Hour 20

Book and Report

Preparation of a book or a report is similar to that of an article discussed in Hour 19. The only difference is that a book or a report contains a number of chapters, where each chapter is like an independent article. A book is prepared through the document-class **book** and a report through the document-class **report**. The structure of a report is similar to that of a book, expect some differences in the output. Therefore, the report preparation is not discussed here separately. The differences can be observed just by changing the document-class of a document from **book** to **report**, or from **report** to **book**. Note that an academic thesis can be prepared in any of the document-classes of **book** and **report**.

20.1 Template of a Book

The general template of a book is shown in Table 20.1 on the following page, where the book is produced in three parts – **\frontmatter** (front matter), **\mainmatter** (main matter), and **\backmatter** (back matter). The **\frontmatter** command covers the title, preface, contents, etc., with page numbering in Roman numerals and no numbering to the **\chapter{}** command. The main part of the book comes under the **\mainmatter** command with page and chapter numbering in Arabic numerals. Finally, appendix, bibliography, index, etc., are inserted under the **\backmatter** command without chapter numbering.

The default template given in Table 20.1 can be altered by user-defined templates, an example of which is shown in Table 20.2 on page 193 (if compiled, its output would be similar to this book to a large extent). The **twoside** option is used in **\documentclass[ ]{}** for producing the book on both sides of a page (which is the usual requirement in a book). The *cover page* is started with **\thispagestyle{empty}** for not assigning any page number to this page. The **\cleardoublepage** command, used after the *cover page* (also after the *Preface* and the *Table of Contents*), leaves the last even-numbered page blank, if the previous unit ends on an odd-numbered page, and starts the following unit on the next odd-numbered page (**\cleardoublepage** is not required

D. Datta, *LaTeX in 24 Hours: A Practical Guide for Scientific Writing*,
DOI 10.1007/978-3-319-47831-9_20

Table 20.1 General template of a book in the document-class **book**

```
\documentclass[11pt,a4paper]{book}
⋮
\title{\LaTeX\ in 24 Hours\\A Practical Guide for Scientific Writing}
\author{Dilip Datta}
%
\begin{document}
\frontmatter                    % First part of the book
\maketitle
\chapter{Preface}
⋮
\tableofcontents
%
\mainmatter                     % Second part of the book
\chapter{Introduction}
⋮
\chapter{Equation1}
⋮
\backmatter                     % Last part of the book
\appendix
\chapter{Appendix}
⋮
\bibliographystyle{plain}
\bibliography{mybib}
\printindex
\end{document}
```

between two chapters as, by default, a chapter is started on an odd-numbered page). Before starting the *Preface*, **\pagenumbering{roman}** is used for numbering pages now onwards in Roman numerals. The *Preface* is inserted as a chapter with the **\chapter*{}** command (instead of **\chapter{}**) for not assigning a serial number to it. Next, the **\tableofcontents** command is used for automatically generating the contents list of the book. Finally, the actual chapters of the book are started, preceded by **\pagenumbering{arabic}** for numbering pages now onwards in Arabic numerals.

20.2 Book Preparation Using a Root File

Table 20.2 shows a template for preparing a book in a single input file. Since a book usually contains a large number of pages, practically it will be difficult to work with such a single file because of its huge size in accommodating the contents of a book. Hence, as a convenient way, a book is generally prepared in a number of input files of smaller sizes. Usually each part of a book, such as the title, preface, dedication,

Table 20.2 User-defined template of a book in the document-class **book**

```
\documentclass[a4paper,11pt,twoside]{book}
⋮
\begin{document}
% Cover page
\thispagestyle{empty}
\begin{center}
    {\Huge\bf \LaTeX\ in 24 Hours}\\[5mm]
    {\Large\bf A Practical Guide for Scientific Writing}
\end{center}
\cleardoublepage
% Preface
\pagenumbering{roman}
\chapter*{Preface}
The necessity for writing this book was felt long back ...
\cleardoublepage
% Contents
\tableofcontents
\cleardoublepage
% Starting chapters
\pagenumbering{arabic}
\chapter{Introduction}
Donald E. Knuth developed \TeX\ in the year 1977 ...
%
\chapter{Fonts Selection}
There are three modes for processing texts in \LaTeX\ ...
⋮
\appendix
\chapter{Appendix}
⋮
\bibliographystyle{plain}
\bibliography{mybib}
\printindex
\end{document}
```

chapters, appendix, and bibliography, is prepared in a separate input file and then all the individual files are compiled together to produce a single output file as the final book. Moreover, the preamble (i.e., the required packages, user-defined commands and environments, page formatting, or any other global setting) may also be prepared in a separate input file. The input file containing the list of bibliographic references is named with `bib` extension, say `mybib.bib`. All other input files are named with `tex` extension, such as, `preamble.tex`, `title.tex`, `preface.tex`, `chapter1.tex`, `chapter2.tex`, etc. Finally, all the individual input files are linked to a single root file, say `mybook.tex`, compilation of which produces the final book. An input file can be linked to the root file either through the **\input{}** or **\include{}** command (§20.4.2 on page 201 discusses the differences between the commands) with the name of the input file (without the extension) as the argument of the command. If prepared in the form shown in Table 14.2 on page 139, the bibliography file is included in the **thebibliography** environment using the **\input{}** command, otherwise (if prepared in the form shown in Table 15.2 on page 145) it is included through the **\bibliography{}** command following the **\bibliographystyle{}** command as shown in Tables 20.1 and 20.2. A sample preamble file, a few other files, and the root file of a book, similar to those of this book, are shown in Tables 20.3, 20.4, and 20.5.

Table 20.3 shows the preamble file (`preamble.tex`), most of the LaTeX instructions of which are already discussed in previous Hours, while others are self-explanatory. As mentioned above, by default a chapter is started on an odd-numbered page by leaving the previous even-numbered page blank, if required. However, it is

Table 20.3 Preamble of a book

```
% File name: preamble.tex
\documentclass[a4paper,11pt,twoside,openany]{book}
% Basic packages
\usepackage{float}                       % For new floating environments
\usepackage{stmaryrd,amssymb,amsmath}    % For mathematical symbols and equations
\usepackage{array}                       % For arrays of equations
\usepackage{epsfig,graphicx,subfigure}   % For inserting figures
\usepackage{wrapfig}                     % For wraping texts around tables and figures
\usepackage{tabularx}                    % For auto-adjusted column widths in tables
\usepackage{multirow}                    % For merging cells in tables
\usepackage{longtable}                   % For multi-page tables
\usepackage{rotating}                    % For rotating a page (landscape) or inclined texts
\usepackage{caption}                     % For adjusting captions of tables and figures
\usepackage{color}                       % For writing colored texts
\usepackage{setspace}                    % For adjusting line spacing
\usepackage{boxedminipage,fancybox}      % For boxed texts
\usepackage{shadow}                      % For creating shaded box
\usepackage{natbib}                      % For bibliographic references
\usepackage{varioref}                    % For referring through \vref{ } & \vpageref{ }
\usepackage{url}                         % For citing URL
\usepackage{makeidx}                     % For generating index
%
\makeindex                               % Generate index
% Blank space adjustment
\abovecaptionskip                        % Skips extra space above a caption
\belowcaptionskip                        % Skips extra space below a caption
\raggedbottom                            % Top aligning a page leaving space at the bottom
% User-defined new commands
\definecolor{ugray}{gray}{0.25}          % User-defined gray color 'ugray'
\newcommand{\tgray}{\textcolor{ugray}}   % '\tgray{ }' for writing in user-defined 'ugray'
\newcommand{\tred}{\textcolor{red}}      % '\tred{ }' for writing in red
\newcommand{\vctr}[1]%
        {\mbox{\boldmath{$#1$}}}         % Prints x as a vector through \vctr{x}
\newtheorem{thm}{Theorem}                % Environment 'thm' for writing theorems
\newtheorem{dfn}{Definition}             % Environment 'dfn' for writing definitions
\newtheorem{lem}{Lemma}                  % Environment 'lem' for writing lemmas
```

generally not desirable to leave any page blank in a book. The **openany** option to the **\documentclass[]{}** command works for that, which instructs to start a new chapter on the immediate next blank page, no matter whether odd- or even-numbered.

The sample files, containing the title, preface, Chapter 1, and appendix of a book, are shown in short form in Table 20.4. Note that neither the **\documentclass[]{}** command nor the **document** environment is required in these files. The **\documentclass[]{}** command is included in the preamble file, and the **document** environment is created in the root file in which these individual files are to be linked. The title page (`title.tex`) is written in the **titlepage** environment, which produces a page without any page number and heading. The **\vspace*{\fill}** command is inserted at the top and bottom of the file for vertically center aligning its contents. The Preface (`preface.tex`) is prepared as a chapter under the **\chapter*{}** command (**\chapter*{}** used instead of **\chapter{}** for preventing it from assigning any chapter number). The introduction (`chap_intro.tex`) is prepared as a normal chapter under the **\chapter{}** command (other chapters of a book are to be prepared in the same way). Even the appendix (`app_symb.tex`) is also prepared as a normal chapter like `chap_intro.tex`.

Table 20.4 Some sample input files of a book

```
% File name: title.tex
\vspace*{\fill}
\begin{titlepage}
\begin{center}
   {\Huge\bf \LaTeX\ in 24 Hours}\\[5mm]
   {\Large\bf A Practical Guide for Scientific Writing}
\end{center}
⋮
\end{titlepage}
\vspace*{\fill}
```

```
% File name: preface.tex
\chapter*{Preface}
The necessity for writing this book was felt long back, during my Ph.D
work, when I saw students and researchers struggling with \LaTeX\ for
writing their articles and theses ...
```

```
% File name: chap_intro.tex
\chapter{Introduction}\label{chap:intro}
%
\section{What is \LaTeX?}\label{sec:latex}
\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 ...
```

```
% File name: appsymb.tex
\chapter{Symbols and Notations}\label{app:symbol}
%
There are unlimited number of symbols and notations which may be required
to be used in different documents. Moreover, there exist many special
letters used in different languages. All such symbols and letters are to
be produced in a \LaTeX\ file through commands ...
```

Finally the root file (mybook.tex), where all the individual input files are linked, is shown in Table 20.5. Different segments of mybook.tex are as follows:

- ▷ At the very beginning, the preamble file (preamble.tex) is included through the \input{preamble} command, and then all other '.tex' files (without the tex extension) are included in the **document** environment through the \include{} command.
- ▷ The BibTeX formatted bibliography file (mybib.bib) is inserted through the \bibliography{mybib} command, preceded by a bibliography style through the

Table 20.5 Root file linking individual input files of a book

```
% File name: mybook.tex
\input{preamble}
\begin{document}
    \begin{spacing}{1.2}
        % Cover Page, Title, Preface and Dedication
        \thispagestyle{empty}  \include{coverpage}  \cleardoublepage
        \pagenumbering{roman}
            \phantomsection\addcontentsline{toc}{chapter}{Title}
                \thispagestyle{empty}  \include{title}  \cleardoublepage
            \phantomsection\addcontentsline{toc}{chapter}{Dedication}
                \thispagestyle{empty}  \include{dedication}  \cleardoublepage
            \phantomsection\addcontentsline{toc}{chapter}{Preface}
                \thispagestyle{empty}  \include{preface}  \cleardoublepage
            % Contents, List of Tables and List of Figures
            \phantomsection\addcontentsline{toc}{chapter}{Contents}
                \thispagestyle{empty}  \tableofcontents  \cleardoublepage
            \phantomsection\addcontentsline{toc}{chapter}{List of Tables}
                \thispagestyle{empty}  \listoftables  \cleardoublepage
            \phantomsection\addcontentsline{toc}{chapter}{List of Figures}
                \thispagestyle{empty}  \listoffigures  \cleardoublepage
        % Chapters
        \pagenumbering{arabic}
            \include{chap_intro}
            \include{chap_font}
            \include{chap_format}
            \include{chap_table}
            ⋮
    \end{spacing}
    % Appendix, Bibliography and Index
    \begin{spacing}{1.0}
        \begin{appendix}  \include{app_symb}  \end{appendix}

        \phantomsection\addcontentsline{toc}{chapter}{Bibliography}
            \bibliographystyle{plain}  \bibliography{mybib}  \clearpage

        \phantomsection\addcontentsline{toc}{chapter}{Index}
            \printindex  \cleardoublepage
    \end{spacing}
\end{document}
```

\bibliographystyle{} command, say **\bibliographystyle{plain}** for printing the bibliographic references in the **plain** style (refer §15.1 on page 141 for detail).

▷ By default the page number is not printed only on the first page of a chapter. Since the files of the cover page, title, and dedication are not prepared as chapters, the inclusion of each of these input files is preceded by the **\thispagestyle{empty}** command for not assigning any page number to these pages.

▷ The **\cleardoublepage** command is used in many places for starting the following unit on the next odd-numbered page.

▷ Since the title, dedication, and preface are not prepared as numbered units (the preface is prepared under **\chapter*{}** for preventing it from numbering), the inclusions of their input files are preceded by the **\addcontentsline{}{}{}** command for including them in the `Contents` list (refer §16.1.1 on page 153 for the use of **\phantomsection** before **\addcontentsline{}{}{}**). The arguments of **\addcontentsline{}{}{}** are, respectively, the location where the unit is to be entered (**toc** for `Contents` list), the type how the unit is to be treated (**chapter**) and the name of the unit to be printed in the `Contents` list (refer §16.1.1 on page 153 for detail).

▷ The **\tableofcontents**, **\listoftables**, and **\listoffigures** commands are used for automatically generating three lists – `Contents` list of sectional units, `List of Tables`, and `List of Figures`, respectively. Since no serial number is assigned, these lists are also included in the `Contents` list through the **\addcontentsline{}{}{}** command as mentioned above.

▷ An appendix is prepared as a normal chapter. It is identified as an appendix through the **\appendix** command or inserting the appendix input file in the **appendix** environment, which generates the label-word ‘`Appendix`’, followed by the serial number of an appendix in an uppercase alphabet, like `A`, `B`, etc. (if the label-word ‘`Appendix`’ is not generated but ‘`Chapter`’, it can be obtained by inserting the **\renewcommand{\chaptername}{**`Appendix`**}** command before the inclusion of the appendix files). The **\appendix** command and **appendix** environment also cause sectional units of an appendix to be numbered properly, like `A.1`, `A.1.1`, etc. In order to assign the continued serial numbers to different appendices, the **\appendix** command is to be used only once before the inclusion of the first appendix file, or all the appendix files are to be inserted in a single **appendix** environment.

▷ The location for producing the index is specified through the **\printindex** command. The index is produced under the heading ‘`Index`’. Since the index is also not assigned any serial number, it is included in the `Contents` list through **\addcontentsline{}{}{}**.

▷ Two different line spacings are used in `mybook.tex`, through the **spacing** environment defined in the **setspace** package. From the cover page onwards, the line spacing is set at 1.2 up to the last chapter of the book, while the appendix and bibliography are produced in single line spacing.

▷ The `\pagenumbering{roman}` command is used before the inclusion of the title page (`title.tex`) for numbering the following pages in Roman numerals. Then the page numbering from the starting of the first chapter (`chap_intro.tex`) is changed to Arabic numerals through the `\pagenumbering{arabic}` command, which is continued up to the last page of the book.

20.3 Dividing a Book into Parts*

Like in articles where different sections can be divided into a number of parts as discussed in §19.2.6 on page 189, different chapters of a book can also be divided into a number of parts through the `\part{}` command. Each `\part{}` command generates the label-word '`Part`' followed by the serial number of the part in a uppercase Roman numeral, such as `Part I` or `Part II`. Any text in the argument of the `\part{}` command is printed as the heading of the part. Similar to articles, although divided into a number of parts, by default the chapters of the book are assigned continuous serial numbers irrespective of the parts in which they belong. For numbering the chapters part-wise, the following four lines of commands may be inserted in the preamble:

```
\makeatletter
\@addtoreset{chapter}{part}
\makeatother
\renewcommand{\thechapter}{\thepart.\arabic{chapter}}
```

where `\@addtoreset{}{}` command resets the chapters to be numbered part-wise. The `\renewcommand{}{}` command here redefines the numbering of chapters (`\thechapter`) to be started by the serial of the part (`\thepart`) in which a chapter belongs, followed by a period (`.`) and then the serial number of the chapter in an Arabic numeral (`\arabic{}`). Without this `\renewcommand{}{}` command, the chapters under different parts will be numbered in the same way, like `Chapter 1` or `Chapter 2`, which would be confusing to understand if a chapter is referred somewhere.

20.4 Compilation of a Book

Since a book is generally composed of a number of chapters, one or more appendices, bibliography, and index, its compilation would be a combination of those discussed in §15.4 on page 149 and §16.2.3 on page 160. As an example, the command-prompt compilation of the book '`mybook.tex`' of Table 20.5 would involve the following five lines of commands (refer §20.4.1 on the next page for detail):

```
$ latex mybook
$ bibtex mybook
$ makeindex mybook
$ latex mybook
$ latex mybook
```

The command ‘latex mybook’ in the first line compiles mybook.tex as well as all other .tex files included in mybook.tex. It also generates the output in the form of mybook.dvi. However, the bibliography and index remain uncompiled. These are required to be compiled separately, which are done through the commands bibtex and makeindex in the second and third lines, respectively. Then the command of the first line is repeated in the fourth line for linking all the compilation. As the last step, the same command is repeated once again in the fifth line for producing the complete and final output file ‘mybook.dvi’.

*20.4.1 Executable File for Compiling a Book**

It is seen in §1.4 on page 4 that only one command is required to compile a simple document in a command-prompt. However, the number of commands increases with the increasing number of provisions. Section 15.4 on page 149 and §16.2.3 on page 160 show that four lines of commands are required if any of bibliography and index is included in a document. The number of commands is increased to five if both bibliography and index are included. This is shown on this page above at the time of compiling a book. Since a document may need to be compiled a number of times during its preparation, it may become a cumbersome job to type all these commands repeatedly. Therefore, all the commands may be stored in a separate compilation file, which can be run to compile all the commands stored in it. The compilation file, named ‘compile’ without any extension, prepared for compiling the book having the root file named as mybook.tex, is given in Table 20.6. The line numbers are included for the purpose of explanation only. The commands in lines 2–6 are used to compile the book. The command of line 2 compiles mybook.tex as well as all other

Table 20.6 Compilation file for compiling a book in command-prompt

```
1   rm *.aux *.log *.dvi *.blg *.bbl *.idx *.ilg *.ind *.toc *.lot *.lof

2   latex mybook
3   bibtex mybook
4   makeindex mybook
5   latex mybook
6   latex mybook

7   dvipdf mybook.dvi

8   rm *.aux *.log *.dvi *.blg *.bbl *.idx *.ilg *.ind *.toc *.lot *.lof
```

.tex files included in mybook.tex, while the commands of lines 3 and 4 compile the bibliography (if prepared through the BIBTEX program) and index, respectively. The commands of lines 5 and 6 (same with that of line 2) link all the three compilations (latex, bibtex, and makeindex) and produce the final output. Finally, the command of line 7 generates the mybook.pdf file. During the entire compilation process, a number of intermediate files are generated, samples of which are shown in Table 20.7.

Table 20.7 Intermediate files generated during the compilation of a book

Command	Location	Generated files
latex mybook	Line 2 in Table 20.6	mybook.aux, coverpage.aux, title.aux, preface.aux, dedication.aux, chap_intro.aux, chap_font.aux, chap_format.aux, ..., mybook.log, mybook.dvi
bibtex mybook	Line 3 in Table 20.6	mybook.bbl, mybook.blg
makeindex mybook	Line 4 in Table 20.6	mybook.idx, mybook.ilg, mybook.ind
\tableofcontents	mybook.tex	mybook.toc
\listoftables	mybook.tex	mybook.lot
\listoffigures	mybook.tex	mybook.lof

Information about the references required in a .tex file is written in an auxiliary file having the same name but with aux extension. The transcript file with log extension (having the same name with the root file of the book) records all the execution information, such as the names of the files read, the numbers of the pages processed, warning and error messages, and other pertinent data. Most of the information of the .log file is displayed on the screen also. The BIBTEX program generates the .bbl and .blg files, where sorted bibliographic references are stored. The files with idx, ind, and ilg extensions are index-related files, where indexed items and associated information are written. Other LATEX output files are related with the contents list, List of Tables, and List of Figures, having extensions toc, lot, and lof, respectively. The final output is written in a *device independent* file, having dvi extension, with a resolution better than a thousandth of an inch.

Coming back to Table 20.6, mybook.dpf file is produced in line 7 from the final mybook.dvi file. Then all the intermediate files, mentioned in Table 20.7, are removed in line 8 using the linux command ‘rm’. Note that, if the execution of the compile file is terminated in between due to some bug in the LATEX input files, sometimes the compiler may fail to overwrite many intermediate files listed in Table 20.7, as a result of which the same bug would be shown even after correcting it. This is the reason why the intermediate files, if any already exists, are removed in line 1 also, prior to the compilation of the book using the commands of lines 2–6.

The compilation file ‘compile’ shown in Table 20.6 can be made executable using the ‘chmod 777 compile’ command (only once in a computer) in a command-prompt terminal, and then it can be run using the ./compile command.

20.4.2 Partial Compilation of a Book*

It is stated in §20.2 on page 192 that a `.tex` file can be included in the root file of a book using either the `\include{}` or `\input{}` command interchangeably. The `\include{}` command has an advantage over the `\input{}` command. Using the `\includeonly{}` command in the preamble, LaTeX can be instructed to compile only some selective files out of those included in the root file through the `\include{}` command, e.g., `\includeonly{intro, font, format}` can be used for compiling `intro.tex`, `font.tex`, and `format.tex` only. The `\includeonly{}` command is useful at the time of editing a large-size document like a book, where changes are to be checked in modified file(s) only. Note that if the number of pages of the document gets changed upon editing, the use of `\includeonly{}` may result in mismatched page numbers, thus restricting the use of the command in the final version of the document.