Hour 1

Introduction

1.1 What Is LaTeX?

Donald E. Knuth developed TeX in the year 1977 as a typesetting system for preparing books, especially those containing a lot of mathematical expressions. Based on it, Leslie Lamport developed LaTeX (named as LaTeX 2.09) in 1985 for preparing documents by concentrating on the structure of a document rather than on its formatting details. LaTeX 2.09 was enhanced in 1994 as LaTeX 2ε by a group of developers led by Frank Mittelbach.

LaTeX is a macro-package used as a language-based approach for typesetting documents. Various LaTeX instructions are interspersed with the input file of a document, say `myfile.tex`, for obtaining the desired output as `myfile.dvi` or directly as `myfile.pdf`. The `myfile.dvi` file can be used to generate `myfile.ps` or even `myfile.pdf` file. However, unlike programming languages for computational works, such as `C` or `C++`, LaTeX is very simple and easy to work with. One can become expert in LaTeX through a little practice. LaTeX can be used for preparing letters, applications, articles, reports, publications, theses, books, or anything of that kind.

1.2 Why LaTeX Over Other Word Processors?

The use of common word processors, in which the effect becomes directly visible, may be easier in preparing simple and small documents. But, as shown roughly in Fig. 1.1, the effort and time required in LaTeX for preparing complicated and big-size documents are quite less than those required in other word processors[1]. LaTeX is especially well suited

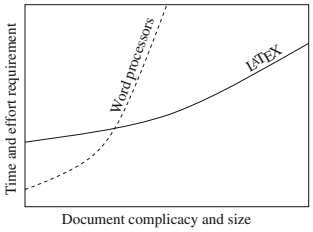

Fig. 1.1 LaTeX and other word processors

[1] Effort and time required in LaTeX for preparing complicated and big-size documents are quite less than those required in other word processors.

D. Datta, *LaTeX in 24 Hours: A Practical Guide for Scientific Writing*,
DOI 10.1007/978-3-319-47831-9_1

for scientific writing, like technical reports, articles, academic dissertations, books, etc. Although learning LaTeX is time and effort taking, it can be realized that the preparation of only one academic dissertation would pay off all additional efforts required in learning LaTeX.

One of the major advantages of using LaTeX is that manual formatting of a document, as usually required in many word processors, can be automated in LaTeX. Therefore, the possibility of doing any mistake in numbering and referring items (sections, tables, figures or equations), in choosing size and type of fonts for different sections and subsections, or in preparing bibliographic references, can be avoided. Further, LaTeX has the provision for automatically generating various lists of contents, index, and glossary.

1.3 How to Prepare a LaTeX Input File?

As shown in Fig. 1.2, the main structure of a LaTeX input file can be divided into two parts – *preamble* and *body*.

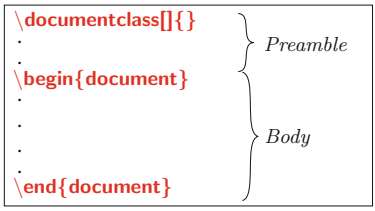

Fig. 1.2 Structure of a LaTeX input file

The preamble is the first part of an input file that contains the global processing parameters for the entire document to be produced, such as the type of the document, page formatting, header and footer setting, inclusion of LaTeX packages for supporting additional instructions, and definitions of new instructions. The simplest preamble is **\documentclass{**`dtype`**}**, where `dtype` in **{}** is a mandatory argument as the class (or type) of the document, such as **letter**, **article**, **report**, or **book**[2]. In the default setting, **\documentclass{}** prints a document on letter-size paper in 10 point fonts (1 point $\approx$ 0.0138 inch $\approx$ 0.3515 mm). Different user-defined formats for a document can be obtained through various options to **\documentclass{}**[3], in which case it takes the form of **\documentclass[**`fo1,fo2,...`**]{**`dtype`**}** with `fo1`, `fo2`, etc., in **[]** as the options (multiple options can be inserted in any order separating two options by a comma), e.g., **\documentclass[a4paper,11pt]{article}** for printing an article on A4 paper in 11 point fonts. Some standard options to **\documentclass[]{}** are listed in Table 1.1 on the next page.

As shown in Fig. 1.2, the main body of a LaTeX input file starts with **\begin{document}** and ends with **\end{document}**. The entire contents to be printed in the output are inserted within the body, mixed with various LaTeX instructions (see §1.5 on page 5 for details). Any text entered after **\end{document}** is simply skipped by a LaTeX compiler.

---

[2] The standard document classes are **letter**, **article**, **report** and **book**. Besides these four standard classes, some other classes are also available, such as **amsart**, **thesis**, **slide**, **slides** and **seminar**.

[3] Different user-defined formats for a document can be obtained through various options to **\documentclass[]{}**.

Table 1.1 Standard options to the **\documentclass[]{}** command

Format	Options	Function
Font size	**10pt** (default), **11pt** and **12pt**	Self-explanatory
paper size	**letterpaper** (default), **a4paper**, **a5paper**, **b5paper**, **legalpaper** and **executivepaper**	Self-explanatory
Page orientation	**portrait** (default) and **landscape**	Self-explanatory
Columns of texts	**onecolumn** (default) and **twocolumn**	Self-explanatory
Type of printing	**oneside** (default for **article** and **report**) and **twoside** (default for **book**)	Self-explanatory
New chapter	**openright** (default for **book**) **openany**	Starts on the next odd page Starts on the next page
Title printing	**titlepage** and **notitlepage**	Self-explanatory
Equation	**leqno** **fleqn**	Number on left hand side Equations are left aligned
Drafting	**final** (default) and **draft**	Self-explanatory
Bibliography	**openbib**	A part on a separate line

A LaTeX input file is named with `tex` extension, say `myfile.tex`. It can be prepared in any operating system using any general-purpose text editor that supports `tex` extension, e.g., gedit or Kate in Linux-based systems. There are also many open-source (free access in internet) text editors developed specifically for preparing LaTeX input files, e.g.,

▷ *For both Windows and Linux*:

- BaKoMa TeX (http://bakoma-tex.com)
- Emacs (www.gnu.org/software/emacs/emacs.html)
- jEdit (http://jedit.org)
- Kile (http://kile.sourceforge.net)
- LyX (www.lyx.org)
- Open LaTeX Studio (http://sebbrudzinski.github.io/Open-LaTeX-Studio)
- TeXlipse (http://texlipse.sourceforge.net)
- TeXmacs (www.texmacs.org)
- Texmaker (www.xm1math.net/texmaker)
- TeXpen (https://sourceforge.net/projects/texpen)
- TeXstudio (www.texstudio.org)
- TeXworks (https://github.com/TeXworks/texworks), etc.

▷ *For Linux only*:

- LaTeXila (https://wiki.gnome.org/Apps/LaTeXila)
- Gummy (https://github.com/alexandervdm/gummi), etc.

▷ *For Windows only*:

- Inlage (www.inlage.com)
- LEd (www.latexeditor.org)
- TeXnicCenter (www.texniccenter.org)
- WinShell (www.winshell.de)
- WinEdt (www.winedt.com), etc.

Table 1.2 A simple LaTeX input file and its output

LaTeX input	Output
\documentclass{article} \begin{document} **LaTeX** is a macro package for typesetting documents. It is a language-based approach, where LaTeX *instructions* are interspersed with the text file of a document, say `myfile.tex`, for obtaining the desired output as `myfile.dvi`. The `myfile.dvi` file can then be used to generate `myfile.ps` or `myfile.pdf` file. \end{document}	LaTeX is a macro package for typesetting documents. It is a language-based approach, where LaTeX instructions are interspersed with the text file of a document, say myfile.tex, for obtaining the desired output as myfile.dvi. The myfile.dvi file can then be used to generate myfile.ps or myfile.pdf file. 1

A simple input file, say `myarticle.tex`, prepared under the document-class of **article** is shown in the left column of Table 1.2, along with its output in the right column. The actual contents to be printed in the output file were inserted within **\begin{document}** and **\end{document}**. Surprisingly, the output is not the one as expected. The differences are shown underlined in the output file. This has happened due to the fact that many things in LaTeX can be obtained through some special instructions only as stated in §1.5. Another thing to be noticed is that the output file is assigned a default page number at the bottom-center. However, before going to such issues, the compilation procedure of a LaTeX file is stated first in the following section.

1.4 How to Compile a LaTeX Input File?

A LaTeX input file can be compiled in many LaTeX editors mentioned in § 1.3 on page 2. Besides, operating-system based many open-source LaTeX compilers are also available, e.g., MiKTeX (http://miktex.org) for Windows operating system, or TeXLive (https://www.tug.org/texlive) for both Windows- and Linux-based operating systems. In a GUI-based compiler, like MiKTex or Kile, a LaTeX file can be compiled just by a mouse-click. In other command-line compilers, a LaTeX file is to be compiled through the latex command, followed by the name of the input file with or without its `tex` extension. For example, `myarticle.tex` of Table 1.2 can be compiled as follows:

```
$ latex myarticle
```

This command will produce three files, namely `myarticle.aux`, `myarticle.log`, and `myarticle.dvi` (refer § 20.4.1 on page 199 for detail). Out of these three files, `myarticle.dvi` is the final output which can be viewed in a document viewer, such as *xdvi* or *Evince*. The `myarticle.dvi` file can also be used for producing `myarticle.ps` or `myarticle.pdf` as the final output of `myarticle.tex`. The commands for these are as follows:

```
$ dvips -o myarticle.ps  myarticle.dvi
Or, $ dvipdf myarticle.dvi
```

A `pdf` file can also be generated directly using the pdflatex command, instead of the latex command as stated above. For example, the following command will also generate `myarticle.pdf`:

```
$ pdflatex myarticle.tex
```

The compilation will need some more commands if a document involves bibliography, lists of contents, index, etc. Such commands will be addressed in relevant Hours.

1.5 LaTeX Syntax

LaTeX syntax consists of commands and environments, which are kinds of instructions interspersed with the texts of a document for performing some specific jobs. Such instructions are defined in different packages[4].

1.5.1 Commands

A LaTeX command is an independent instruction used either for producing something new or to change the form of an existing item, e.g., producing the symbol α or printing `italic` as *italic*. Different properties of the LaTeX commands are stated below:

- ▷ A command usually starts with a **** (backslash), followed by one or more characters without any gap in between. For example, **\LaTeX** and **\copyright** for producing LaTeX and © respectively.
- ▷ Non-alphabetic characters normally cannot appear in the name of a LaTeX command. Some LaTeX internal commands may start as **\@**, which (commands) are to be put in the preamble preceded and followed, respectively, by the **\makeatletter** and **\makeatother** commands.
- ▷ Many commands require some mandatory arguments, up to a maximum limit of nine, each in a separate pair of **{}** (preferably without any gap in between), e.g., **\textcolor{blue}{**`this is blue colored`**}** (detail is in §2.4 on page 13) for printing the second argument in blue color.
- ▷ Many commands have the provision for accepting some optional instructions also, written in **[]** separating two options (instructions) by a comma, e.g., **\documentclass[a4paper,11pt,twoside]{article}**. A command with the provision for optional arguments must have at least one mandatory argument.
- ▷ A command ended by an alphabet (i.e., a command not having any argument) ignores trailing blank spaces. Hence, if followed by a word or a number, such a command should be ended by **\␣** (the ␣ symbol is used to indicate a blank

[4] In this book, LaTeX commands, environments and packages, as well as other LaTeX syntax, are printed in **red colored** (for online version) **and boldfaced sans serif fonts** for their clear distinction from other texts of the book.

space obtained by pressing the Spacebar or Tab button of the keyboard). For example, \copyright␣2007 will produce ©2007, while \copyright\␣2007 will produce © 2007. However, if such a command is followed by any punctuation, it needs not to be ended by \␣ as a punctuation is not to be preceded by any blank space.

1.5.2 Environments

A LaTeX environment is a structure composed of two complementary commands, within which some particular job can be performed, e.g., writing an equation or inserting a figure. Different properties of the LaTeX environments are as follows:

- ▷ The pair of complementary commands creating an environment structure are \begin{ename} and \end{ename}, where ename is the name of the environment, e.g., \begin{document} and \end{document} as shown in Fig. 1.2 on page 2 creates the document environment (or the body) in a LaTeX input file.
- ▷ It is possible to use a command inside an environment, or to nest two or more environments, e.g., within the document environment, the \LaTeX command for printing LaTeX or the figure environment for inserting an external figure.
- ▷ Many environments require some mandatory arguments, which are placed after \begin{}, e.g., \begin{spacing}{1.3} for creating a line spacing of 1.3 pt through the spacing environment, or \begin{tabularx}{10cm}{XXX} for creating a table of three equal-width columns over 10 cm length through the tabularx environment.
- ▷ Like a command, many environments also have the provision for accepting some optional instructions written in a pair of [], e.g., \begin{table}[t] preferring through the option t in the table environment to place a table at the top of a page.

1.5.3 Packages

The class (or type) of a document, incorporated through the mandatory argument of the \documentclass{} command, includes some basic features of the document, like page layout and sectioning. Provision is also there to invoke additional commands and environments in a document for adding extra features that are not parts of the standard document class. Such commands and environments are defined in separate files, known as packages.

- ▷ A package is loaded (included) in the preamble, in between the \documentclass{} and \begin{document} commands[5], through the \usepackage{pname} command, where pname is the name of the package, e.g., \usepackage{color} for producing colored texts or \usepackage{amssymb,amsmath} for producing AMS type mathematical symbols and expressions.

[5] If anything (a command or a package) is asked to be put in the preamble of a LaTeX input file, it is to be inserted in between the \documentclass{} and \begin{document} commands.

▷ Like commands and environments, many packages also have the provision for accepting some optional instructions in **[]**, e.g., **\usepackage[tight]{subfigure}** preferring through the option **tight** to reduce extra space between figures.
▷ Unlike an option to **\documentclass[]{}**, which is global to the entire document – including other packages too, an option to **\usepackage[]{}** is local only to the features defined in the package(s) loaded through the **\usepackage[]{}** command.

1.6 Keyboard Characters in LaTeX

Not all, but only those characters of an *English keyboard*[6], shown in Table 1.3, can be

Table 1.3 Keyboard characters that can be produced directly

Type of character	Characters
Uppercase letters	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
Lowercase letters	a b c d e f g h i j k l m n o p q r s t u v w x y z
Digits	0 1 2 3 4 5 6 7 8 9
Parentheses	( )
Brackets	[ ]
Quotations	` ' "
Punctuation	, ; : ! . ?
Math operators	+ - * / =
Other symbols	@

printed directly in a LaTeX document (the left-hand quotation mark (`), shown in Table 1.3, generally appears on the same button with the ~ symbol). All other characters of an English keyboard, which are not included in Table 1.3, need to be produced in a LaTeX document through some commands (most of these characters are reserved in LaTeX for special purposes). Table 1.4 on the following page lists those special characters, commands for producing them in a LaTeX document, and also the purposes, if any, for which these are reserved in LaTeX. For the commands starting and ending with the **$** symbol (i.e., in **$ $**), the **amssymb** package may be required (when used in an equation, as addressed in Hour 11 on page 101, these commands need not to be enclosed in **$ $**).

The special keyboard characters, listed in Table 1.4, can be produced in text-mode using the **\verb" "** or **\verb! !** command also, e.g., **\verb"**$**"** for printing $, or **\verb!**~**!** for printing ~ (the **\verb" "** and **\verb! !** commands are used for printing *as-it-is* what is written within **" "** or **! !** in the input file).

The commands for producing language-specific keyboard characters, as well as many other characters and mathematical symbols, are given in Appendix A on page 247.

[6]Many computers are manufactured for particular countries where a keyboard contains some language-specific characters, in addition to those used in the English language. However, for general purpose, a keyboard containing the characters, used in the English language only, will be discussed in this book.

Table 1.4 Keyboard characters to be produced through commands

Character	Command	Function in LaTeX
$	\$	A pair of $ creates a math-mode* within text-mode.
%	\%	Texts of a line preceded by % are commented.
{ }	\{ \}	Mandatory arguments of a command are written within {}.
_	_	Generates a subscript in math-modes.
^	\^\	Generates a superscript in math-modes.
&	\&	Separates the entries of two columns in a Table.
#	\#	Miscellaneous symbol.
\	$\backslash$	Most of the LaTeX commands start with \.
~	$\sim$	Binds two words to be printed in the same line.
|	$\|$	Generates a vertical (column) line in a Table.
<	$<$	—
>	$>$	—

*Text processing modes are discussed in the first paragraph of Hour 2 on the next page.

1.7 How to Read this Book?

The version LaTeX 2ε of LaTeX is discussed in this book. However, without referring the version, only the general term LaTeX is used throughout the book.

It is not a reference manual, but a practical guide prepared based only on the author's own experiences. The book is intended for beginners, and hence it discusses mainly the basic elements available in the standard LaTeX 2ε distribution. Many relevant advanced topics are also discussed marking them as starred (*), which can be skipped in the initial stage of learning. Many statements are repeated in the book so that a reader is not necessarily required to remember everything. All LaTeX syntax, including commands and environments, are printed in **red colored** (for online version) **and boldfaced sans serif fonts** to make them easily distinguishable from normal texts.

Since the book is meant for beginners, it is suggested to read the entire book for better understanding of LaTeX. However, as mentioned in the Preface that many students and professionals are interested to get their works done in less time and least efforts, one can move directly to Hour 19 on page 181 or Hour 20 on page 191 to start writing a document immediately. Then other Hours can be followed for required additional information, where different topics – writing equations, drawing tables, inserting figures, etc., are discussed in detail. Also, many important points are highlighted in foot notes on various pages. For quickly locating the availability of different topics, one may browse through the `Contents`, `List of Tables` and `List of Figures` at the beginning of the book, or `Index` at the end of the book for searching different terminologies, where attempts are made to include information almost about all the materials that are discussed in this book.