On-line Tutorial on \LaTeX

The Tutorial Team
Indian \TeX Users Group, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
2000

Prof. (Dr.) K. S. Nambooripad, Director, Center for Mathematical Sciences, Trivandrum; (Editor); Dr. E. Krishnan, Reader in Mathematics, University College, Trivandrum; Mohit Agarwal, Department of Aerospace Engineering, Indian Institute of Science, Bangalore; T. Rishi, Focal Image (India) Pvt. Ltd., Trivandrum; L. A. Ajith, Focal Image (India) Pvt. Ltd., Trivandrum; A. M. Shan, Focal Image (India) Pvt. Ltd., Trivandrum; C. V. Radhakrishnan, River Valley Technologies, Software Technology Park, Trivandrum constitute the Tutorial team.

This document is generated from \TeX sources compiled with pdf\TeX v. 1.14e in an INTEL Pentium III 700 mHz system running Linux kernel version 2.2.14-12. The packages used are hyperref.sty and pdfscreen.sty

©2000, Indian \TeX Users Group. This document may be distributed under the terms of the \TeX Project Public License, as described in lppl.txt in the base \TeX distribution, either version 1.0 or, at your option, any later version
13.1. Page layout in \LaTeX

A page in a \LaTeX document is built from various elements as shown in figure 13.1. In a two-sided document, some parameters will be different for the even and odd pages. The figure shown gives the layout as on any odd page in the document. It also shows most of the parameters required in order to change the page style including the headers, footers and the margins. We shall now briefly discuss these and the other parameters that can be effectively used to control the page layout.

- The horizontal placement of the text can be set by specifying the following parameters:

  - \texttt{\oddsidemargin} It denotes the leftside margin (on odd numbered pages). It should be noted that \texttt{\leftmargin} does not denote the leftside margin, it is instead used for the indentation of lists.

  - \texttt{\evensidemargin} It denotes the leftside margin (on even numbered pages). Note that unless the \texttt{twoside} option is chosen, the \texttt{\oddsidemargin} and the \texttt{\evensidemargin} should be the same.

  - \texttt{\textwidth} The width of the text.

- The parameters that control the vertical measurements are:

  - \texttt{\topmargin} Denotes the space between the header and the vertical offset. The latter is equal to \texttt{1in + \voffset}. \texttt{1in} is the default produced by \LaTeX.
\headheight  It denotes the height of the header.
\headsep   Refers to the distance between the header and the body of the text.
 textheight Is the height of the actual text.

The parameters that control the placement of the footer are:

\footskip  It is the distance between the body of the text and the footer.
\footheight Denotes the height of the footer.

Margin notes can be created by using the \marginpar command. The parameters controlling the margins are:

\marginparsep  Denotes the separation between the body of the text and the margin. It should be noted that in a two-sided document the margins appear on different sides on two consecutive pages.
\marginwidth Denotes the width of the margin.
\marginparpush  It is the minimum vertical separation between two marginal notes.

The commands that are needed in order to control paragraphing are:

\parskip  Denotes the vertical space between two paragraphs.
\parindent Denotes the width of paragraph indentation.
\par Equivalent to a blank line.
\topsep  It is extra vertical space (in addition to \parskip), that is added above and below list and paragraphing environments.
\itemsep  It is extra vertical space (in addition to \parskip), that is added between two list items.

The parameters defined above can be set to a particular value using the command

\setlength{parameter}{length}
Another command that can be used to change the value of a parameter by a given length is

\addtolength{parameter}{length}

### 13.1.1. Page headers and footers

The page headers and footers in \LaTeX are defined by the `\pagestyle` and `\pagenumbering` commands. The `\pagestyle` command defines the content of the headers & footers and provides the following options:

- **empty**: No headers or footers.
- **plain**: No header, footer contains the page number centered. This is the default provided by \LaTeX.
- **headings**: No footer, header contains the name of the chapter/section and/or subsection and the page number.
- **myheadings**: No footer is provided, and the header contains the page number and the information given by the `\markright` and `\markboth` commands. However, for a much better control of the headers and footers, it’s recommended to use the `fancyhdr` package.

The command `\thispagestyle` can be used to change the pagestyle of the current page in the document.

The `\pagenumbering` command defines the format of the page number. The different parameters that can be used are:

- **arabic**: roman numerals (default)
roman lower case roman numerals
Roman upper case roman numerals
alph lower case letter
Alph upper case letter $\texttt{thepage}$ produces the page number in the format defined by $\texttt{pagenumbering}$.

13.1.2. **The fancyhdr package**

The `fancyhdr` package provides another parameter for specifying the pagestyle, the `fancy` style. By use of `\pagestyle{fancy}`, one can specify three-part headers and footers. We shall illustrate it’s use with the help of some examples. The example below shows the page layout that can be created using the package `fancyhdr`.

```
<table>
<thead>
<tr>
<th>LeftHeader</th>
<th>CenteredHeader</th>
<th>RightHeader</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>page body</td>
<td></td>
</tr>
<tr>
<td>LeftFooter</td>
<td>CenteredFooter</td>
<td>RightFooter</td>
</tr>
</tbody>
</table>
```

Here is another nice example from the `fancyhdr` documentation.

```
<table>
<thead>
<tr>
<th>The performance of new graduates</th>
</tr>
</thead>
<tbody>
<tr>
<td>page body</td>
</tr>
<tr>
<td>From: K. Grant</td>
</tr>
<tr>
<td>To: Dean A. Smith</td>
</tr>
</tbody>
</table>
```
Figure 13.1  Page elements. The values shown are those in effect in the current document (on odd pages), not the default.
This is accomplished by the commands following \texttt{\textbackslash pagestyle\{fancy\}}:

\begin{verbatim}
\lhead{}
\chead{}
\rhead{\textbf{The performance of new graduates}}
\lfoot{From: K. Grant}
\cfoot{To: Dean A. Smith}
\rfoot{\texttt{thepage}}
\renewcommand{\headrulewidth}{0.4pt}
\renewcommand{\footrulewidth}{0.4pt}
\end{verbatim}

13.1.3. Using \texttt{fancyhdr} in two-sided documents

The \texttt{fancyhdr} package also provides the commands \texttt{\textbackslash fancyhead} and \texttt{\textbackslash fancyfoot} which are more general than the commands described above to define the header and the footer. These provide an additional parameter that specifies for which pages and/or parts of the header/footer those apply. The selectors that can be used are:

\begin{itemize}
  \item E Even page
  \item O Odd page
  \item L Left field
  \item C Center field
  \item R Right field
  \item H Header
  \item F Footer
\end{itemize}

Using these we can produce a two-sided document. Assuming the page layout shown above to be for the odd pages, we can have the following for the even pages:
The performance of new graduates

This can be produced by using the commands:

\fancyhead{} % clear all fields
\fancyhead[RO,LE]{{\bf The performance of new graduates}}
\fancyfoot[LE,RO]{{\thepage}}
\fancyfoot[LO,CE]{{From: K. Grant}}
\fancyfoot[CO,RE]{{To: Dean A. Smith}}
\renewcommand{\headrulewidth}{{0.4pt}}
\renewcommand{\footrulewidth}{{0.4pt}}

The default layout in fancyhdr is produced by the following commands:

\fancyhead[LE,RO]{{\slshape\rightmark}}
\fancyhead[LO,RE]{{\slshape\leftmark}}
\fancyfoot[C]{{\thepage}}

The default values for \headrulewidth and \footrulewidth are 0.4pt and 0pt respectively.
13.2. Groups

\LaTeX{} has an extremely nice feature of keeping text in groups thus enabling one to have different kinds of text wherever required. For example, one can have:

The available font sizes are:

\tiny, \scriptsize, \footnotesize, \small, \normalsize, \large, \Large, \LARGE, \huge, and \Huge.

A new group is started by the character \{ and terminated by the character \}. It is also possible to have groups nested within groups.

If some paragraphs need to be typeset in a different way (like this one!), then it is necessary to include \par or to use a blank line before closing the group, since otherwise the normal defaults will be restored before the paragraph is actually typeset.

The empty group {} enables one to get some space after \TeX{} in the output. One can also print a tilde using \~{} (this will produce ~). By using \sim{} in math mode, we get ∼.

And to quote the advice in The Not So Short Introduction to \LaTeX{} 2ε:

Remember! The \textit{MO} RE \textbf{fonts} \textbf{YOU} use in a document, \textit{the} more \textbf{READABLE} and \textbf{beautiful} it becomes.

13.3. Creating a nomenclature

In the process of writing a big document\footnote{esp. mathematical documents, theses, books etc.} which involve a number of symbols, one often feels the need to include a nomenclature for the various symbols used in the text. The nomencl package provides a convenient way of doing so. It makes use of the \textit{MakeIndex} program to generate such a list automatically by using the information provided by the author in the text.
13.3.1. Package options

The \texttt{nomenc1} package provides the following options:

- \texttt{refeq} The phrase “, see equation (⟨\textit{eq}⟩)” is appended to every entry in the nomenclature where ⟨\textit{eq}⟩ is the number of the last equation in front of the corresponding command \texttt{\nomenclature}.
- \texttt{norefeq} This is the default option; using this no equation number is printed.
- \texttt{refpage} The phrase “, see page ⟨⟨\textit{page}⟩⟩” is appended to every entry in the nomenclature where ⟨\textit{page}⟩ is the number of the page on which the corresponding command \texttt{\nomenclature} appeared.
- \texttt{norefpage} No page reference is printed; default option.
- \texttt{prefix} Every sort key is preceded by the letter “a” (changeable); default option.
- \texttt{noprefix} No prefix is used for sorting.
- \texttt{cfg} A configuration file \texttt{nomencl.cfg} is loaded, if it exists; default option.
- \texttt{nocfg} The configuration file is not loaded.

\text{croatian, danish, english, french, german, italian, polish, russian, spanish, ukranian}

The reference texts and the nomenclature title will appear in the corresponding language. In order to use Russian or Ukranian, you’ll need to have Cyrillic fonts installed and might need a replacement for \textit{MakeIndex}, e.g. \texttt{xindy}\textsuperscript{2}. The default option is english.

13.3.2. Usage and examples

The \texttt{\nomenclature} command has the following syntax:

\begin{verbatim}
\nomenclature[⟨\texttt{prefix}⟩]{⟨\texttt{symbol}⟩}{⟨\texttt{description}⟩}
\end{verbatim}

\textsuperscript{2} For more information on \texttt{xindy}, please see http://gemini.iti.informatik.tu-darmstadt.de/xindy/ or http://sourceforge.net/projects/xindy/.
where \(\text{prefix}\) is used for fine tuning the sort order, \(\text{symbol}\) is the symbol to be described, and \(\text{description}\) is the actual description. The package provides macros in order to change the referencing behavior for single entries. These macros are: \texttt{\textbackslash refeq}, \texttt{\textbackslash norefeq}, \texttt{\textbackslash refpage}, \texttt{\textbackslash norefpage}, \texttt{\textbackslash refeqpage}, \texttt{\textbackslash norefeqpage}. Note that the use of these macros locally inside the command \texttt{\textbackslash nomenclature} always supersedes the package options, and can be used in order to produce the desired effect. The following example will more clearly illustrate the usage of the package.

\begin{document}
\usepackage{nomencl}
\makeglossary
\renewcommand{\nomgroup}[1]{\ifthenelse{\equal{#1}{A}}{\item\textbf{Roman symbols}}{\ifthenelse{\equal{#1}{G}}{\item\textbf{Greek symbols}}{}}}
\begin{document}
\printglossary
\section{Dimensionless ratios of transport coefficients}
The \textit{Lewis number} is defined as
\begin{equation}
\mathrm{Le} \equiv \frac{\lambda}{\rho C_p \mathcal{D}} = \frac{\alpha}{\mathcal{D}}
\end{equation}
\nomenclature[ax ]{$\mathrm{Le}$}{Lewis number}\nomenclature[ga ]{$\lambda$}{Thermal conductivity}\nomenclature[ga ]{$\rho$}{Density}\nomenclature[a ]{$C_p$}{Constant-pressure specific heat}\nomenclature[ga ]{$\mathcal{D}$}{Mass diffusivity}\nomenclature[ga ]{$\alpha$}{Thermal diffusivity}
The \textit{Prandtl number} is defined as

\end{document}
As mentioned above, the `nomencl` package makes use of the `MakeIndex` program in order to produce the nomenclature list. On running the file through `\LaTeX`, the command `\makeglossary` instructs it to open the glossary file `⟨jobname⟩.glo` corresponding to the `\LaTeX` file `⟨jobname⟩.tex` and writes the information from the `\nomenclature` commands to this file. The next step is to invoke `MakeIndex` in order to produce the `⟨jobname⟩.gls` file. This can be achieved by making use of the command:

```
makeindex ⟨jobname⟩.glo -s nomencl.ist -o ⟨jobname⟩.gls
```

The next step is to invoke `\LaTeX` on the file `⟨jobname⟩.tex` once more. This will input the `.gls` file and process it according to the given options.

The code given in the above example produces the following nomenclature list:
Nomenclature

Roman symbols

$C_p$  Constant-pressure specific heat

Le  Lewis number

Pr  Prandtl number

Sc  Schmidt number

Greek symbols

$\alpha$  Thermal diffusivity

$\mathcal{D}$  Mass diffusivity

$\lambda$  Thermal conductivity

$\mu$  Dynamic viscosity

$\nu$  Momentum diffusivity

$\rho$  Density

13.4.  Fun with floats

13.4.1.  The subfigure package

Using this package it is possible to include several small figures and tables within a single figure or table environment. This provides a convenient way of referring the subfigures; adding entries to the table of figures is also made possible.
13.4.1.1. **Usage**

The package can be loaded by using

```
\usepackage[⟨options⟩]{subfigure}
```

in the document preamble. The various options included in the package are:

- **normal**  
  Provides ‘normal’ captions; this is the default.
- **hang**  
  Gives a hanging indentation to the caption paragraph.
- **center**  
  This causes each line of the caption paragraph to be separately centered.
- **centerlast**  
  Only the last line of the caption paragraph is centered.
- **nooneline**  
  A caption line fitting on a single line is centered by default; this option causes the same to be left-justified.

- **scriptsize, . . ., Large**  
  Sets the font size of the captions.
- **up, it, sl, sc, md, bf, rm, sf, tt**  
  Sets the font attributes of the captions.

The following commands can be used within a `figure` or `table` environment to create subfigures or subtables. The amount of vertical space between the figure and the caption can be controlled by `\subfigcapskip`. By default, this is set to `10pt`. `\subfigbottomskip` denotes the amount of vertical space added at the bottom; the default value is `10pt`.

13.4.1.2. **Examples**

The following example makes use of the `subfigure` package to put two figures side by side.
Figure 13.2 A simple example

Note that the subfigures 13.2(a) and 13.2(b) in the figure 13.2 are aligned along the bottom. These are obtained using the following code:

\begin{figure}
\centering
\subfigure[First figure]{$\label{fig-a} . . . \hspace{.75cm}$}
\subfigure[Second figure]{$\label{fig-b} . . . \}$
\caption{A simple example}
\label{two-figs}
\end{figure}

It is similarly possible to obtain tables side by side.

\begin{tabular}{|c|c|}
\hline
One & Two \\
Three & Four \\
\hline
\end{tabular}

Another small table
But slightly bigger than previous one

(a) Table 1
(b) Second table

Table 13.11 This is it!
13.4.2. Rotating figures

The `rotating` package provides the \texttt{\textbackslash rotcaption} command which makes it possible to rotate the caption thus enabling to typeset a figure in landscape mode.

\begin{verbatim}
\begin{figure}
\centering
\begin{minipage}[c]{0.6in}
\rotatebox{90}{\fcolorbox{orange}{gray10}{\myfont TEST}}
\end{minipage}
\begin{minipage}[c]{0.4in}
\rotcaption{A rotated figure.}
\end{minipage}
\end{figure}
\end{verbatim}

Another option to obtain the rotated caption is to use the command \texttt{\textbackslash rotatebox} in the same way as in the previous example and include the argument in a \texttt{\textbackslash parbox}. The `rotating` package also provides two environments `sidewaysfigure` and `sidewaystable` which are very similar to the regular `figure` and `table` environments except that these turn the contents through 90 degrees counterclockwise. The package also provides the `turn` environment that allows to rotate the contents through an arbitrary angle.

13.5. Items and lists

13.5.1. The shortlst package

The `shortlst` package is very useful for typesetting a list of short items. The regular `itemize` environment leaves
The `shortlst` package provides the following environments:

- `shortitemize`
- `shortenumerate`
- `runitemize`
- `runenumerate`

The `shortitemize` and the `shortenumerate` environments can be used for small list items in a manner very similar to the regular `itemize` and `enumerate` environments. The following example illustrates the use of `shortitemize`:

```latex
\begin{shortitemize}
\item the `\textsf{itemize}` environment
\item leaves
\item a lot
\item of
\item white space.
\end{shortitemize}
```

The environment also provides an optional argument that can be used to specify the width of the default allotment of space (the default is `65pt`). For example, using `\begin{shortitemize}[the `\textsf{itemize}` environment]` will produce:

- the `itemize` environment
- leaves
- a lot
- of
- white space.
Instead of using the optional argument, the width of the item can also be set using the command \shortitemwidth. The use of the shortenumerate environment is very similar to that of shortitemize. Both these environments can be a part of an item of a regular list environment. However, note that no list environment can be used within any of these list environments. The other two environments, runenumerate and runitemize, provided with this package can be used for items that do not need a displayed paragraph. The following example illustrates the use of the runenumerate environment:

You have three choices:
\begin{runenumerate}
\item wash your hands,
\item postpone it until tomorrow, or
\item \label{choice} stay dirty.
\end{runenumerate}

I choose \ref{choice}!

You have three choices:
(1) wash your hands, 
(2) postpone it until tomorrow, or
(3) stay dirty.
I choose 3!

The commands \parbox or \minipage can be used in case a few lists are too long to fit on a single line. The length \labelsep denotes the separation between the label and the item; and \labelwidth denotes the width of the labels. \runitemsep denotes the space between the items of a \runenumerate or \runitemize environment.

13.5.2. The \texttt{multienum} package

This package is especially useful for generating an enumerated list involving short items, e.g. the solutions manual for a text. The package provides the \texttt{multienumerate} environment that has an optional argument for enumerating even-only or odd-only arrays.

\begin{multienumerate}[]\langle\text{option}\rangle\ldots\end{multienumerate}
where the \textit{option} \texttt{evenlist} produces an enumerated array using only even numbers, the \textit{option} \texttt{oddlist} produces one using only odd numbers, and no \textit{option} produces a consecutively enumerated array. Each row of the enumerated array is set using commands of the following form:

\begin{verbatim}
\mitemx{} A single item in the row.
\mitemxx{}{} Two items in the row.
\mitemxxx{}{}{} Three items in the row.
\mitemxox{}{}{} Three items in the row with the center item space left blank so the first item can extend into its space.
\mitemxxo{}{}{} Three items in the row with the last item left blank so the second item can extend into its space.
\mitemxxxx{}{}{}{} Four items in the row.
\mitemxoxox{}{}{}{} Four items in the row with the second space left blank so the first item can extend into its space.
\mitemxxox{}{}{}{} Four items in the row with the third space left blank so the second item can extend into its space.
\mitemxxxxo{}{}{}{} Four items in the row with the last space left blank so the third item can extend into its space.
\end{verbatim}

There can be a maximum of 4 enumerated entries in a single line\textsuperscript{3}. The character \texttt{x} in the above commands refer to an entry, while the character \texttt{o} refers to a blank entry, and the space for that entry gets utilized by the previous entry.

The following example illustrates the use of the different commands that can be used to generate the enumerated list:

\begin{verbatim}
2. 3 X 2 4. 2 6. 3 8. 1 10. Not defined
\end{verbatim}

\textsuperscript{3} The example below illustrates 5 enumerated entries in a line; this is obtained by adding some simple macros in the package.
12. \[ \begin{pmatrix} -5 \\ 1 \\ 5 \end{pmatrix} \]
14. \[ \begin{pmatrix} 20 \\ -5 \end{pmatrix} \]
16. \[ \begin{pmatrix} -2 \\ 4 \\ 0 \end{pmatrix} \]
18. \[ \begin{pmatrix} 41 \\ 52 \end{pmatrix} \]
20. \[ \begin{pmatrix} 12 \\ 8 \\ 4 \end{pmatrix} \]
22. \[ \arccos\left(\frac{9}{\sqrt{85}}\right) \approx 0.22 \text{ radians} \]
24. \[ \sqrt{10} \]
26. \[ \sqrt{3} \]
28. Not defined
30. \[ x = 2 \text{ and } y = 1/2 \]
32. \[ C + A = 2\pi r + \pi r^2 \]
34. \[ \begin{pmatrix} -1 \\ 2 \end{pmatrix} \]

The code that produced the above enumerated list is given below:\footnote{The \texttt{\textbackslash mitemxxxx} and \texttt{\textbackslash mitemxoxox} commands have been defined in a similar manner to the other commands in the package.}

\begin{verbatim}
def\Matrix#1{\begin{pmatrix}#1\end{pmatrix}}
\begin{multienumerate}[evenlist]
\mitemxxxxx{3 X 2}{2}{3}{1}{Not defined}
\mitemxxxxx{\Matrix{-5 \cr 1 \cr 5}}{\Matrix{20 \cr -5}}
\mitemxxxxx{\Matrix{-2 \cr 4 \cr 0}}{\Matrix{41 \cr 52}}{\Matrix{12 \cr 8 \cr 4}}
\mitemxoxox{\arccos\left(\frac{9}{\sqrt{85}}\right) \approx 0.22 \text{ radians}}
\mitemxoxox{\sqrt{10}}{\sqrt{3}}{\text{Not defined}}
\mitemxoxox{x = 2 \text{ and } y = 1/2}{C + A = 2\pi r + \pi r^2}{\Matrix{-1 \cr 2}}
\end{multienumerate}
\end{verbatim}
13.6. Some more tricks

13.6.1. The romannum package

The `romannum` package can be used to change the numbers generated by \LaTeX\ for chapters, sections, equations, list items, footnotes, etc. from arabic to roman numerals. The package options, as described below, can be used to typeset uppercase or lowercase roman numerals.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Section</code></td>
<td>Sectional numbers in uppercase roman.</td>
</tr>
<tr>
<td><code>section</code></td>
<td>Sectional numbers in lowercase roman.</td>
</tr>
<tr>
<td><code>Equation</code></td>
<td>Equation numbers in uppercase roman.</td>
</tr>
<tr>
<td><code>equation</code></td>
<td>Equation numbers in lowercase roman.</td>
</tr>
<tr>
<td><code>Caption</code></td>
<td>Table and Figure caption numbers in uppercase roman.</td>
</tr>
<tr>
<td><code>caption</code></td>
<td>Table and Figure caption numbers in lowercase roman.</td>
</tr>
<tr>
<td><code>Footnote</code></td>
<td>Footnote numbers in uppercase roman.</td>
</tr>
<tr>
<td><code>footnote</code></td>
<td>Footnote numbers in lowercase roman.</td>
</tr>
<tr>
<td><code>Enumerate</code></td>
<td>First level items in uppercase roman and third level items in lowercase roman.</td>
</tr>
<tr>
<td><code>enumerate</code></td>
<td>First level items in lowercase roman and third level items in uppercase roman.</td>
</tr>
<tr>
<td><code>Year</code></td>
<td>The year number from the <code>\today</code> command in uppercase roman.</td>
</tr>
<tr>
<td><code>Day</code></td>
<td>The year number from the <code>\today</code> command in uppercase roman and the day number in uppercase roman.</td>
</tr>
<tr>
<td><code>day</code></td>
<td>The year number from the <code>\today</code> command in uppercase roman and the day number in uppercase roman.</td>
</tr>
<tr>
<td><code>Most</code></td>
<td>A shorthand option equivalent to using all these options: <code>Section</code>, <code>Equation</code>, <code>Caption</code>, <code>Footnote</code>, <code>Enumerate</code>; that is, all the uppecasing options except for <code>Year</code> and <code>Day</code>.</td>
</tr>
<tr>
<td><code>most</code></td>
<td>A shorthand option equivalent to using all these options: <code>section</code>, <code>equation</code>, <code>caption</code>, <code>footnote</code>, <code>enumerate</code>; that is, all the lowercasing options except for <code>day</code>.</td>
</tr>
</tbody>
</table>
13.6.2. The \texttt{epigraph} package

A good question is never answered. It is not a bolt to be tightened into place but a seed to be planted and to bear more seed toward the hope of greening the landscape of idea.

\begin{flushright}
\textit{John Ciardi}
\end{flushright}

This package provides fancy styles for typesetting quotes just after a sectional heading. The epigraphs can be typeset either at the left, the center, or the right of the typeblock. The command

\begin{verbatim}
\epigraph{⟨text⟩}{⟨source⟩}
\end{verbatim}

typesets an epigraph using \textit{⟨text⟩} as the main text of the epigraph, and the \textit{⟨source⟩} as it’s reference. The package provides the following commands:

- \textbf{\texttt{\qitem}}\texttt{⟨text⟩}{⟨source⟩} The \texttt{\qitem}{⟨text⟩}{⟨source⟩} command is used in the \texttt{epigraphs} environment in order to specify each epigraph in the list. It’s use is essentially similar to the \texttt{\item} command in the ordinary list environments.

- \textbf{\texttt{\epigraphwidth}} It denotes the width of the epigraph; the default is \texttt{0.4\textwidth}.

- \textbf{\texttt{\textflush}} It controls the \textit{⟨text⟩} typesetting style; set to \texttt{flushleft} by default.

- \textbf{\texttt{\epigraphflush}} The default position of the epigraphs is at the right hand side of the textblock (set to \texttt{flushright}). Using this command, the position of the textblock can be changed.

- \textbf{\texttt{\sourceflush}} It controls the position of the \textit{⟨source⟩}; default is \texttt{flushright}.

- \textbf{\texttt{\epigraphsize}} It can be used to redefine the fontsize in which the epigraphs are typeset; default is \texttt{small}.

- \textbf{\texttt{\epigraphrule}} This denotes the thickness of the rule drawn between the \textit{⟨text⟩} and the \textit{⟨source⟩}; default is \texttt{0.4pt}. 
\beforeepigraphskip, \afterepigraphskip

These commands control the amount of vertical space inserted before and after the typeset epigraphs; default value for both the lengths is \baselineskip.