Making Friends with LaTeX

Version 2.00

Rohit Vishal Kumar
Reader
Department of Marketing & Finance
Xavier Institute of Social Service
P.O. Box No: 7, Purulia Road
Ranchi - 834001, Jharkhand, India
Email: rohitvishalkumar@yahoo.com

October 9, 2006

Front Matter

acknowledgement

Many hours of work has gone in to making this presentation. But most of all, this presentation has benefitted from the criticisms and comments from various people.

Subha Natarajan for proof reading the earlier version of the document without knowing an iota of LaTeX . **Dan Luecking** for a detailed list of erratta which proved extremely helpful in developing this version of the document. **Kenneth Jacker** for his encouraging words and **Phil M Perry** for his slide by slide comment on the earlier version of the document. **D Venu Gopal** for sending me his excellent presentation on LaTeX.

Thanks are also due to **Hendri Adriaens and Christopher Ellison** for the `powerdot` package and **Denis Girou, Sebastian Rahtz and Timothy Van Zandt** for the `fancyvrb` package.

Finally, thanks are due to my wife, Neerajakshi, and my son, Devansh, who tolerated my late nights on computer with the outmost of patience. I could not have done this without your support. And of course, to everyone, who downloaded MFwL Version 1 and found it useful. Of course, errors will remain and the only person responsible for them, to quote Harry Potter, is *"you-know-who"*.

license

This document is primarily focussed on new users of LaTeX on Windows platform. The reason is that most of the computer users are first exposed to MS Windows and therefore to MS Word. This in turn leads to sloppy writing skills, where more time is spent on choosing fonts, colours and what-not; than on writing.

There are some conventions used in this presentation:

Code: Code is presented in a frame with line numbers as follows:

Begin Code

```
1 Code Line
2 ....
3 Code Line
```

End Code

Output: Output is presented below the following marker: `c:\output>`

Links: Hyperlinks are in blue like this

Introduction to LaTeX

the history of TEX & LATEX

- TEXwas created by Prof. Donald Knuth
- He was dissatisfied with the final proof of his *Magnum Opus* “The Art of Computer Programming”
- Started work on TEX in 1977 and the first version was released in 1982
- His efforts were supported by American Mathematical Society
- However, the TEX was not a user friendly software
- In 1985, Leslie Lamport created the LATEX interface for TEX
- LATEX was further developed by a group of people - Frank Mittelbach, David Carlisle *et. al.* and was released as LATEX 2_ε in 1994

word processors vs. LaTeX

Writing a large document has four basic stages:

- Writing the draft
- Typesetting
- Proof reading
- Making the final copy

Users of Word Processors — which are WYSIWYG in nature — normally end up writing, typesetting and to an extent, proof reading simultaneously. For example, changing the font size for headings, or spending time on trying to correct a presumed grammatical error or spelling mistake. This distracts from the main task of translating the thoughts into words.

LaTeX minimizes the distraction by taking away (almost) the task of typesetting and forcing the user to think in terms of the logical structure of the document.

pros of LaTeX

LaTeX provides the following advantages:

1. **Structured Documents:** LaTeX forces the user to think in terms of the structure of the document. So the user has to think in terms of chapters, sections, subsections etc. This leads to more organised presentation of thoughts.
2. **Consistent Formatting:** LaTeX documents are formatted consistently. All chapter, section, subsection style would be the same through out the document.
3. **Professional Output:** LaTeX produces it output in formats which render same across computers and operating systems. The default output is in DVI (Device Independent) format. It can also produce PS (Postscript) or PDF (Portable Document Format)
4. **Reusable Source:** LaTeX documents can be thought of as source-code which is compiled to get the output. As the source-code is in plain text it can be used across any operating system where LaTeX is available
5. **Extremely Extensible:** LaTeX can be extended using packages. Packages exist for doing any possible task. If none exist, user can write his own package with a little programming knowledge, thus providing flexibility to meet all possible user needs
6. **Free:** LaTeX is freely available, freely distributable and free to use

cons of LaTeX

Personally, since I discovered LaTeX I have stopped using word processors — except in office where people still require it. However it does has certain disadvantages:

1. **Not Easy to Configure:** Configuring LaTeX is not an easy task. If you wish to change styles, shapes etc. you may have to use low level commands
2. **Requires Patience:** Yes. LaTeX requires patience to learn and use. If you are addicted to the click-select-change method of Word Processors you may find it hard work

Our First LaTeX Source

our first LaTeX document

Begin Code

```
1  \documentclass[a4paper,12pt]{article}
2
3  \begin{document}
4
5  Let $D$ be a subset of $\bf R$ and let $f \colon D
6  \to \mathbf{R}$ be a real-valued function on $D$.
7  The function $f$ is said to be \emph{continuous}
8  on $D$ if, for all $\epsilon > 0$ and for all $x
9  \in D$, there exists some $\delta > 0$ (which may
10 depend on $x$) such that if $y \in D$ satisfies
11 \[ |y - x| < \delta \] then  \[ |f(y) - f(x)|
12 < \epsilon. \]
13
14 \end{document}
```

End Code

compiling the LaTeX Source

Assuming that you are in a Windows environment and already have MiKTeX you will need to do the following:

- Type the source in a text file using any text editor
- Save the file and give it a name, say, `myfile.tex`
- The default extension of LaTeX source file is `.tex`
- Now go to the command prompt and type:

Begin Command

```
15 c:\>latex myfile.tex
16 c:\>yap myfile.dvi
```

End Command

The first line tells LaTeX to process the source. If there are no errors, LaTeX produces a DVI (DeVice Independent) file which can be viewed using the program YAP (Yet Another Previewer). Under Linux, the steps are the same, except that you would probably use XDVI to preview the `.dvi` file.

The Output of the above code would be as follows:

```
c:\output>
```

Let D be a subset of $\mathbf{R}$ and let $f\colon D \to \mathbf{R}$ be a real-valued function on D. The function f is said to be *continuous* on D if, for all $\epsilon > 0$ and for all $x \in D$, there exists some $\delta > 0$ (which may depend on x) such that if $y \in D$ satisfies

$$|y - x| < \delta$$

then

$$|f(y) - f(x)| < \epsilon.$$

LaTeX Document Structure

A LaTeX document is divided in to two parts:

- PREAMBLE: Contains all formatting information, declarations about which packages to use, title of the document, authorship etc.
- BODY: contains the material that is to be typeset.

The Preamble is the contains the following commands:

Begin Code

```
17 \documentclass[option-list]{class-name}
18 \usepackage[option-list]{package-name}
19 \title{Name of the Article}
20 \author{Name of the Author(s)}
21 \date{17th November, 2005}
```

End Code

- `\documentclass[...]{class-name]` is mandatory.
- In LaTeX the following are valid document classes:
  - `book:` This class is used for typesetting books
  - `report:` This class is used for typesetting reports
  - `article:` This is used for typesetting articles
  - `letter:` This is used for writing letters
  - `slides:` This is used for making presentations
- `[...]` encloses the optional arguments, which may or may not be given. If optional arguments are not given LaTeX uses the default parameters.

documentclass: optional arguments

LaTeX provides lots of optional parameters for the documentclass.

Argument	Possible Values	Default Value
Typeface Size	10pt, 11pt, 12pt	10pt
Paper Size	a4paper, a5paper, letterpaper, legalpaper executivepaper b5paper	letterpaper
Paper Orientation	portrait, landscape	portrait
Title Page	titlepage, notitlepage	titlepage
Equation Numbering	leqno	Right side
Equation Alignment	fleqn	Centered
Output Type	draft, final	final
Layout Type	oneside, twoside	oneside
Chapter Opening	openright, openany	openright
Columns	onecolumn, twocolumn	onecolumn

NOTE: The slides class does not accepts all the above mentioned arguments. Layout Type, Chapter Opening, Columns are not available to slides class.

preamble: other options

- `\usepackage[option-list]{package-name}` is used to include the various packages that control the layout of various elements in the document. Packages normally have the extension `.sty` for style
- Various packages are available from CTAN (Comprehensive TeX Archive Network). Read the package documentation carefully before attempting to use a package
- `\title{...}` is used to insert the title of the document
- `\author{...}` is used to insert the name of the author(s) and affiliation
- `\\` is used to insert a new line
- Multiple authors are separated by `\and`
- `\date{...}` is used to insert the date. If date is not specified, LaTeX uses the current system date
- % introduces a comment till the end of the line

Here is the preamble in all its glory.

Begin Code

```
22 \documentclass[a4paper,twocolumn,draft]{article}
23 \usepackage{graphicx}        % use package graphicx
24 \usepackage{setspace}        % use package setspace
25 ..                           % add more packages
26 \title{This is my Thesis}    % This is the title
27 \author{My Name      \\      % This is my name
28 My Institution       \\      % This is my Intitute
29 ...                   \\      % Address (if you wish)
30 \and Another Author   \\      % Another author's name
31 His Institution      \\      % This is his institute
32 ...                   \\      % His Address
33 }                            % End of author block
34 \date{14 Sept 1972}          % This gives the date
```

End Code

- Body is enclosed within the following command:

Begin Code

```
35 \begin{document}     % Start of the Document
36 ...                  % Some Matter
37 \end{document}       % End of the Document
```

End Code

- LaTeX works in three different modes within the body
  - PARAGRAPH MODE: Used for processing normal text
  - MATH MODE: Used for processing Mathematical Equations and Formulas. It has three different sub-modes viz. **Math**, **Displaymath** and **Equation**
  - LEFT-RIGHT MODE: A special kind of mode used for specific purposes

paragraph mode

- In paragraph mode, LaTeX works by defining environments
- It is a special area in the document which tells LaTeX to treat the matter present in a separate manner
- Any environment is within a `\begin{environment}` and `\end{environment}` command.
- LaTeX provides numerous prespecified environments
- Environments can be customised or user defined
- We shall look at some available environments in the next section

The following special characters cannot be entered directly in the document.

Special Character	Type the Following	Output
# (hash)	`\#`	#
$ (dollar)	`\$`	$
% (percent)	`\%`	%
_ (underscore)	`\_`	_
{ (left curly brace)	`\{`	{
} (right curly brace)	`\}`	}
~ (tilde)	`\~{}`	~
^ (caret)	`\^{}`	^
\ (backslash)	`$\backslash$`	\

- `~` is normally used to put tilde accent over the following letter (Ã); hence it needs the `{}` to ensure correct display.
- `^` is normally used to put a superscript in math mode (X^A); hence it needs the `{}` to ensure correct display.
- `\` escaped with a `\` is a `\\` which is a newline insertion. To correctly get the `\` we use the `$\backslash$`

Some Environments

Alignments are useful if we want to move a block of text to left right or center. See the code and the output below:

Begin Code

```
38 \begin{flushleft}
39 This is flushed to the left
40 \end{flushleft}
41 \begin{flushright}
42 This is flushed to the Right
43 \end{flushright}
44 \begin{center}
45 And this is dead center
46 \end{center}
```

End Code

```
c:\output>
```

This is flushed to the left

This is flushed to the Right

And this is dead center

LaTeX provides three different methods for itemising your document:

- `itemize` environment produces bullets
- `enumerate` environment produces number
- `description` environment provides a labelled item
- Item within `itemize` and `enumerate` environment are preceded with the `\item` command.
- Item within `description` environment are preceded with the `\item[label]` command.
- Items can be nested within each other

bullets and lists (code)

Begin Code

```
47 \begin{itemize}
48 \item This is the an itemized item
49 \end{itemize}
50 \begin{enumerate}
51 \item This is an enumerated item
52 \end{enumerate}
53 \begin{description}
54 \item[hello] This is a description
55 \end{description}
```

End Code

```
c:\output>
```

- This is the an itemized item

1. This is an enumerated item

hello This is a description

quotation

LaTeX has provided two environment for including quotation.

- `quote` is used for inserting short quotation
- `quotation` is used for inserting longer quotation with a blank line

There is not much difference between the `quote` and the `quotation` environment; except that the margins of the `quotation` environment are indented on the left and right. Text is justified on both the margins and leaving a blank line produces a new paragraph

quotation (code)

Begin Code

```
\begin{quote}
Now go forth and conquer the World and
keep on going forth  and forth and forth
until you reach the fjord of the forth
\end{quote}
\begin{quotation}
Now go forth and conquer the World and
keep on going forth  and forth and forth
until you reach the fjord of the forth
\end{quotation}
```

End Code

```
c:\output>
```

> Now go forth and conquer the World and keep on going forth and forth and forth until you reach the fjord of the forth

> Now go forth and conquer the World and keep on going forth and forth and forth until you reach the fjord of the forth

LaTeX provides the `verse` environment for typesetting poetry. The margins are intended on the left and right and each line of the stanza is separated with `\\`. Each stanza is separated from each other by a blank line.

Begin Code

```
66 \begin{verse}
67 Early in the morning \\
68 come to me \\
69 I will teach you \\
70 A, B, C ... \\
71 \end{verse}
```

End Code

```
c:\output>
```

Early in the morning
come to me
I will teach you
A, B, C ...

The `abstract` environment is to typeset abstracts in books, reports and articles.

Begin Code

```
72 \begin{abstract}
73 This is the abstract of my pathbreaking paper
74 \end{abstract}
```

End Code

```
c:\output>
```

Abstract

This is the abstract of my pathbreaking paper

Complex Environments

tabular

LaTeX allows the creation of tabular output using the `tabular` environment. The following should be noted:

- `\begin{tabular}{no-of-columns}` required the user to specify the number of columns the environment should create.
- This alignment of each column is determined by a single alphabet - `l` (left aligned), `r` (right aligned) or `c` (centered).
- Each column entry is separated by `&` and each row by `\\`
- Horizontal lines are entered with the `\hline` command and vertical lines are inserted by `|`. Vertical lines can be entered only when the number of columns is specified.
- `\multicolumn{N}{A}{T}` command allows us to span columns; where N indicated the number of columns to span, A indicates the alignment of the column and T indicates the Text of the spanned column

tabular (code)

Begin Code

```
75 \begin{tabular}{|l|c|r|}
76 \hline
77 City        & State   & Population \\
78 \hline
79 New Delhi   & NCR     & 1,279,000  \\
80 \hline
81 Kolkata     & WB      & 1,322,000  \\
82 \hline
83 \end{tabular}
```

End Code

```
c:\output>
```

City	State	Population
New Delhi	NCR	1,279,000
Kolkata	WB	1,322,000

tabbing

LaTeX allows use of the `tabbing` environment to align text in columns. It works by setting tab stops and allow jumping between the tabs as in old fashioned typewriter. The following are useful in the `tabbing` environment.

- `\=` sets a tab stop at the current position
- `\>` advances to the next tab stop
- `\+` moves the left margin (of the next and all following commands) one tab to the right
- `\-` moves the left margin (of the next and all following commands) one tab to the left
- `\kill` sets the tab stops without producing any text
- `\pushtabs` saves all current tab position and allows temporary changing of tab stops position. A subsequent use of `\pushtabs` restores all previous tab stops

Personally i have never preferred the tabbing environment. But then, it is pretty useful at times.

Begin Code

```
84 \begin{tabbing}
85 function \= fact(n : integer) : integer;\\
86 \> begin \= \+ \\
87 \> if \= n $>$ 1 then \+ \\
88 fact := n * fact(n-1) \- \\
89 end;\\
90 \end{tabbing}
```

End Code

```
c:\output>
```

```
function fact(n : integer) : integer;
        begin
                if n > 1 then
                fact := n * fact(n-1)
        end;
```

Floating Environments

what do you mean: floating?

Yes. You heard right. LaTeX has two environments which it treats as floats — `figure` and `table`. Floating bodies are treated in special way following the given logic:

- **Step 1:** Try to place the float on the desired page
  - success - carry on and typeset the page
  - no success - place the float in a FIFO queue and typeset the page
- **Step 2:** Start a new page and check whether this page can be treated as a special float page
  - success - place as many floats from the queue here
  - no success - treat the page as a normal page and try to place the first float from the queue on this page. Any new float occurring in the text gets added to the float queue
- **Step 3** Repeat **Step 1** and **Step 2** as long as there are floats to process
  - success - Celebrate and dance with joy
  - no success - Give up and wait for the user to correct the problem

- A float cannot occur before its given position in the text
- Users can request LaTeX to try and place the float at a desired place by using the following placement specifiers:

  t place the float at the **top** of the page
  b place the float at the **bottom** of the page
  h place the float **here** please
  p place the float at in a separate **page** of floats
  ! Just **bang** it here **[Not Recommended at all]**

- The general syntax for a floating environment is as follows:

Begin Code

```
91 \begin{float-type}[placement-specifier]
92 ...
93 Matter of the float
94 ...
95 \caption[short]{A Long Caption}
96 \label{myfloat}
97 \end{float-type}
```

End Code

Tables are treated as floating objects in LaTeX . The `table` environment could be looked upon as a wrapper for the `tabular` environment. See the `table` code

Begin Code

```
98  \begin{table}[tbh]
99  \begin{tabular}{|l|c|r|}
100 \hline
101 City        & State    & Population \\
102 \hline
103 New Delhi   & NCR      & 1,279,000  \\
104 Kolkata     & WB       & 1,322,000  \\
105 \hline
106 \end{tabular}
107 \caption{My Table}
108 \label{tab:1}
109 \end{table}
```

End Code

The output of the table code is as follows:

```
c:\output>
```

City	State	Population
New Delhi	NCR	1,279,000
Kolkata	WB	1,322,000

Table 1: My Table

The table reference is produced using `\ref{label-tag}` and the page reference is produced using `\pageref{label-tag}`. So if we use the following code:

Begin Code

```
The table~\ref{tab:1} is reproduced
on page~\pageref{tab:1}
```

End Code

```
c:\output>
```

The table 1 is reproduced on page 42

figure

- Figures can be inserted in a LaTeX document using the `figure` environment
- Inserting figures require the use of `graphics` or `graphicx` package
- LaTeX can handle many types of figures — Post Script (PS), Encapsulated Post Script (EPS) being the preferred option.
- The full command sequence would be as follows:

Begin Code

```
\documentclass{article}
\usepackage{graphicx}          % MANDATORY
...
\begin{document}
\begin{figure}[htb]            % Start
\includegraphics{figure.eps} % Include figure.eps
\caption{Transistor}           % The caption
\label{fig:fig1}               % The label
\end{figure}                   % End
...
\end{document}
```

End Code

The output of the code inserts the figure transistor.eps in to the document

```
c:\output>
```

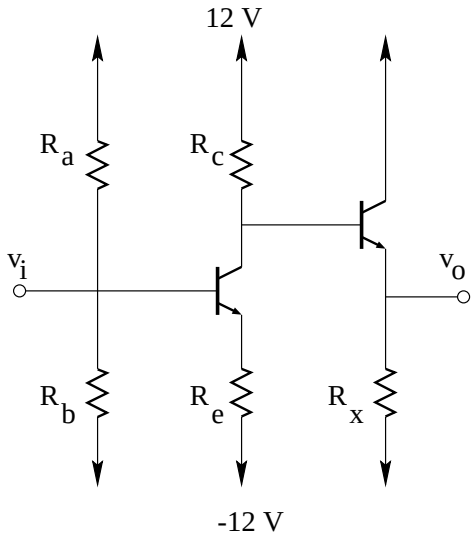
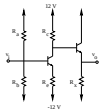


Figure 1: Transistor

You can play along with the figures too. Make it this small  by using the `\includegraphics[scale=.1]{transistor.eps}` or even rotate it. Read the `graphics` manual before attempting to insert graphics.

Working with Maths

LaTeX can be used to typeset mathematics with ease. LaTeX makes use of a special mode known as `math mode` for typesetting mathematics. In `math mode` LaTeX works using three different environments:

1. **Math Mode:** This mode is enclosed between `$` and `$`. It can also be entered using `\(` and `\)` or `\begin{math}` and `\end{math}`. This produces inline equations such as follows: $\sigma^2 = \frac{\Sigma_{i=1}^{n}(X_i - \overline{X})}{N}$.
2. **Display Math Mode:** is entered using `\begin{diaplaymath}` and `\end{displaymath}` or by using `\[` and `\]` and produces the equation in a separate line. It does not produces equation numbering.

$$\sigma^2 = \frac{\sum_{i=1}^{n}(X_i - \overline{X})}{N}$$

3. **Equation Mode:** is entered using `\begin{equation}` and `\end{equation}` and produces the equation in a separate line with an equation number.

$$\sigma^2 = \frac{\sum_{i=1}^{n}(X_i - \overline{X})}{N} \tag{1}$$

The `eqnarry` environment is used to display a series of equations. It is a three column array environment with consecutive rows separated by `\\` and consequetive items separated by `&`. It places an equation number on every line unless that line has a `\nonumber` command.

Begin Code

```
123 \begin{eqnarray}
124 \sigma & = & \sqrt{\frac{ax^2}{N}} \\
125 N \sigma^2 & = & {ax^2}
126 \end{eqnarray}
```

End Code

```
c:\output>
```

$$\sigma = \sqrt{\frac{ax^2}{N}} \tag{2}$$

$$N\sigma^2 = ax^2 \tag{3}$$

The `array` environment is similar to the `tabular` environment. The only difference is that it can only be used in a math mode.

Begin Code

```
127 \begin{equation}
128 \begin{array}{llll}
129 X_{11} & X_{12} & \cdots & X_{1n} \\
130 X_{21} & X_{22} & \cdots & X_{2n} \\
131 \vdots & \vdots & \ddots & \vdots \\
132 X_{n1} & X_{n2} & \cdots & X_{nn} \\
133 \end{array}
134 \end{equation}
```

End Code

```
c:\output>
```

$$\begin{array}{llll} X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nn} \end{array} \tag{4}$$

You want this	You type this
x^{y}	`$ x^{y} $`
x_{i}	`$ x_{i} $`
x_{1}^{y}	`$ x^{y}_{1} $`
$\frac{ax}{b}$	`$ \frac{ax}{b} $`
$\sqrt[n]{x+y}$	`$ \sqrt[n]{x + y} $`
$\overline{\overline{x}^{2}+1}$	`$ \overline{\overline{x}^{2} + 1} $`
$\overbrace{a+\underbrace{b+c}+d}$	`$ \overbrace{a + \underbrace{b+c} + d} $`
$\int_{0}^{1}x\,dx = 0$	`$ \int_{0}^{1}x dx = 0 $`

Working with BIBTeX

introduction to BIBTEX

- BIBTEX was written by Oren Patashnik
- It facilitates entering bibliographical data into LATEX documents
- It was developed along with LATEX
- Support for BIBTEXis built into all versions of LATEX
- It is an autonomous program that has to be invoked outside the main LATEX run
- An external database file (`.bib`) keeps the bibliographical records
- The BIBTEX file can keep records of the following entities:

Article	Book	Booklet
Conference	Inbook	Incollection
Inproceedings	Manual	Mastersthesis
Misc	Other	Phdthesis
Proceedings	Techreport	Unpublished

The `.bib` file contains entries in the following format:

Begin Code

```
135 @ARTICLE{RVK,
136 author={Rohit Vishal Kumar},
137 title={{Making Friends with LaTeX}},
138 journal={Journal of University},
139 year={2005},
140 volume={I},
141 pages={1--20},
142 month={September},
143 }
```

End Code

The first line identifies the type of entry `@ARTICLE{` and the citation key `RVK` For each type of entry there are some fields. Depending on the entry type some fields may be required, optional or ignored

using BIBTEX

- BIBTEX , by default, uses the numerical citation style in which the citation are numbered within `[ and ]`
- To use BIBTEX we would use the `\cite{citation-key}` in the document
- The command `\nocite{citation-key}` suppresses the citation from occurring in the text but includes it in the bibliography
- The command `\bibliography{style-name}` is used to inform BIBTEX the bibliography-style file which is to be used for formatting the bibliography
- The command `\bibilography{file-name}` is used to provide the name of the bibliography database to BIBTEX
- Support for author-date citation style is provided by various packages like `natbib`, `apacite`, `harvard`, `chicago` etc. These should be used via the `\usepackage{package-name}` command.

Some of the author-date citation packages use additional citation commands. Please read the documentation of the respective package(s) for better understanding of how the package works

The Full setup would be as follows:

Begin Code

```
\documentclass{article}
...
\begin{document}
In a recent article \cite{RVK} % Citing the entry
it was found that
...
\bibliographystyle{plain}       % Default Bib Style
\bibliography{myref}            % Bib file myref.bib
\end{document}
```

End Code

```
c:\output>
```

In a recent article [1] it was found that

References

[1] Rohit Vishal Kumar, Making Friends with LATEX , *Journal of University*, I:1–20, September, 2005

Miscellaneous

sectioning commands

- LaTeX allows for structuring the document by providing various sectioning commands
- These commands are used for numbering the various logical structures of a document

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

- `\part` and `\chapter` are not present in the article class
- `\appendix` command changes the way the sectional units like `chapter, section` are numbered in appendix

You want this	You type this
A Table of Contents	`\tableofcontents`
A List of Tables	`\listoftables`
A List of Figures	`\listoffigures`
Italics	`\textit{Italics}`
Bold Face	`\textbf{Bold Face}`
Sans Serif	`\textsf{Sans Serif}`
`Type writer style`	`\texttt{Type writer style}`
SMALL CAPS	`\textsc{Small Caps}`

extending LaTeX

- LaTeX can be extended by using packages
- More than 1000 packages exist for taking on any possible task
- It can be used to produce documents in almost any known language
- Check out CTAN for a definitive set of packages

Installing LaTeX

setting your own LaTeX system

Under the assumption, that you are on a Windows system, download the following:

MiKTeX It is the LaTeXsystem of choice under Windows. If you are a beginner, I recommend that you download the MikTeX small package. (Version: 2.4.1661 Size: 25.50 MB)

TeXnicCenter Free and preferred IDE for using LaTeX on windows. (Version: 7.01 Size: 4.43 MB)

Adobe Acrobat Reader Useful for viewing the Portable Document Format (.pdf) file created using LaTeX. (Version: 5.00 Size:8.41 MB)

Ghostscript The Ghostscript engine for producing the postscript (.ps) files created using LaTeX. (Version: 8.15 Size: 9.26 MB)

Ghostview The viewer for viewing the postscript (.ps) files created using LaTeX. (Version: 4.80 Size: 1.42 MB)

Clicking on the blue program name should take you to the website of each program. You may find newer versions on the web. Download the programs to a directory of your choice and then install them one by one in the following order: MikTeX, Ghostscript, Ghostview, Acrobat Reader and finally TeXnicCenter.

Under LINUX, teTeX is normally installed. You can use any editor of your choice to edit the `.tex` files. For installation on Operating Systems other than LINUX, UNIX or Windows, please refer to CTAN or your local guru

In the end

LaTeX error messages are of three types:

error(s) This is the most critical. It means something has gone drastically wrong. If a * is issued, that means LaTeX needs more inputs. Most of the times, errors are generated due to (i) a misspelled command (ii) a mismatched brace (iii) improper use of special character (iv) using characters or symbols which require math mode and or (v) forgetting to use the required package. Check your document carefully

warning(s) This is the second level and is less severe in nature. It normally implies that LaTeX has not been able to process the document correctly and more runs of LaTeX are required to get the cross-referencing right

bad box(es) This is the least critical. It normally implies that LaTeX is not happy with the document layout. LaTeX has a defined tolerance for typesetting paragraphs and documents. Anything which crosses this threshold generates this error. For example, Figures may be bigger than page, Hyphenation was not done properly etc.

suggested readings

1. A Gentle Introduction to TeX , A manual for self study , Michael Dobb
2. LaTeX for Word Processor Users , Guido Gonzato
3. The not So Short Introduction to LaTeX 2_ε , Or LaTeX 2_ε in 129 minutes, Tobias Oetiker
4. An Essential Guide to LaTeX 2_ε usage, Obsolete Commands and Packages, Mark Trettin. (Translated into English by Jurgen Fenn)
5. References for TeX and friends, Peter Karp and Michael Wiedmann
6. The UK TeX FAQ, Your 407 questions answered, UK TUG
7. LaTeX user guide and reference manual, Leslie Lamport, Pearson Education Asia, First Indian Reprint, 2000

Besides the above, I suggest that you subscribe to the local TeX user group (TUG) for quick answer to your queries

I shall be happy to hear from you anything regarding MFwL Version 2. Comments, Criticism, Improvements and Suggestions, all are welcome. Please send them to `rohitvishalkumar@yahoo.com`.

Thank you very much for your interest in LaTeX.

Happy TeX ing

THANK YOU