}
\date{Date text}
\maketitle
```

The `\maketitle` command will trigger the typesetting of the frontmatter part.

#### 3.3.1.1 Abstract

Abstract is produced with the command:

```

\begin{abstract}
Text for the abstract
\end{abstract}

```

In document class `report`, the abstract appears on a separate page without a page number; in `article`, it comes after the title heading on the first page. An abstract is not possible in document class `book`.

### 3.3.2 Mainmatter

This portion is the body of the document. In the case of `book` class this contains `\chapter`, `\section`, `\subsection` and so on. In the case of `article` class the rest except the `\chapter` command will appear.

### 3.3.3 Backmatter

Backmatter is the portion where the References, or bibliography, containing the names of other works that are referred to within the text appear.

## 3.4 Sectioning commands and its logical relations

The following commands are available for producing automatic, sequential sectioning:

```

\part      \chapter   \subsection   \paragraph
           \section   \subsubsection \subparagraph

```

With the exception of `\part`, these commands form a sectioning hierarchy. In document classes `book` and `report`, the highest sectioning level is `\chapter`. The chapters are divided into sections using the `\section` command, which are further subdivided by means of `\subsection`, and so on. In document class `article`, the hierarchy begins with `\section` since `\chapter` is not available.

### 3.4.1 Sample body of article

```

\documentclass{article}
\usepackage{amsmath}

\begin{document}

\title[Short title]{This is the title}
\author{Author, Affiliation}
\date{}

\maketitle

\begin{abstract}
This is sample abstract.
\end{abstract}

\section{Introduction}
This is sample section.

\subsection{Subsection}
This is sample subsection.

\subsubsection{Subsubsection}

```

This is sample subsection.

```
\paragraph{Paragraph}
This is sample paragraph.
```

```
\begin{thebibliography}{00}
```

```
\bibitem{1} This is sample bibitem one.
\bibitem{2} This is sample bibitem two.
\bibitem{3} This is sample bibitem three.
```

```
\end{thebibliography}
```

```
\end{document}
```

### 3.4.2 Numbering of heading levels

In  $\LaTeX$  the numbering of heading levels are automatically taken care of by  $\LaTeX$ . The default numbering is Arabic. In article class, the top level sectioning unit is `\section` which will start with Arabic '1' and subsequent sections will increment by one. `\subsection` will start with Arabic number '1.1' (*ie.*, section-no.subsection-no) and next subsection will be 1.2 but the subsection counter will reset automatically when next `\section` command is encountered. Similarly, all the sectioning units behave within its heirarchical order.

If you want to avoid the number of a particular heading level, you might add \* character to the command, which is called a *starred version* of that command. The usual convention in  $\LaTeX$  is that all the starred versions do not have counter numbers. An example of a `\section` heading without number will be:

```
\section*{<some heading>}
```

## 3.5 Notes

The above discussion is not exhaustive. Readers might be left with many doubts in formatting the various heading levels, different types of numbers, formatting numbers, alignment, custom page-styles, etc. Queries concerning the above are invited from subscribers which will be answered in detail and will be added to this chapter as FAQ.

Even otherwise, we will revisit the documentclass issues little later, when we shall discuss the advanced features that we consciously missed this time.