

LaTeX

COLLABORATORS

	<i>TITLE :</i> LaTeX		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 24, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	LaTeX	1
1.1	LaTeX.guide	1
1.2	LaTeX.guide/License	1
1.3	LaTeX.guide/Overview	2
1.4	LaTeX.guide/Commands	3
1.5	LaTeX.guide/Counters	4
1.6	LaTeX.guide/addtocounter	5
1.7	LaTeX.guide/alph	5
1.8	LaTeX.guide/arabic	6
1.9	LaTeX.guide/fnsymbol	6
1.10	LaTeX.guide/newcounter	6
1.11	LaTeX.guide/roman	6
1.12	LaTeX.guide/setcounter	7
1.13	LaTeX.guide/usecounter	7
1.14	LaTeX.guide/value	7
1.15	LaTeX.guide/Cross References	8
1.16	LaTeX.guide/label	8
1.17	LaTeX.guide/pageref	8
1.18	LaTeX.guide/ref	8
1.19	LaTeX.guide/Definitions	9
1.20	LaTeX.guide/newcommand	9
1.21	LaTeX.guide/newenvironment	9
1.22	LaTeX.guide/newtheorem	10
1.23	LaTeX.guide/newfont	11
1.24	LaTeX.guide/Document Styles	11
1.25	LaTeX.guide/flushbottom	12
1.26	LaTeX.guide/onecolumn	12
1.27	LaTeX.guide/raggedbottom	12
1.28	LaTeX.guide/twocolumn	12
1.29	LaTeX.guide/Environments	13

1.30	LaTeX.guide/array	14
1.31	LaTeX.guide/center	14
1.32	LaTeX.guide^centering	15
1.33	LaTeX.guide/description	15
1.34	LaTeX.guide/enumerate	15
1.35	LaTeX.guide/eqnarray	16
1.36	LaTeX.guide/equation	16
1.37	LaTeX.guide/figure	16
1.38	LaTeX.guide/flushleft	17
1.39	LaTeX.guide^raggedright	18
1.40	LaTeX.guide/flushright	18
1.41	LaTeX.guide^raggedleft	18
1.42	LaTeX.guide/itemize	19
1.43	LaTeX.guide/list	19
1.44	LaTeX.guide/minipage	19
1.45	LaTeX.guide/picture	20
1.46	LaTeX.guide^circle	22
1.47	LaTeX.guide^dashbox	22
1.48	LaTeX.guide^frame	22
1.49	LaTeX.guide^framebox (picture)	22
1.50	LaTeX.guide^line	23
1.51	LaTeX.guide^linethickness	23
1.52	LaTeX.guide^makebox (picture)	23
1.53	LaTeX.guide^multipt	24
1.54	LaTeX.guide^oval	24
1.55	LaTeX.guide^put	24
1.56	LaTeX.guide^shortstack	25
1.57	LaTeX.guide^vector	25
1.58	LaTeX.guide/quotation	25
1.59	LaTeX.guide/quote	25
1.60	LaTeX.guide/tabbing	26
1.61	LaTeX.guide^=	26
1.62	LaTeX.guide^>	27
1.63	LaTeX.guide^<	27
1.64	LaTeX.guide^+	27
1.65	LaTeX.guide^- (tabbing)	27
1.66	LaTeX.guide^A	27
1.67	LaTeX.guide^`	28
1.68	LaTeX.guide^kill	28

1.69	LaTeX.guide/table	28
1.70	LaTeX.guide/tabular	29
1.71	LaTeX.guide/\cline	30
1.72	LaTeX.guide/\hline	31
1.73	LaTeX.guide/\multicolumn	31
1.74	LaTeX.guide/\vline	31
1.75	LaTeX.guide/thebibliography	31
1.76	LaTeX.guide/\bibitem	32
1.77	LaTeX.guide/\cite	32
1.78	LaTeX.guide/\nocite	33
1.79	LaTeX.guide/theorem	33
1.80	LaTeX.guide/titlepage	33
1.81	LaTeX.guide/verbatim	33
1.82	LaTeX.guide/\verb	34
1.83	LaTeX.guide/verse	34
1.84	LaTeX.guide/Footnotes	34
1.85	LaTeX.guide/\footnote	35
1.86	LaTeX.guide/\footnotemark	35
1.87	LaTeX.guide/\footnotetext	35
1.88	LaTeX.guide/Lengths	35
1.89	LaTeX.guide/\newlength	36
1.90	LaTeX.guide/\setlength	36
1.91	LaTeX.guide/\addtolength	36
1.92	LaTeX.guide/\settowidth	37
1.93	LaTeX.guide/Letters	37
1.94	LaTeX.guide/\opening	38
1.95	LaTeX.guide/\closing	38
1.96	LaTeX.guide/Declarations	38
1.97	LaTeX.guide/\address	38
1.98	LaTeX.guide/\signature	39
1.99	LaTeX.guide/\location	39
1.100	LaTeX.guide/\telephone	39
1.101	LaTeX.guide/Line & Page Breaking	40
1.102	LaTeX.guide/\	40
1.103	LaTeX.guide/\- (hyphenation)	41
1.104	LaTeX.guide/\cleardoublepage	41
1.105	LaTeX.guide/\clearpage	41
1.106	LaTeX.guide/\hyphenation	41
1.107	LaTeX.guide/\linebreak	41

1.108	<code>LaTeX.guide/newline</code>	42
1.109	<code>LaTeX.guide/newpage</code>	42
1.110	<code>LaTeX.guide/nolinebreak</code>	42
1.111	<code>LaTeX.guide/nopagebreak</code>	43
1.112	<code>LaTeX.guide/pagebreak</code>	43
1.113	<code>LaTeX.guide/Making Paragraphs</code>	43
1.114	<code>LaTeX.guide/indent</code>	43
1.115	<code>LaTeX.guide/noindent</code>	44
1.116	<code>LaTeX.guide/par</code>	44
1.117	<code>LaTeX.guide/Math Formulae</code>	44
1.118	<code>LaTeX.guide/Subscripts & Superscripts</code>	45
1.119	<code>LaTeX.guide/Math Symbols</code>	45
1.120	<code>LaTeX.guide/Spacing in Math Mode</code>	45
1.121	<code>LaTeX.guide/Math Miscellany</code>	46
1.122	<code>LaTeX.guide/Modes</code>	47
1.123	<code>LaTeX.guide/Page Styles</code>	47
1.124	<code>LaTeX.guide/maketitle</code>	48
1.125	<code>LaTeX.guide/author</code>	48
1.126	<code>LaTeX.guide/date</code>	49
1.127	<code>LaTeX.guide/thanks</code>	49
1.128	<code>LaTeX.guide/title</code>	49
1.129	<code>LaTeX.guide/pagenumbering</code>	49
1.130	<code>LaTeX.guide/pagestyle</code>	50
1.131	<code>LaTeX.guide/mark</code>	50
1.132	<code>LaTeX.guide/thispagestyle</code>	51
1.133	<code>LaTeX.guide/Sectioning</code>	51
1.134	<code>LaTeX.guide/appendix</code>	52
1.135	<code>LaTeX.guide/Spaces & Boxes</code>	52
1.136	<code>LaTeX.guide/addvspace</code>	53
1.137	<code>LaTeX.guide/bigskip</code>	53
1.138	<code>LaTeX.guide/dotfill</code>	53
1.139	<code>LaTeX.guide/fbox</code>	54
1.140	<code>LaTeX.guide/framebox</code>	54
1.141	<code>LaTeX.guide/hfill</code>	54
1.142	<code>LaTeX.guide/hrulefill</code>	54
1.143	<code>LaTeX.guide/hspace</code>	55
1.144	<code>LaTeX.guide/makebox</code>	55
1.145	<code>LaTeX.guide/mbox</code>	56
1.146	<code>LaTeX.guide/medskip</code>	56

1.147	<code>\newsavebox</code>	56
1.148	<code>\parbox</code>	56
1.149	<code>\raisebox</code>	57
1.150	<code>\rule</code>	57
1.151	<code>\savebox</code>	57
1.152	<code>\smallskip</code>	58
1.153	<code>\usebox</code>	58
1.154	<code>\vfill</code>	58
1.155	<code>\vspace</code>	58
1.156	Special Characters	59
1.157	Splitting the Input	59
1.158	<code>\include</code>	59
1.159	<code>\includeonly</code>	60
1.160	<code>\input</code>	60
1.161	Starting & Ending	60
1.162	Table of Contents	61
1.163	<code>\addcontentsline</code>	61
1.164	<code>\addtocontents</code>	62
1.165	Terminal Input-Output	62
1.166	<code>\typeout</code>	62
1.167	<code>\typein</code>	63
1.168	Typefaces	63
1.169	Styles	63
1.170	Sizes	64
1.171	Parameters	65
1.172	List of Commands	65
1.173	<code>_{exp}</code> (subscript)	71
1.174	<code>^{exp}</code> (superscript)	71
1.175	<code>\;</code>	72
1.176	<code>\COLON</code>	72
1.177	<code>\COMMA</code>	72
1.178	<code>\!</code>	73
1.179	<code>\bf</code>	73
1.180	<code>\cdots</code>	73
1.181	<code>\ddots</code>	74
1.182	<code>\em</code>	74
1.183	<code>\footnotesize</code>	75
1.184	<code>\frac</code>	75
1.185	<code>\huge</code>	75

1.186	<code>\Huge</code> (capital `h`)	76
1.187	<code>\it</code>	76
1.188	<code>\large</code>	76
1.189	<code>\Large</code> (capital `I`)	77
1.190	<code>\LARGE</code> (all caps)	77
1.191	<code>\ldots</code>	77
1.192	<code>\normalsize</code>	78
1.193	<code>\overbrace</code>	78
1.194	<code>\overline</code>	78
1.195	<code>\rm</code>	79
1.196	<code>\sc</code>	79
1.197	<code>\scriptsize</code>	79
1.198	<code>\sf</code>	80
1.199	<code>\sl</code>	80
1.200	<code>\small</code>	81
1.201	<code>\sqrt</code>	81
1.202	<code>\tiny</code>	81
1.203	<code>\tt</code>	82
1.204	<code>\underbrace</code>	82
1.205	<code>\underline</code>	82
1.206	<code>\vdots</code>	83
1.207	<code>\Index</code>	83

Chapter 1

LaTeX

1.1 LaTeX.guide

License	Licensing information.
Overview	What is LaTeX?
Commands	Commands within a LaTeX document.
Parameters	The command line.

also available...

List of Commands	An alphabetical list of all LaTeX commands.
Index	An alphabetical "Concept Index".

1.2 LaTeX.guide/License

Licensing Information

The programs currently being distributed that relate to Texinfo include two portions of GNU Emacs, plus two other separate programs (texindex and texinfo.tex). These programs are free; this means that everyone is free to use them and free to redistribute them on a free basis. The Texinfo related programs are not in the public domain; they are copyrighted and there are restrictions on their distribution,

but these restrictions are designed to permit everything that a good cooperating citizen would want to do. What is not allowed is to try to prevent others from further sharing any version of these programs that they might get from you.

Specifically, we want to make sure that you have the right to give away copies of the programs that relate to Texinfo, that you receive source code or else can get it if you want it, that you can change these programs or use pieces of them in new free programs, and that you know you can do these things.

To make sure that everyone has such rights, we have to forbid you to deprive anyone else of these rights. For example, if you distribute copies of the Texinfo related programs, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must tell them their rights.

Also, for our own protection, we must make certain that everyone finds out that there is no warranty for the programs that relate to Texinfo. If these programs are modified by someone else and passed on, we want their recipients to know that what they have is not what we distributed, so that any problems introduced by others will not reflect on our reputation.

The precise conditions of the licensees for the programs currently being distributed that relate to Texinfo are found in the General Public Licensees that accompany them. The programs that are part of GNU Emacs are covered by the GNU Emacs copying terms (see License), and other programs are covered by licensees that are contained in their source files.

1.3 LaTeX.guide/Overview

Overview of LaTeX and Local Guide

The LaTeX command typesets a file of text using the TeX program and the LaTeX Macro package for TeX. To be more specific, it processes an input file containing the text of a document with interspersed commands that describe how the text should be formatted. It produces at least three files as output:

1. A "Device Independent", or .dvi file. This contains commands that can be translated into commands for a variety of output devices. You can view the output of LaTeX by using a program such as xdvi, which actually uses the .dvi file.
2. A "transcript" or .log file that contains summary information and diagnostic messages for any errors discovered in the input file.
3. An "auxiliary" or .aux file. This is used by LaTeX itself, for things such as sectioning.

For a description of what goes on inside TeX, you should consult 'The TeXbook' by Donald E. Knuth, ISBN 0-201-13448-9, published jointly

by the American Mathematical Society and Addison-Wesley Publishing Company.

For a description of LaTeX, you should consult:

'LaTeX: A Document Preparation System', by Leslie Lamport, ISBN 0-201-15790-X, published jointly by the American Mathematical Society and Addison-Wesley Publishing Company, 1985.

'LaTeX for Engineers & Scientists', by David J. Buerger, ISBN 0-07-008845-4, McGraw-Hill, 1990.

'The LaTeX Cookbook', by F. Teagle, Informatics Department, Rutherford Appelton Laboratory, Chilton, Didcot, Oxon, OX11 0QX, 1991.

'LaTeX for Everyone', by Jane Hahn, available from PCTeX in California, and from the TeX Users Group, 1991.

'LaTeX Line by Line: Tips and Techniques for Document Processing' by Antoni Diller, Chichester: John Wiley and Sons, 1992.

'A Guide to LaTeX', by Helmut Kopka and Patrick Daly, Addison-Wesley, 1992.

1.4 LaTeX.guide/Commands

Commands

A LaTeX command begins with the command name, which consists of a \ followed by either (a) a string of letters or (b) a single non-letter. Arguments contained in square brackets, [], are optional while arguments contained in braces, {}, are required.

NOTE: LaTeX is case sensitive. Enter all commands in lower case unless explicitly directed to do otherwise.

Counters

Information on internal counters.

Cross References

Automatic referencing.

Definitions

Define your own commands etc.

Document Styles

Some of the various styles available.

Environments

Such as enumerate & itemize.

Footnotes	How to produce footnotes.
Lengths	The length commands.
Letters	The letter style.
Line & Page Breaking	How do insert pagebreaks etc.
Making Paragraphs	Paragraph commands.
Math Formulae	How to create mathematical formulae.
Modes	Paragraph, Math or LR modes.
Page Styles	Various styles of page layout.
Sectioning	How to section properly.
Spaces & Boxes	All the associated commands.
Special Characters	Special reserved characters.
Splitting the Input	Dealing with big files by splitting.
Starting & Ending	The formal start & end layouts.
Table of Contents	How to create a table of contents.
Terminal Input-Output	User interaction.
Typefaces	Such as bold, italics etc.

1.5 LaTeX.guide/Counters

Counters

=====

Everything LaTeX numbers for you has a counter associated with it. The name of the counter is the same as the name of the environment or command that produces the number, except with no `\`. (`enumi` - `enumiv` are used for the nested enumerate environment.) Below is a list of the counters used in LaTeX's standard document styles to control numbering.

<code>part</code>	<code>paragraph</code>	<code>figure</code>	<code>enumi</code>
<code>chapter</code>	<code>subparagraph</code>	<code>table</code>	<code>enumii</code>
<code>section</code>	<code>page</code>	<code>footnote</code>	<code>enumiii</code>
<code>subsection</code>	<code>equation</code>	<code>mpfootnote</code>	<code>enumiv</code>
<code>subsubsection</code>			

`\addtocounter`

`\alph`

`\arabic`

`\fnsymbol`

`\newcounter`

`\roman`

`\setcounter`

`\usecounter`

`\value`

1.6 LaTeX.guide/\addtocounter

`\addtocounter`

```
\addtocounter{counter}{value}
```

The `\addtocounter` command increments the counter by the amount specified by the `value` argument. The `value` argument can be negative.

1.7 LaTeX.guide/\alph

`\alph`

```
\alph{counter}
```

This command causes the value of the counter to be printed in

alphabetic characters. The `\alph` command causes lower case alphabetic characters, i.e., a, b, c... while the `\Alph` command causes upper case alphabetic characters, i.e., A, B, C....

1.8 LaTeX.guide\arabic

`\arabic`

```
\arabic{counter}
```

The `\arabic` command causes the value of the counter to be printed in arabic numbers, i.e., 3.

1.9 LaTeX.guide\fnsymbol

`\fnsymbol`

```
\fnsymbol{counter}
```

The `\fnsymbol` command causes the value of the counter to be printed in a specific sequence of nine symbols that can be used for numbering footnotes.

NB. counter must have a value between 1 and 9 inclusive.

1.10 LaTeX.guide\newcounter

`\newcounter`

```
\newcounter{foo}[counter]
```

The `\newcounter` command defines a new counter named foo. The optional argument [counter] causes the counter foo to be reset whenever the counter named in the optional argument is incremented.

1.11 LaTeX.guide\roman

`\roman`

```
\roman{counter}
```

This command causes the value of the counter to be printed in roman numerals. The `\roman` command causes lower case roman numerals, i.e., `i`, `ii`, `iii...`, while the `\Roman` command causes upper case roman numerals, i.e., `I`, `II`, `III...`.

1.12 LaTeX.guide\setcounter

```
\setcounter
```

```
\setcounter{counter}{value}
```

The `\setcounter` command sets the value of the counter to that specified by the value argument.

1.13 LaTeX.guide\usecounter

```
\usecounter
```

```
\usecounter{counter}
```

The `\usecounter` command is used in the second argument of the list environment to allow the counter specified to be used to number the list items.

1.14 LaTeX.guide\value

```
\value
```

```
\value{counter}
```

The `\value` command produces the value of the counter named in the mandatory argument. It can be used where LaTeX expects an integer or number, such as the second argument of a `\setcounter` or `\addtocounter` command, or in:

```
\hspace{\value{foo}\parindent}
```

It is useful for doing arithmetic with counters.

1.15 LaTeX.guide/Cross References

Cross References

=====

One reason for numbering things like figures and equations is to refer the reader to them, as in "See Figure 3 for more details."

`\label`

`\pageref`

`\ref`

1.16 LaTeX.guide/\label

`\label`

`\label{key}`

A `\label` command appearing in ordinary text assigns to the key the number of the current sectional unit; one appearing inside a numbered environment assigns that number to the key.

A key can consist of any sequence of letters, digits, or punctuation characters. Upper and lowercase letters are different.

1.17 LaTeX.guide/\pageref

`\pageref`

`\pageref{key}`

The `\pageref` command produces the page number of the place in the text where the corresponding `\label` command appears. ie. where `\label{key}` appears.

1.18 LaTeX.guide/\ref

`\ref`

`\ref{key}`

The `\ref` command produces the number of the sectional unit, equation number, ... of the corresponding `\label` command.

1.19 LaTeX.guide/Definitions

Definitions

=====

`\newcommand`

`\newenvironment`

`\newtheorem`

`\newfont`

1.20 LaTeX.guide/\newcommand

`\newcommand`

```
\newcommand{cmd}[args]{def}
\renewcommand{cmd}[args]{def}
```

These commands define (or redefine) a command.

`cmd`

A command name beginning with a `\`. For `\newcommand` it must not be already defined and must not begin with `\end`; for `\renewcommand` it must already be defined.

`args`

An integer from 1 to 9 denoting the number of arguments of the command being defined. The default is for the command to have no arguments.

`def`

The text to be substituted for every occurrence of `cmd`; a parameter of the form `#n` in `cmd` is replaced by the text of the `n`th argument when this substitution takes place.

1.21 LaTeX.guide/\newenvironment

`\newenvironment`

```
\newenvironment{nam}[args]{begdef}{enddef}
\renewenvironment{nam}[args]{begdef}{enddef}
```

These commands define or redefine an environment.

`nam`

The name of the environment. For `\newenvironment` there must be no currently defined environment by that name, and the command `\nam` must be undefined. For `\renewenvironment` the environment must already be defined.

`args`

An integer from 1 to 9 denoting the number of arguments of the newly-defined environment. The default is no arguments.

`begdef`

The text substituted for every occurrence of `\begin{nam}`; a parameter of the form `#n` in `cmd` is replaced by the text of the `n`th argument when this substitution takes place.

`enddef`

The text substituted for every occurrence of `\end{nam}`. It may not contain any argument parameters.

1.22 LaTeX.guide/`\newtheorem`

`\newtheorem`

```
\newtheorem{env_name}{caption}[within]
\newtheorem{env_name}[numbered_like]{caption}
```

This command defines a theorem-like environment.

`env_name`

The name of the environment to be defined. A string of letters. It must not be the name of an existing environment or counter.

`caption`

The text printed at the beginning of the environment, right before the number. This may simply say "Theorem", for example.

`within`

The name of an already defined counter, usually of a sectional unit. Provides a means of resetting the new theorem counter within the sectional unit.

`numbered_like`

The name of an already defined theorem-like environment.

The `\newtheorem` command may have at most one optional argument.

1.23 LaTeX.guide/\newfont

`\newfont`

```
\newfont{cmd}{font_name}
```

Defines the command name `cmd`, which must not be currently defined, to be a declaration that selects the font named `font_name` to be the current font.

1.24 LaTeX.guide/Document Styles

Document Styles

=====

Valid LaTeX document styles include:

- * `article`
- * `report`
- * `letter`
- * `book`

Other document styles are often available. See [Overview](#), for details.

They are selected with the following command:

```
\documentstyle [options] {style}
```

The options for the different styles are:

1. `article` - `11pt`, `12pt`, `twoside`, `twocolumn`, `draft`, `fleqn`, `leqno`, `acm`
2. `report` - `11pt`, `12pt`, `twoside`, `twocolumn`, `draft`, `fleqn`, `leqno`, `acm`
3. `letter` - `11pt`, `12pt`, `fleqn`, `leqno`, `acm`
4. `book` - `11pt`, `12pt`, `twoside`, `twocolumn`, `draft`, `fleqn`, `leqno`

If you specify more than one option, they must be separated by a comma.

```
\flushbottom  
  
\onecolumn  
  
\raggedbottom  
  
\twocolumn
```

1.25 LaTeX.guide\flushbottom

```
\flushbottom  
-----
```

The `\flushbottom` declaration makes all text pages the same height, adding extra vertical space when necessary to fill out the page.

This is the standard for the book style.

1.26 LaTeX.guide\onecolumn

```
\onecolumn  
-----
```

The `\onecolumn` declaration starts a new page and produces single-column output.

1.27 LaTeX.guide\raggedbottom

```
\raggedbottom  
-----
```

The `\raggedbottom` declaration makes all pages the height of the text on that page. No extra vertical space is added.

1.28 LaTeX.guide\twocolumn

```
\twocolumn  
-----
```

The `\twocolumn` declaration starts a new page and produces two-column output.

1.29 LaTeX.guide/Environments

Environments

=====

LaTeX provides a number of different paragraph-making environments. Each environment begins and ends in the same manner.

```
\begin{environment-name}  
.  
.  
.  
\end{environment-name}
```

array

center

description

enumerate

eqnarray

equation

figure

flushleft

flushright

itemize

list

minipage

picture

quotation

quote

tabbing

table

tabular

thebibliography

theorem

```

titlepage

verbatim

verse

```

1.30 LaTeX.guide/array

array

```

\begin{array}{col1col2...coln}
column 1 entry & column 2 entry ... & column n entry \
.
.
.
\end{array}

```

Math arrays are produced with the array environment. It has a single mandatory argument describing the number of columns and the alignment within them. Each column, *coln*, is specified by a single letter that tells how items in that row should be formatted.

- * c - for centred
- * l - for flushleft
- * r - for flushright

Column entries must be separated by an &. Column entries may include other LaTeX commands. Each row of the array must be terminated with the string \ .

1.31 LaTeX.guide/center

center

```

\begin{center}
Text on line 1 \
Text on line 2 \
.
.
.
\end{center}

```

The center environment allows you to create a paragraph consisting of lines that are centred within the left and right margins on the current page. Each line must be terminated with the string \ .

`\centering`

1.32 LaTeX.guide/\centering

`\centering`

This declaration corresponds to the `center` environment. This declaration can be used inside an environment such as `quote` or in a `parbox`. The text of a figure or table can be centered on the page by putting a `\centering` command at the beginning of the figure or table environment.

Unlike the `center` environment, the `\centering` command does not start a new paragraph; it simply changes how LaTeX formats paragraph units. To affect a paragraph unit's format, the scope of the declaration must contain the blank line or `\end` command (of an environment like `quote`) that ends the paragraph unit.

1.33 LaTeX.guide/description

`description`

```
\begin{description}
\item [label] First item
\item [label] Second item
.
.
.
\end{description}
```

The `description` environment is used to make labelled lists. The label is bold face and flushed right.

1.34 LaTeX.guide/enumerate

`enumerate`

```
\begin{enumerate}
\item First item
\item Second item
.
.
```

```
.  
\end{enumerate}
```

The `enumerate` environment produces a numbered list. Enumerations can be nested within one another, up to four levels deep. They can also be nested within other paragraph-making environments.

Each item of an enumerated list begins with an `\item` command. There must be at least one `\item` command within the environment.

1.35 LaTeX.guide/eqnarray

```
eqnarray  
-----
```

```
\begin{eqnarray}  
math formula 1 \  
math formula 2 \  
.\  
.\  
.\  
\end{eqnarray}
```

The `eqnarray` environment is used to display a sequence of equations or inequalities. It is very much like a three-column array environment, with consecutive rows separated by `\` and consecutive items within a row separated by an `&`. An equation number is placed on every line unless that line has a `\nonumber` command.

1.36 LaTeX.guide/equation

```
equation  
-----
```

```
\begin{equation}  
math formula  
\end{equation}
```

The `equation` environment centres your equation on the page and places the equation number in the right margin.

1.37 LaTeX.guide/figure

```
figure  
-----
```

```
\begin{figure} [placement]
```

body of the figure

```
\caption{figure title}
\end{figure}
```

Figures are objects that are not part of the normal text, and are usually "floated" to a convenient place, like the top of a page. Figures will not be split between two pages.

The optional argument [placement] determines where LaTeX will try to place your figure. There are four places where LaTeX can possibly put a float:

1. `h` (Here) - at the position in the text where the figure environment appears.
2. `t` (Top) - at the top of a text page.
3. `b` (Bottom) - at the bottom of a text page.
4. `p` (Page of floats) - on a separate float page, which is a page containing no text, only floats.

The standard report and article styles use the default placement `tbp`.

The body of the figure is made up of whatever text, LaTeX commands, etc. you wish. The `\caption` command allows you to title your figure.

1.38 LaTeX.guide/flushleft

flushleft

```
\begin{flushleft}
Text on line 1 \
Text on line 2 \
.
.
.
\end{flushleft}
```

The `flushleft` environment allows you to create a paragraph consisting of lines that are flushed left, to the left-hand margin. Each line must be terminated with the string `\`.

`\raggedright`

1.39 LaTeX.guide/\raggedright

`\raggedright`

This declaration corresponds to the `flushleft` environment. This declaration can be used inside an environment such as `quote` or in a `parbox`.

Unlike the `flushleft` environment, the `\raggedright` command does not start a new paragraph; it simply changes how LaTeX formats paragraph units. To affect a paragraph unit's format, the scope of the declaration must contain the blank line or `\end` command (of an environment like `quote`) that ends the paragraph unit.

1.40 LaTeX.guide/flushright

`flushright`

```
\begin{flushright}
Text on line 1 \
Text on line 2 \
.
.
.
\end{flushright}
```

The `flushright` environment allows you to create a paragraph consisting of lines that are flushed right, to the right-hand margin. Each line must be terminated with the string `\ .`

`\raggedleft`

1.41 LaTeX.guide/\raggedleft

`\raggedleft`

This declaration corresponds to the `flushright` environment. This declaration can be used inside an environment such as `quote` or in a `parbox`.

Unlike the `flushright` environment, the `\raggedleft` command does not start a new paragraph; it simply changes how LaTeX formats paragraph units. To affect a paragraph unit's format, the scope of the declaration must contain the blank line or `\end` command (of an environment like `quote`) that ends the paragraph unit.

1.42 LaTeX.guide/itemize

itemize

```
\begin{itemize}
\item First item
\item Second item
.
.
.
\end{itemize}
```

The `itemize` environment produces a "bulleted" list. Itemizations can be nested within one another, up to four levels deep. They can also be nested within other paragraph-making environments.

Each item of an itemized list begins with an `\item` command. There must be at least one `\item` command within the environment.

1.43 LaTeX.guide/list

list

```
\begin{list}{label}{spacing}
\item First item
\item Second item
.
.
.
\end{list}
```

The `{label}` argument specifies how items should be labelled. This argument is a piece of text that is inserted in a box to form the label. This argument can and usually does contain other LaTeX commands.

The `{spacing}` argument contains commands to change the spacing parameters for the list. This argument will most often be null, i.e., `{}`. This will select all default spacing which should suffice for most cases.

1.44 LaTeX.guide/minipage

minipage

```
\begin{minipage}[position]{width}
  text
\end{minipage}
```

The minipage environment is similar to a `\parbox` command. It takes the same optional position argument and mandatory width argument. You may use other paragraph-making environments inside a minipage.

Footnotes in a minipage environment are handled in a way that is particularly useful for putting footnotes in figures or tables. A `\footnote` or `\footnotetext` command puts the footnote at the bottom of the minipage instead of at the bottom of the page, and it uses the `mpfootnote` counter instead of the ordinary footnote counter.

NOTE: Don't put one minipage inside another if you are using footnotes; they may wind up at the bottom of the wrong minipage.

1.45 LaTeX.guide/picture

picture

```
\begin{picture}(width,height)(x offset,y offset)
.
  picture commands
.
\end{picture}
```

The picture environment allows you to create just about any kind of picture you want containing text, lines, arrows and circles. You tell LaTeX where to put things in the picture by specifying their coordinates. A coordinate is a number that may have a decimal point and a minus sign – a number like 5, 2.3 or -3.1416. A coordinate specifies a length in multiples of the unit length `\unitlength`, so if `\unitlength` has been set to `1cm`, then the coordinate 2.54 specifies a length of 2.54 centimeters. You can change the value of `\unitlength` anywhere you want, using the `\setlength` command, but strange things will happen if you try changing it inside the picture environment.

A position is a pair of coordinates, such as $(2.4, -5)$, specifying the point with x-coordinate 2.4 and y-coordinate -5. Coordinates are specified in the usual way with respect to an origin, which is normally at the lower-left corner of the picture. Note that when a position appears as an argument, it is not enclosed in braces; the parentheses serve to delimit the argument.

The picture environment has one mandatory argument, which is a position. It specifies the size of the picture. The environment produces a rectangular box with width and height determined by this argument's x- and y-coordinates.

The `picture` environment also has an optional position argument, following the size argument, that can change the origin. (Unlike ordinary optional arguments, this argument is not contained in square brackets.) The optional argument gives the coordinates of the point at the lower-left corner of the picture (thereby determining the origin). For example, if `\unitlength` has been set to `1mm`, the command...

```
\begin{picture}(100,200)(10,20)
```

...produces a picture of width 100 millimeters and height 200 millimeters, whose lower-left corner is the point `(10,20)` and whose upper-right corner is therefore the point `(110,220)`. When you first draw a picture, you will omit the optional argument, leaving the origin at the lower-left corner. If you then want to modify your picture by shifting everything, you just add the appropriate optional argument.

The environment's mandatory argument determines the nominal size of the picture. This need bear no relation to how large the picture really is; LaTeX will happily allow you to put things outside the picture, or even off the page. The picture's nominal size is used by TeX in determining how much room to leave for it.

Everything that appears in a picture is drawn by the `\put` command. The command...

```
\put (11.3,-.3){...}
```

...puts the object specified by `...` in the picture, with its reference point at coordinates `(11.3,-.3)`. The reference points for various objects will be described below.

The `\put` command creates an LR box. You can put anything in the text argument of the `\put` command that you'd put into the argument of an `\mbox` and related commands. When you do this, the reference point will be the lower left corner of the box.

```
\circle  
  
\dashbox  
  
\frame  
  
\framebox (picture)  
  
\line  
  
\linethickness  
  
\makebox (picture)  
  
\multipt  
  
\oval  
  
\put  
  
\shortstack
```

`\vector`

1.46 LaTeX.guide\circle

`\circle`
.....

`\circle[*]{diameter}`

The `\circle` command produces a circle of the specified diameter. If the `*`-form of the command is used, LaTeX draws a solid circle.

1.47 LaTeX.guide\dashbox

`\dashbox`
.....

`\dashbox{dash_length}(width,height){...}`

The `\dashbox` has an extra argument which specifies the width of each dash. A dashed box looks best when the width and height are multiples of the `dash_length`.

1.48 LaTeX.guide\frame

`\frame`
.....

`\frame{...}`

The `\frame` command puts a rectangular frame around the object specified in the argument. The reference point is the bottom left corner of the frame. No extra space is put between the frame and the object.

1.49 LaTeX.guide\framebox (picture)

`\framebox`
.....

`\framebox(width,height) [position]{...}`

The `\framebox` command is exactly the same as the `\makebox` command,

except that it puts a frame around the outside of the box that it creates.

The `framebox` command produces a rule of thickness `\fboxrule`, and leaves a space `\fboxsep` between the rule and the contents of the box.

1.50 LaTeX.guide/\line

`\line`
.....

```
\line(x slope,y slope){length}
```

The `\line` command draws a line of the specified length and slope.

1.51 LaTeX.guide/\linethickness

`\linethickness`
.....

```
\linethickness{dimension}
```

Declares the thickness of horizontal and vertical lines in a picture environment to be `dimension`, which must be a positive length. It does not affect the thickness of slanted lines and circles, or the quarter circles drawn by `\oval` to form the corners of an oval.

1.52 LaTeX.guide/\makebox (picture)

`\makebox`
.....

```
\makebox(width,height) [position]{...}
```

The `\makebox` command for the picture environment is similar to the normal `\makebox` command except that you must specify a width and height in multiples of `\unitlength`.

The optional argument, `[position]`, specifies the quadrant that your text appears in. You may select up to two of the following:

- * `t` - Moves the item to the top of the rectangle
- * `b` - Moves the item to the bottom
- * `l` - Moves the item to the left

* r - Moves the item to the right

See

`\makebox`

.

1.53 LaTeX.guide\multiput

`\multiput`

.....

`\multiput(x coord,y coord)(delta x,delta y){number of copies}{object}`

The `\multiput` command can be used when you are putting the same object in a regular pattern across a picture.

1.54 LaTeX.guide\oval

`\oval`

.....

`\oval(width,height)[portion]`

The `\oval` command produces a rectangle with rounded corners. The optional argument, `[portion]`, allows you to select part of the oval.

* t - Selects the top portion

* b - Selects the bottom portion

* r - Selects the right portion

* l - Selects the left portion

1.55 LaTeX.guide\put

`\put`

.....

`\put(x coord,y coord){ ... }`

The `\put` command places the item specified by the mandatory argument at the given coordinates.

1.56 LaTeX.guide\shortstack

\shortstack
.....

```
\shortstack[position]{... \ ... \ ...}
```

The \shortstack command produces a stack of objects. The valid positions are:

- * r - Moves the objects to the right of the stack
- * l - Moves the objects to the left of the stack
- * c - Moves the objects to the center of the stack (default)

1.57 LaTeX.guide\vector

\vector
.....

```
\vector(x slope,y slope){length}
```

The \vector command draws a line with an arrow of the specified length and slope. The x and y values must lie between -4 and +4, inclusive.

1.58 LaTeX.guide/quotation

quotation

```
\begin{quotation}
  text
\end{quotation}
```

The margins of the quotation environment are indented on the left and the right. The text is justified at both margins and there is paragraph indentation. Leaving a blank line between text produces a new paragraph.

1.59 LaTeX.guide/quote

quote

```
\begin{quote}
text
\end{quote}
```

The margins of the quote environment are indented on the left and the right. The text is justified at both margins. Leaving a blank line between text produces a new paragraph.

1.60 LaTeX.guide/tabbing

tabbing

```
\begin{tabbing}
text \= more text \= still more text \= last text \
second row \> \> more \
.
.
.
\end{tabbing}
```

The tabbing environment provides a way to align text in columns. It works by setting tab stops and tabbing to them much the way you do with an ordinary typewriter.

```
\=
\>
\<
\+
\+ (tabbing)
\A
\`
\kill
```

1.61 LaTeX.guide/\=

```
\=
..
```

The `\=` command sets the tab stops.

1.62 LaTeX.guide/\>

\>
..

The \> command causes LaTeX to advance to the next tab stop.

1.63 LaTeX.guide/\<

\<
..

The \< command allows you to put something to the left of the local margin without changing the margin.

1.64 LaTeX.guide/\+

\+
..

The \+ command moves the left margin of the next and all the following commands one tab stop to the right.

1.65 LaTeX.guide/\- (tabbing)

\-
..

The \- command moves the left margin of the next and all the following commands one tab stop to the left.

1.66 LaTeX.guide/\A

\'
..

The \' command moves everything that you have typed so far in the current column, everything starting from the most recent \>, \<, \', \, or \kill command, to the right of the previous column, flush against the current column's tab stop.

1.67 LaTeX.guide/\`

```
\`
..
```

The `\`` command allows you to put text flushed right against any tab stop, including tab stop 0. However, it can't move text to the right of the last column because there's no tab stop there. The `\`` command moves all the text that follows it, up to the `\` or `\end{tabbing}` command that ends the line, to the right margin of the tabbing environment. There must be no `\>` or `\'` command between the `\`` and the command that ends the line.

1.68 LaTeX.guide/\kill

```
\kill
.....
```

The `\kill` command allows you to set tab stops without producing text. It works just like the `\`` except that it throws away the current line instead of producing output for it. The effect of any `\=`, `\+` or `\-` commands in that line remain in effect.

1.69 LaTeX.guide/table

```
table
-----
```

```
\begin{table}[placement]

  body of the table

\caption{table title}
\end{table}
```

Tables are objects that are not part of the normal text, and are usually "floated" to a convenient place, like the top of a page. Tables will not be split between two pages.

The optional argument `[placement]` determines where LaTeX will try to place your table. There are four places where LaTeX can possibly put a float:

- * `h` : Here - at the position in the text where the table environment appears.
- * `t` : Top - at the top of a text page.
- * `b` : Bottom - at the bottom of a text page.

- * `p` : Page of floats - on a separate float page, which is a page containing no text, only floats.

The standard report and article styles use the default placement [tbp].

The body of the table is made up of whatever text, LaTeX commands, etc., you wish. The `\caption` command allows you to title your table.

1.70 LaTeX.guide/tabular

tabular

```
\begin{tabular}[pos]{cols}
column 1 entry & column 2 entry ... & column n entry \
.
.
.
\end{tabular}
```

or

```
\begin{tabular*}{width}[pos]{cols}
column 1 entry & column 2 entry ... & column n entry \
.
.
.
\end{tabular*}
```

These environments produce a box consisting of a sequence of rows of items, aligned vertically in columns. The mandatory and optional arguments consist of:

`width`

Specifies the width of the `tabular*` environment. There must be rubber space between columns that can stretch to fill out the specified width.

`pos`

Specifies the vertical position; default is alignment on the center of the environment.

- * `t` - align on top row

- * `b` - align on bottom row

`cols`

Specifies the column formatting. It consists of a sequence of the following specifiers, corresponding to the sequence of columns and intercolumn material.

- * `l` - A column of left-aligned items.

- * `r` - A column of right-aligned items.
- * `c` - A column of centred items.
- * `|` - A vertical line the full height and depth of the environment.
- * `@` - This inserts text in every row. An `@`-expression suppresses the intercolumn space normally inserted between columns; any desired space between the inserted text and the adjacent items must be included in text. An `\extracolsep{wd}` command in an `@`-expression causes an extra space of width `wd` to appear to the left of all subsequent columns, until countermanded by another `\extracolsep` command. Unlike ordinary intercolumn space, this extra space is not suppressed by an `@`-expression. An `\extracolsep` command can be used only in an `@`-expression in the `cols` argument.
- * `p{wd}` - Produces a column with each item typeset in a parbox of width `wd`, as if it were the argument of a `\parbox[t]{wd}` command. However, a `\` may not appear in the item, except in the following situations:
 1. inside an environment like `minipage`, `array`, or `tabular`.
 2. inside an explicit `\parbox`.
 3. in the scope of a `\centering`, `\raggedright`, or `\raggedleft` declaration. The latter declarations must appear inside braces or an environment when used in a `p`-column element.
- * `*{num}{cols}` - Equivalent to `num` copies of `cols`, where `num` is any positive integer and `cols` is any list of column-specifiers, which may contain another `*`-expression.

`\cline`

`\hline`

`\multicolumn`

`\vline`

1.71 LaTeX.guide/\cline

`\cline`

.....

`\cline{i-j}`

The `\cline` command draws horizontal lines across the columns

specified, beginning in column *i* and ending in column *j*, which are identified in the mandatory argument.

1.72 LaTeX.guide/hline

`\hline`
.....

The `\hline` command will draw a horizontal line the width of the table. It's most commonly used to draw a line at the top, bottom, and between the rows of the table.

1.73 LaTeX.guide/multicolumn

`\multicolumn`
.....

```
\multicolumn{cols}{pos}{text}
```

The `\multicolumn` is used to make an entry that spans several columns. The first mandatory argument, *cols*, specifies the number of columns to span. The second mandatory argument, *pos*, specifies the formatting of the entry; *c* for centred, *l* for flushleft, *r* for flushright. The third mandatory argument, *text*, specifies what text is to make up the entry.

1.74 LaTeX.guide/vline

`\vline`
.....

The `\vline` command will draw a vertical line extending the full height and depth of its row. An `\hfill` command can be used to move the line to the edge of the column. It can also be used in an @-expression.

1.75 LaTeX.guide/thebibliography

thebibliography

```
\begin{thebibliography}{widest-label}
\bibitem[label]{cite_key}
```

.

```
.
.
\end{thebibliography}
```

The `thebibliography` environment produces a bibliography or reference list. In the article style, this reference list is labelled "References"; in the report style, it is labelled "Bibliography".

- * `widest-label`: Text that, when printed, is approximately as wide as the widest item label produces by the `\bibitem` commands.

```
\bibitem
\cite
\nocite
```

1.76 LaTeX.guide/\bibitem

```
\bibitem
.....
```

```
\bibitem[label]{cite_key}
```

The `\bibitem` command generates an entry labelled by `label`. If the `label` argument is missing, a number is generated as the label, using the `enumi` counter. The `cite_key` is any sequence of letters, numbers, and punctuation symbols not containing a comma. This command writes an entry on the `.aux` file containing `cite_key` and the item's label. When this `.aux` file is read by the `\begin{document}` command, the item's label is associated with `cite_key`, causing the reference to `cite_key` by a `\cite` command to produce the associated label.

1.77 LaTeX.guide/\cite

```
\cite
.....
```

```
\cite[text]{key_list}
```

The `key_list` argument is a list of citation keys. This command generates an in-text citation to the references associated with the keys in `key_list` by entries on the `.aux` file read by the `\begin{document}` command.

1.78 LaTeX.guide/\nocite

\nocite
.....

```
\nocite{key_list}
```

The \nocite command produces no text, but writes key_list, which is a list of one or more citation keys, on the .aux file.

1.79 LaTeX.guide/theorem

theorem

```
\begin{theorem}  
theorem text  
\end{theorem}
```

The theorem environment produces "Theorem x" in boldface followed by your theorem text.

1.80 LaTeX.guide/titlepage

titlepage

```
\begin{titlepage}  
text  
\end{titlepage}
```

The titlepage environment creates a title page, i.e. a page with no printed page number or heading. It also causes the following page to be numbered page one. Formatting the title page is left to you. The \today command comes in handy for title pages.

1.81 LaTeX.guide/verbatim

verbatim

```
\begin{verbatim}  
text  
\end{verbatim}
```

The verbatim environment is a paragraph-making environment that gets

LaTeX to print exactly what you type in. It turns LaTeX into a typewriter with carriage returns and blanks having the same effect that they would on a typewriter.

```
\verb
```

1.82 LaTeX.guide/\verb

```
\verb
.....
```

```
\verb char literal_text char \verb*char literal_text char
```

Typesets `literal_text` exactly as typed, including special characters and spaces, using a typewriter (`\tt`) type style. There may be no space between `\verb` or `\verb*` and `char` (space is shown here only for clarity). The `*`-form differs only in that spaces are printed.

1.83 LaTeX.guide/verse

```
verse
-----
```

```
\begin{verse}
  text
\end{verse}
```

The `verse` environment is designed for poetry, though you may find other uses for it.

1.84 LaTeX.guide/Footnotes

```
Footnotes
=====
```

Footnotes can be produced in one of two ways. They can be produced with one command, the `\footnote` command. They can also be produced with two commands, the `\footnotemark` and the `\footnotetext` commands. See the specific command for information on why you would use one over the other.

```
\footnote
```

```
\footnotemark
```

```
\footnotetext
```

1.85 LaTeX.guide/\footnote

```
\footnote
```

```
-----
```

```
\footnote[number]{text}
```

The `\footnote` command places the numbered footnote text at the bottom of the current page. The optional argument, *number*, is used to change the default footnote number. This command can only be used in outer paragraph mode.

1.86 LaTeX.guide/\footnotemark

```
\footnotemark
```

```
-----
```

The `\footnotemark` command puts the footnote number in the text. This command can be used in inner paragraph mode. The text of the footnote is supplied by the `\footnotetext` command.

1.87 LaTeX.guide/\footnotetext

```
\footnotetext
```

```
-----
```

```
\footnotetext[number]{text}
```

The `\footnotetext` command produces the text to be placed at the bottom of the page. This command can come anywhere after the `\footnotemark` command. The `\footnotetext` command must appear in outer paragraph mode.

The optional argument, *number*, is used to change the default footnote number.

1.88 LaTeX.guide/Lengths

Lengths

=====

A length is a measure of distance. Many LaTeX commands take a length as an argument.

```
\newlength
\setlength
\addtolength
\settowidth
```

1.89 LaTeX.guide/\newlength

```
\newlength
-----
```

```
\newlength{\gnat}
```

The `\newlength` command defines the mandatory argument, `\gnat`, as a length command with a value of 0in. An error occurs if a `\gnat` command already exists.

1.90 LaTeX.guide/\setlength

```
\setlength
-----
```

```
\setlength{\gnat}{length}
```

The `\setlength` command is used to set the value of a length command. The length argument can be expressed in any terms of length LaTeX understands, i.e., inches (in), millimeters (mm), points (pt), etc.

1.91 LaTeX.guide/\addtolength

```
\addtolength
-----
```

```
\addtolength{\gnat}{length}
```

The `\addtolength` command increments a "length command" by the amount

specified in the length argument. It can be a negative amount.

1.92 LaTeX.guide/\settowidth

`\settowidth`

```
\settowidth{\gnat}{text}
```

The `\settowidth` command sets the value of a length command equal to the width of the text argument.

1.93 LaTeX.guide/Letters

Letters

=====

You can use LaTeX to typeset letters, both personal and business. The letter document style is designed to make a number of letters at once, although you can make just one if you so desire.

Your `.tex` source file has the same minimum commands as the other document styles, i.e., you must have the following commands as a minimum:

```
\documentstyle{letter}
\begin{document}
... letters ...
\end{document}
```

Each letter is a letter environment, whose argument is the name and address of the recipient. For example, you might have:

```
\begin{letter}{Mr. Joe Smith\ 2345 Princess St.
\ Edinburgh, EH1 1AA}
...
\end{letter}
```

The letter itself begins with the `\opening` command. The text of the letter follows. It is typed as ordinary LaTeX input. Commands that make no sense in a letter, like `\chapter`, don't work. The letter closes with a `\closing` command.

After the closing, you can have additional material. The `\cc` command produces the usual "cc: ...". There's also a similar `\encl` command for a list of enclosures.

`\opening`

`\closing`

Declarations

1.94 LaTeX.guide\opening`\opening`

`\opening{text}`

The letter begins with the `\opening` command. The mandatory argument, `text`, is whatever text you wish to start your letter, i.e.,

`\opening{Dear Joe,}`**1.95 LaTeX.guide\closing**`\closing`

`\closing{text}`

The letter closes with a `\closing` command, i.e.,

`\closing{Best Regards,}`**1.96 LaTeX.guide/Declarations**

Declarations

The following commands are declarations which take a single argument.

`\address``\signature``\location``\telephone`**1.97 LaTeX.guide\address**

`\address`

```
\address{Return address}
```

The return address, as it should appear on the letter and the envelope. Separate lines of the address should be separated by `\` commands. If you do not make an `\address` declaration, then the letter will be formatted for copying onto your organisation's standard letterhead. (See

Overview

, for details on your local implementation).

If you give an `\address` declaration, then the letter will be formatted as a personal letter.

1.98 LaTeX.guide\`\signature`

`\signature`

```
\signature{Your name}
```

Your name, as it should appear at the end of the letter underneath the space for your signature. Items that should go on separate lines should be separated by `\` commands.

1.99 LaTeX.guide\`\location`

`\location`

```
\location{address}
```

This modifies your organisation's standard address. This only appears if the `firstpage` `pagestyle` is selected.

1.100 LaTeX.guide\`\telephone`

`\telephone`

```
\telephone{number}
```

This is your telephone number. This only appears if the `firstpage` `pagestyle` is selected.

1.101 LaTeX.guide/Line & Page Breaking

Line & Page Breaking

=====

The first thing LaTeX does when processing ordinary text is to translate your input file into a string of glyphs and spaces. To produce a printed document, this string must be broken into lines, and these lines must be broken into pages. In some environments, you do the line breaking yourself with the `\` command, but LaTeX usually does it for you.

```
\\
\- (hyphenation)
\cleardoublepage
\clearpage
\hyphenation
\linebreak
\newline
\newpage
\nolinebreak
\nopagebreak
\pagebreak
```

1.102 LaTeX.guide/\

```
\
--
```

```
\[*][extra-space]
```

The `\` command tells LaTeX to start a new line. It has an optional argument, `extra-space`, that specifies how much extra vertical space is to be inserted before the next line. This can be a negative amount.

The `*` command is the same as the ordinary `\` command except that it tells LaTeX not to start a new page after the line.

1.103 LaTeX.guide/\- (hyphenation)

```
\-
--
```

The `\-` command tells LaTeX that it may hyphenate the word at that point. LaTeX is very good at hyphenating, and it will usually find all correct hyphenation points. The `\-` command is used for the exceptional cases.

1.104 LaTeX.guide/\cleardoublepage

```
\cleardoublepage
-----
```

The `\cleardoublepage` command ends the current page and causes all figures and tables that have so far appeared in the input to be printed. In a two-sided printing style, it also makes the next page a right-hand (odd-numbered) page, producing a blank page if necessary.

1.105 LaTeX.guide/\clearpage

```
\clearpage
-----
```

The `\clearpage` command ends the current page and causes all figures and tables that have so far appeared in the input to be printed.

1.106 LaTeX.guide/\hyphenation

```
\hyphenation
-----
```

```
\hyphenation{words}
```

The `\hyphenation` command declares allowed hyphenation points, where `words` is a list of words, separated by spaces, in which each hyphenation point is indicated by a `-` character.

1.107 LaTeX.guide/\linebreak

`\linebreak`

`\linebreak[number]`

The `\linebreak` command tells LaTeX to break the current line at the point of the command. With the optional argument, `number`, you can convert the `\linebreak` command from a demand to a request. The number must be a number from 0 to 4. The higher the number, the more insistent the request is.

The `\linebreak` command causes LaTeX to stretch the line so it extends to the right margin.

1.108 LaTeX.guide/\newline

`\newline`

The `\newline` command breaks the line right where it is. The `\newline` command can be used only in paragraph mode.

1.109 LaTeX.guide/\newpage

`\newpage`

The `\newpage` command ends the current page.

1.110 LaTeX.guide/\nolinebreak

`\nolinebreak`

`\nolinebreak[number]`

The `\nolinebreak` command prevents LaTeX from breaking the current line at the point of the command. With the optional argument, `number`, you can convert the `\nolinebreak` command from a demand to a request. The number must be a number from 0 to 4. The higher the number, the more insistent the request is.

1.111 LaTeX.guide/\nopagebreak

\nopagebreak

\nopagebreak[number]

The \nopagebreak command prevents LaTeX from breaking the current page at the point of the command. With the optional argument, *number*, you can convert the \nopagebreak command from a demand to a request. The number must be a number from 0 to 4. The higher the number, the more insistent the request is.

1.112 LaTeX.guide/\pagebreak

\pagebreak

\pagebreak[number]

The \pagebreak command tells LaTeX to break the current page at the point of the command. With the optional argument, *number*, you can convert the \pagebreak command from a demand to a request. The number must be a number from 0 to 4. The higher the number, the more insistent the request is.

1.113 LaTeX.guide/Making Paragraphs

Making Paragraphs

=====

A paragraph is ended by one or more completely blank lines - lines not containing even an %. A blank line should not appear where a new paragraph cannot be started, such as in math mode or in the argument of a sectioning command.

\indent

\noindent

\par

1.114 LaTeX.guide/\indent

`\indent`

`\indent`

This produces a horizontal space whose width equals the width of the paragraph indentation. It is used to add paragraph indentation where it would otherwise be suppressed.

1.115 LaTeX.guide/\noindent

`\noindent`

`\noindent`

When used at the beginning of the paragraph, it suppresses the paragraph indentation. It has no effect when used in the middle of a paragraph.

1.116 LaTeX.guide/\par

`\par`

Equivalent to a blank line; often used to make command or environment definitions easier to read.

1.117 LaTeX.guide/Math Formulae

Math Formulae

=====

There are three environments that put LaTeX in math mode:

`math`

For Formulae that appear right in the text.

`displaymath`

For Formulae that appear on their own line.

`equation`

The same as the `displaymath` environment except that it adds an equation number in the right margin.

The `math` environment can be used in both paragraph and LR mode, but the `displaymath` and `equation` environments can be used only in paragraph

mode. The `math` and `displaymath` environments are used so often that they have the following short forms:

`\(...\)` instead of `\begin{math}...\end{math}`

`\[...\]` instead of `\begin{displaymath}...\end{displaymath}`

In fact, the `math` environment is so common that it has an even shorter form:

`$... $` instead of `\(...\)`

Subscripts & Superscripts

Math Symbols

Spacing in Math Mode

Math Miscellany

1.118 LaTeX.guide/Subscripts & Superscripts

Subscripts & Superscripts

To get an expression `exp` to appear as a subscript, you just type `_{exp}`. To get `exp` to appear as a superscript, you type `^{exp}`. LaTeX handles superscripted superscripts and all of that stuff in the natural way. It even does the right thing when something has both a subscript and a superscript.

1.119 LaTeX.guide/Math Symbols

Math Symbols

LaTeX provides almost any mathematical symbol you're likely to need. The commands for generating them can be used only in math mode. For example, if you include `$$\pi$` in your source, you will get the symbol "pi" in your output.

1.120 LaTeX.guide/Spacing in Math Mode

Spacing in Math Mode

In a math environment, LaTeX ignores the spaces you type and puts in

the spacing that it thinks is best. LaTeX formats mathematics the way it's done in mathematics texts. If you want different spacing, LaTeX provides the following four commands for use in math mode:

1. `\;` - a thick space
2. `\:` - a medium space
3. `\,` - a thin space
4. `\!` - a negative thin space

1.121 LaTeX.guide/Math Miscellany

Math Miscellany

`\cdots`

The `\cdots` command produces a horizontal ellipsis where the dots are raised to the center of the line.

`\ddots`

The `\ddots` command produces a diagonal ellipsis.

`\frac`

`\frac{num}{den}` The `\frac` command produces the fraction num divided by den.

`\ldots`

The `\ldots` command produces an ellipsis. This command works in any mode, not just math mode.

`\overbrace`

`\overbrace{text}` The `\overbrace` command generates a brace over text.

`\overline`

`\overline{text}` The `\overline` command causes the argument text to be overlined.

`\sqrt`

`\sqrt[root]{arg}` The `\sqrt` command produces the square root of its argument. The optional argument, *root*, determines what root to produce, i.e., the cube root of $x+y$ would be typed as `\sqrt[3]{x+y}`.

`\underbrace`

`\underbrace{text}` The `\underbrace` command generates text with a brace underneath.

`\underline`

`\underline{text}` The `\underline` command causes the argument text to be underlined. This command can also be used in paragraph and LR modes.

`\vdots`

The `\vdots` command produces a vertical ellipsis.

1.122 LaTeX.guide/Modes

Modes

=====

When LaTeX is processing your input text, it is always in one of three modes:

- * Paragraph mode
- * Math mode
- * Left-to-right mode, called LR mode for short

LaTeX changes mode only when it goes up or down a staircase to a different level, though not all level changes produce mode changes. Mode changes occur only when entering or leaving an environment, or when LaTeX is processing the argument of certain text-producing commands.

"Paragraph mode" is the most common; it's the one LaTeX is in when processing ordinary text. In that mode, LaTeX breaks your text into lines and breaks the lines into pages. LaTeX is in "math mode" when it's generating a mathematical formula. In "LR mode", as in paragraph mode, LaTeX considers the output that it produces to be a string of words with spaces between them. However, unlike paragraph mode, LaTeX keeps going from left to right; it never starts a new line in LR mode. Even if you put a hundred words into an `\mbox`, LaTeX would keep typesetting them from left to right inside a single box, and then complain because the resulting box was too wide to fit on the line.

LaTeX is in LR mode when it starts making a box with an `\mbox` command. You can get it to enter a different mode inside the box - for example, you can make it enter math mode to put a formula in the box. There are also several text-producing commands and environments for making a box that put LaTeX in paragraph mode. The box made by one of these commands or environments will be called a parbox. When LaTeX is in paragraph mode while making a box, it is said to be in "inner paragraph mode". Its normal paragraph mode, which it starts out in, is called "outer paragraph mode".

1.123 LaTeX.guide/Page Styles

Page Styles

=====

The `\documentstyle` command determines the size and position of the page's head and foot. The page style determines what goes in them.

```

\maketitle

\pagenumbering

\pagestyle

\thispagestyle

```

1.124 LaTeX.guide/maketitle

```

\maketitle

```

```

\maketitle

```

The `\maketitle` command generates a title on a separate title page - except in the article style, where the title normally goes at the top of the first page. Information used to produce the title is obtained from the following declarations:

See

```

Page Styles
for the commands to give the information.

```

```

\author

\date

\thanks

\title

```

1.125 LaTeX.guide/author

```

\author
-----

```

```

\author{names}

```

The `\author` command declares the author(s), where `names` is a list of authors separated by `\and` commands. Use `\` to separate lines within a single author's entry - for example, to give the author's institution or address.

NOTE: The `milstd` and `book-form` styles have re-defined the `\maketitle`

command. The `\title` declaration is the only command of those shown below that has any meaning.

1.126 LaTeX.guide/\`date`

`\date`

```
\date{text}
```

The `\date` command declares `text` to be the document's date. With no `\date` command, the current date is used.

1.127 LaTeX.guide/\`thanks`

`\thanks`

```
\thanks{text}
```

The `\thanks` command produces a footnote to the title.

1.128 LaTeX.guide/\`title`

`\title`

```
\title{text}
```

The `\title` command declares `text` to be the title. Use `\` to tell LaTeX where to start a new line in a long title.

1.129 LaTeX.guide/\`pagenumbering`

`\pagenumbering`

```
\pagenumbering{num_style}
```

Specifies the style of page numbers. Possible values of `num_style` are:

- * `arabic` - Arabic numerals

- * roman - Lowercase roman numerals
- * Roman - Uppercase roman numerals
- * alph - Lowercase letters
- * Alph - Uppercase letters

1.130 LaTeX.guide\pagestyle

\pagestyle

\pagestyle{option}

The \pagestyle command changes the style from the current page on throughout the remainder of your document.

The valid options are:

- * plain - Just a plain page number.
- * empty - Produces empty heads and feet - no page numbers.
- * headings - Puts running headings on each page. The document style specifies what goes in the headings.
- * myheadings - You specify what is to go in the heading with the \markboth or the \markright commands.

\mark

1.131 LaTeX.guide\mark

\mark

\markboth{left head}{right head}
 \markright{right head}

The \markboth and \markright commands are used in conjunction with the page style myheadings for setting either both or just the right heading. In addition to their use with the myheadings page style, you can use them to override the normal headings in the headings style, since LaTeX uses these same commands to generate those heads. You should note that a "left-hand heading" is generated by the last \markboth command before the end of the page, while a "right-hand

heading" is generated by the first `\markboth` or `\markright` that comes on the page if there is one, otherwise by the last one before the page.

1.132 LaTeX.guide/\thispagestyle

`\thispagestyle`

`\thispagestyle{option}`

The `\thispagestyle` command works in the same manner as the `\pagestyle` command except that it changes the style for the current page only.

1.133 LaTeX.guide/Sectioning

Sectioning
=====

Sectioning commands provide the means to structure your text into units.

- * `\part`
- * `\chapter` (report style only)
- * `\section`
- * `\subsection`
- * `\subsubsection`
- * `\paragraph`
- * `\subparagraph`
- * `\subsubparagraph` (milstd and book-form styles only)
- * `\subsubsubparagraph` (milstd and book-form styles only)

All sectioning commands take the same general form, i.e.,

`\chapter[optional]{title}`

In addition to providing the heading in the text, the mandatory argument of the sectioning command can appear in two other places:

1. The table of contents
2. The running head at the top of the page

You may not want the same thing to appear in these other two places as appears in the text heading. To handle this situation, the sectioning commands have an optional argument that provides the text for these other two purposes.

The "sectioning commands" have *-forms that print a title, but do not include a number and do not make an entry in the table of contents. For example, the *-form of the `\subsection` command could look like:

```
\subsection*{Example subsection}
```

```
\appendix
```

1.134 LaTeX.guide/\appendix

```
\appendix
```

```
-----
```

```
\appendix
```

The `\appendix` command changes the way sectional units are numbered. The `\appendix` command generates no text and does not affect the numbering or parts.

1.135 LaTeX.guide/Spaces & Boxes

```
Spaces & Boxes
```

```
=====
```

```
\addvspace
```

```
\bigskip
```

```
\dotfill
```

```
\fbox
```

```
\framebox
```

```
\hfill
```

```
\hrulefill
```

```
\hspace
```

`\makebox`
`\mbox`
`\medskip`
`\newsavebox`
`\parbox`
`\raisebox`
`\rule`
`\savebox`
`\smallskip`
`\usebox`
`\vfill`
`\vspace`

1.136 LaTeX.guide/\addvspace

`\addvspace`

`\addvspace{length}`

The `\addvspace` command normally adds a vertical space of height `length`. However, if vertical space has already been added to the same point in the output by a previous `\addvspace` command, then this command will not add more space than needed to make the natural length of the total vertical space equal to `length`.

1.137 LaTeX.guide/\bigskip

`\bigskip`

The `\bigskip` command is equivalent to `\vspace{bigskipamount}` where `bigskipamount` is determined by the document style.

1.138 LaTeX.guide/\dotfill

```
\dotfill  
-----
```

The `\dotfill` command produces a "rubber length" that produces dots instead of just spaces.

1.139 LaTeX.guide/\fbox

```
\fbox  
-----
```

```
\fbox{text}
```

The `\fbox` command is exactly the same as the `\mbox` command, except that it puts a frame around the outside of the box that it creates.

1.140 LaTeX.guide/\framebox

```
\framebox  
-----
```

```
\framebox[width][position]{text}
```

The `\framebox` command is exactly the same as the `\makebox` command, except that it puts a frame around the outside of the box that it creates.

The `framebox` command produces a rule of thickness `\fboxrule`, and leaves a space `\fboxsep` between the rule and the contents of the box.

1.141 LaTeX.guide/\hfill

```
\hfill  
-----
```

The `\hfill` fill command produces a "rubber length" which can stretch or shrink horizontally. It will be filled with spaces.

1.142 LaTeX.guide/\hrulefill

`\hrulespace`

The `\hrulefill` fill command produces a "rubber length" which can stretch or shrink horizontally. It will be filled with a horizontal rule.

1.143 LaTeX.guide/\hspace

`\hspace`

`\hspace[*]{length}`

The `\hspace` command adds horizontal space. The length of the space can be expressed in any terms that LaTeX understands, i.e., points, inches, etc. You can add negative as well as positive space with an `\hspace` command. Adding negative space is like backspacing.

LaTeX removes horizontal space that comes at the end of a line. If you don't want LaTeX to remove this space, include the optional `*` argument. Then the space is never removed.

1.144 LaTeX.guide/\makebox

`\makebox`

`\makebox[width][position]{text}`

The `\makebox` command creates a box just wide enough to contain the text specified. The width of the box is specified by the optional width argument. The position of the text within the box is determined by the optional position argument.

* `c` - centered (default)

* `l` - flushleft

* `r` - flushright

See

`\makebox (picture)`

.

1.145 LaTeX.guide\mbox

\mbox

```
\mbox{text}
```

The `\mbox` command creates a box just wide enough to hold the text created by its argument.

1.146 LaTeX.guide\medskip

\medskip

The `\medskip` command is equivalent to `\vspace{medskipamount}` where `medskipamount` is determined by the document style.

1.147 LaTeX.guide\newsavebox

\newsavebox

```
\newsavebox{cmd}
```

Declares `cmd`, which must be a command name that is not already defined, to be a bin for saving boxes.

1.148 LaTeX.guide\parbox

\parbox

```
\parbox[position]{width}{text}
```

A `parbox` is a box whose contents are created in paragraph mode. The `\parbox` has two mandatory arguments:

- * `width` - specifies the width of the `parbox`, and
- * `text` - the text that goes inside the `parbox`.

LaTeX will position a `parbox` so its center lines up with the center of the text line. An optional first argument, `position`, allows you to line up either the top or bottom line in the `parbox`.

A `\parbox` command is used for a parbox containing a small piece of text, with nothing fancy inside. In particular, you shouldn't use any of the paragraph-making environments inside a `\parbox` argument. For larger pieces of text, including ones containing a paragraph-making environment, you should use a `minipage` environment.

1.149 LaTeX.guide/\raisebox

`\raisebox`

```
\raisebox{distance}[extend-above][extend-below]{text}
```

The `\raisebox` command is used to raise or lower text. The first mandatory argument specifies how high the text is to be raised (or lowered if it is a negative amount). The text itself is processed in LR mode.

Sometimes it's useful to make LaTeX think something has a different size than it really does - or a different size than LaTeX would normally think it has. The `\raisebox` command lets you tell LaTeX how tall it is.

The first optional argument, `extend-above`, makes LaTeX think that the text extends above the line by the amount specified. The second optional argument, `extend-below`, makes LaTeX think that the text extends below the line by the amount specified.

1.150 LaTeX.guide/\rule

`\rule`

```
\rule[raise-height]{width}{thickness}
```

The `\rule` command is used to produce horizontal lines. The arguments are defined as follows:

- * `raise-height` - specifies how high to raise the rule (optional)
- * `width` - specifies the length of the rule (mandatory)
- * `thickness` - specifies the thickness of the rule (mandatory)

1.151 LaTeX.guide/\savebox

`\savebox`

```
\sbox{cmd}[text]
\savebox{cmd}[width][pos]{text}
```

These commands typeset text in a box just as for `\mbox` or `\makebox`. However, instead of printing the resulting box, they save it in bin `cmd`, which must have been declared with `\newsavebox`.

1.152 LaTeX.guide/\smallskip

`\smallskip`

```
\smallskip
```

The `\smallskip` command is equivalent to `\vspace{smallskipamount}` where `smallskipamount` is determined by the document style.

1.153 LaTeX.guide/\usebox

`\usebox`

```
\usebox{cmd}
```

Prints the box most recently saved in bin `cmd` by a `\savebox` command.

1.154 LaTeX.guide/\vfill

`\vfill`

The `\vfill` fill command produces a rubber length which can stretch or shrink vertically.

1.155 LaTeX.guide/\vspace

`\vspace`

```
\vspace[*]{length}
```

The `\vspace` command adds vertical space. The length of the space can be expressed in any terms that LaTeX understands, i.e., points, inches, etc. You can add negative as well as positive space with an `\vspace` command.

LaTeX removes vertical space that comes at the end of a page. If you don't want LaTeX to remove this space, include the optional `*` argument. Then the space is never removed.

1.156 LaTeX.guide/Special Characters

Special Characters

=====

The following characters play a special role in LaTeX and are called "special printing characters", or simply "special characters".

\$ % & ~ _ ^ \ { }

Whenever you put one of these special characters into your file, you are doing something special. If you simply want the character to be printed just as any other letter, include a `\` in front of the character. For example, `\$` will produce \$ in your output.

The exception to the rule is the `\` itself because `\` has its own special meaning. A `\` is produced by typing `\backslash` in your file.

1.157 LaTeX.guide/Splitting the Input

Splitting the Input

=====

A large document requires a lot of input. Rather than putting the whole input in a single large file, it's more efficient to split it into several smaller ones. Regardless of how many separate files you use, there is one that is the root file; it is the one whose name you type when you run LaTeX.

`\include`

`\includeonly`

`\input`

1.158 LaTeX.guide/\include

`\include`

`\include{file}`

The `\include` command is used in conjunction with the `\includeonly` command for selective inclusion of files. The file argument is the first name of a file, denoting `file.tex`. If file is one the file names in the file list of the `\includeonly` command or if there is no `\includeonly` command, the `\include` command is equivalent to

`\clearpage \input{file} \clearpage`

except that if the file `file.tex` does not exist, then a warning message rather than an error is produced. If the file is not in the file list, the `\include` command is equivalent to `\clearpage`.

The `\include` command may not appear in the preamble or in a file read by another `\include` command.

1.159 LaTeX.guide/\includeonly

`\includeonly`

`\includeonly{file_list}`

The `\includeonly` command controls which files will be read in by an `\include` command. It can only appear in the preamble.

1.160 LaTeX.guide/\input

`\input`

`\input{file}`

The `\input` command causes the indicated file to be read and processed, exactly as if its contents had been inserted in the current file at that point. The file name may be a complete file name with extension or just a first name, in which case the file `file.tex` is used.

1.161 LaTeX.guide/Starting & Ending

Starting & Ending
=====

Your input file must contain the following commands as a minimum...

```

\documentstyle{style}
\begin{document}
  ... your text goes here ...
\end{document}
...where the style selected is one the valid styles for LaTeX. See

```

Document Styles
 , and also see
 Overview
 , for details of the various
 document styles available locally.

You may include other LaTeX commands between the `\documentstyle` and the `\begin{document}` commands.

1.162 LaTeX.guide/Table of Contents

Table of Contents

=====

A table of contents is produced with the `\tableofcontents` command. You put the command right where you want the table of contents to go; LaTeX does the rest for you. It produces a heading, but it does not automatically start a new page. If you want a new page after the table of contents, include a `\newpage` command after the `\tableofcontents` command.

There are similar commands `\listoffigures` and `\listoftables` for producing a list of figures and a list of tables, respectively. Everything works exactly the same as for the table of contents.

NOTE: If you want any of these items to be generated, you cannot have the `\nofiles` command in your document.

`\addcontentsline`

`\addtocontents`

1.163 LaTeX.guide/\addcontentsline

`\addcontentsline`

```
\addcontentsline{file}{sec_unit}{entry}
```

The `\addcontentsline` command adds an entry to the specified list or

table where...

- * `file` is the extension of the file on which information is to be written: `toc` (table of contents), `lof` (list of figures), or `lot` (list of tables).
- * `sec_unit` controls the formatting of the entry. It should be one of the following, depending upon the value of the `file` argument:
 1. `toc` - the name of the sectional unit, such as part or subsection.
 2. `lof` - figure
 3. `lot` - table
- * `entry` is the text of the entry.

1.164 LaTeX.guide/addtocontents

`\addtocontents`

```
\addtocontents{file}{text}
```

The `\addtocontents` command adds text (or formatting commands) directly to the file that generates the table of contents or list of figures or tables.

- * `file` is the extension of the file on which information is to be written: `toc` (table of contents), `lof` (list of figures), or `lot` (list of tables).
- * `text` is the information to be written.

1.165 LaTeX.guide/Terminal Input-Output

Terminal Input/Output

=====

```
\typeout
```

```
\typein
```

1.166 LaTeX.guide/typeout

`\typeout`

`\typeout{msg}`

Prints `msg` on the terminal and in the log file. Commands in `msg` that are defined with `\newcommand` or `\renewcommand` are replaced by their definitions before being printed.

LaTeX's usual rules for treating multiple spaces as a single space and ignoring spaces after a command name apply to `msg`. A `\space` command in `msg` causes a single space to be printed.

1.167 LaTeX.guide/\typein

`\typein`

`\typein[cmd]{msg}`

Prints `msg` on the terminal and causes LaTeX to stop and wait for you to type a line of input, ending with return. If the `cmd` argument is missing, the typed input is processed as if it had been included in the input file in place of the `\typein` command. If the `cmd` argument is present, it must be a command name. This command name is then defined or redefined to be the typed input.

1.168 LaTeX.guide/Typefaces

Typefaces

=====

The typeface is specified by giving the "size" and "style". A typeface is also called a "font".

Styles

Sizes

1.169 LaTeX.guide/Styles

`\Styles`

The following type style commands are supported by LaTeX.

```
\rm      Roman.
\it      Italics.
\em      Emphasis (toggles between \it and \rm).
\bf      Boldface.
\sl      Slanted.
\sff     Sans serif.
\sc      Small caps.
\tt      Typewriter.
```

1.170 LaTeX.guide/Sizes

Sizes

The following type size commands are supported by LaTeX.

```
\tiny
\scriptsize
\footnotesize
\small
\normalsize
    (default)
\large
\Large
    (capital "l")
\LARGE
    (all caps)
\huge
\Huge
    (capital "h")
```

1.171 LaTeX.guide/Parameters

Parameters

The input file specification indicates the file to be formatted; TeX uses .tex as a default file extension. If you omit the input file entirely, TeX accepts input from the terminal. You specify command options by supplying a string as a parameter to the command. eg.

```
latex "\scrollmode\input foo.tex"
```

...will process foo.tex without pausing after every error.

Output files are always created in the current directory. When you fail to specify an input file name, TeX bases the output names on the file specification associated with the logical name TEX_OUTPUT, typically texput.log.

1.172 LaTeX.guide/List of Commands

Alphabetical List of Commands

=====

$_exp$ (subscript)

$\^exp$ (superscript)

\backslash

\hyphenation

$\;$

\colon

\comma

$\!$

$\=$

$\>$

$\<$

$\+$

\tabbing

\A

$\`$

`\addcontentsline`

`\addtocontents`

`\addtocounter`

`\address`

`\addtolength`

`\addvspace`

`\alph`

`\appendix`

`\arabic`

`array`

`\author`

`\bf`

`\bibitem`

`\bigskip`

`\cdots`

`center`

`\centering`

`\circle`

`\cite`

`\cleardoublepage`

`\clearpage`

`\cline`

`\closing`

`\dashbox`

`\date`

`\ddots`

`description`

`\dotfill`

`\em`
`enumerate`
`eqnarray`
`equation`
`figure`
`\fbox`
`\flushbottom`
`flushleft`
`flushright`
`\fnsymbol`
`\footnote`
`\footnotemark`
`\footnotesize`
`\footnotetext`
`\frac`
`\frame`
`\framebox`
`\hfill`
`\hline`
`\hrulefill`
`\hspace`
`\huge`
`\Huge` (capital 'h')
`\hyphenation`
`\include`
`\includeonly`
`\indent`
`\input`
`\it`

itemize
\kill
\label
\large
\Large (capital 'l')
\LARGE (all caps)
\ldots
\line
\linebreak
\linethickness
list
\location
\makebox
\maketitle
\mark
\mbox
\medskip
minipage
\multicolumn
\multirow
\newcommand
\newcounter
\newenvironment
\newfont
\newlength
\newline
\newpage
\newsavebox

`\newtheorem`
`\nocite`
`\noindent`
`\nolinebreak`
`\normalsize`
`\nopagebreak`
`\onecolumn`
`\opening`
`\oval`
`\overbrace`
`\overline`
`\pagebreak`
`\pagenumbering`
`\pageref`
`\pagestyle`
`\par`
`\parbox`
`picture`
`\put`
`quotation`
`quote`
`\raggedbottom`
`\raggedleft`
`\raggedright`
`\raisebox`
`\ref`
`\rm`
`\roman`
`\rule`

`\savebox`
`\sc`
`\scriptsize`
`\setcounter`
`\setlength`
`\settowidth`
`\sf`
`\shortstack`
`\signature`
`\sl`
`\small`
`\smallskip`
`\sqrt`
`tabbing`
`table`
`tabular`
`\telephone`
`\thanks`
`thebibliography`
`theorem`
`\thispagestyle`
`\tiny`
`\title`
`titlepage`
`\tt`
`\twocolumn`
`\typeout`
`\typein`

`\underbrace`
`\underline`
`\usebox`
`\usecounter`
`\value`
`\vdots`
`\vector`
`\verb`
`verbatim`
`verse`
`\vfill`
`\vline`
`\vspace`

1.173 LaTeX.guide/_{exp} (subscript)

`_ {exp}` (subscript)

To get an expression `exp` to appear as a subscript, you just type `_ {exp}`. Use in math mode.

See

Math Formulae

.

See

Subscripts & Superscripts

.

1.174 LaTeX.guide/^{exp} (superscript)

`^ {exp}` (superscript)

To get an expression `exp` to appear as a superscript, you just type `^ {exp}`. Use in math mode.

See
Math Formulae
.

See
Subscripts & Superscripts
.

1.175 LaTeX.guide^{\;}

--
 $\;$

Include a thick space in math mode.

See
Math Formulae
.

See
Spacing in Math Mode
.

1.176 LaTeX.guide^{\:}

--
 $\:$

Include a medium space in math mode.

See
Math Formulae
.

See
Spacing in Math Mode
.

1.177 LaTeX.guide^{\,}

--
 $\,$

Include a thin space in math mode.

See

Math Formulae

.

See

Spacing in Math Mode

.

1.178 LaTeX.guide\!

`\!`

--

Include a negative thin space in math mode.

See

Math Formulae

.

See

Spacing in Math Mode

.

1.179 LaTeX.guide\bf

`\bf`

Boldface typeface.

See

Typefaces

.

See

Styles

.

1.180 LaTeX.guide\cdots

`\cdots`

The `\cdots` command produces a horizontal ellipsis where the dots are raised to the center of the line.

See

Math Formulae

.

See

Math Miscellany

.

1.181 LaTeX.guide/`\ddots`

`\ddots`

The `\ddots` command produces a diagonal ellipsis.

See

Math Formulae

.

See

Math Miscellany

.

1.182 LaTeX.guide/`\em`

`\em`

Emphasis (toggles between `\it` and `\rm`).

See

Typefaces

.

See

Styles

.

1.183 LaTeX.guide/\footnotesize

`\footnotesize`

Third smallest of 10 typefaces available. This is the default size for footnotes.

See

Typefaces

.

See

Sizes

.

1.184 LaTeX.guide/\frac

`\frac`

`\frac{num}{den}`

The `\frac` command produces the fraction num divided by den.

See

Math Formulae

.

See

Math Miscellany

.

1.185 LaTeX.guide/\huge

`\huge`

Second largest of 10 typefaces available.

See

Typefaces

.

See

Sizes

.

1.186 LaTeX.guide/\Huge (capital `h`)

\Huge

Largest of 10 typefaces available. All fonts may not be available in this size.

See

Typefaces

.

See

Sizes

.

1.187 LaTeX.guide/\it

\it

Italics typeface.

See

Typefaces

.

See

Styles

.

1.188 LaTeX.guide/\large

\large

Slightly larger than default typeface size.

See

Typefaces

.

See

Sizes

.

1.189 LaTeX.guide/Large (capital `l`)

`\Large`

Fourth largest of typefaces available. Is generally the default for titles.

See

Typefaces

.

See

Sizes

.

1.190 LaTeX.guide/LARGE (all caps)

`\LARGE`

Third largest of typefaces available.

See

Typefaces

.

See

Sizes

.

1.191 LaTeX.guide/ldots

`\ldots`

The `\ldots` command produces an ellipsis. This command works in any mode, not just math mode.

See

Math Formulae

.

See
Math Miscellany
.

1.192 LaTeX.guide/\normalsize

`\normalsize`

The size of `\normalsize` is defined by as 10pt unless the 11pt or 12pt document style option is used.

See
Typefaces
.

See
Sizes
.

1.193 LaTeX.guide/\overbrace

`\overbrace`

`\overbrace{text}`

The `\overbrace` command generates a brace over text.

See
Math Formulae
.

See
Math Miscellany
.

1.194 LaTeX.guide/\overline

`\overline`

`\overline{text}`

The `\overline` command causes the argument text to be overlined.

See

Math Formulae

.

See

Math Miscellany

.

1.195 LaTeX.guide/\rm

`\rm`

Roman typeface (default).

See

Typefaces

.

See

Styles

.

1.196 LaTeX.guide/\sc

`\sc`

Small caps typeface.

See

Typefaces

.

See

Styles

.

1.197 LaTeX.guide/\scriptsize

`\scriptsize`

Second smallest of 10 typefaces available.

See

Typefaces

.

See

Sizes

.

1.198 LaTeX.guide/\sf

`\sf`

Sans serif typeface.

See

Typefaces

.

See

Styles

.

1.199 LaTeX.guide/\sl

`\sl`

Slanted typeface.

See

Typefaces

.

See

Styles

.

1.200 LaTeX.guide\small

`\small`

Slightly smaller than default typeface size.

See

Typefaces

.

See

Sizes

.

1.201 LaTeX.guide\sqrt

`\sqrt`

`\sqrt[root]{arg}`

The `\sqrt` command produces the square root of its argument. The optional argument, *root*, determines what root to produce, i.e. the cube root of $x+y$ would be typed as $\sqrt[3]{x+y}$.

See

Math Formulae

.

See

Math Miscellany

.

1.202 LaTeX.guide\tiny

`\tiny`

Smallest of 10 typefaces available. All fonts may not be available in this size.

See

Typefaces

.

See

Sizes

.

1.203 LaTeX.guide/\tt

\tt

Typewriter typeface.

See

Typefaces

.

See

Styles

.

1.204 LaTeX.guide/\underbrace

\underbrace

\underbrace{text}

The \underbrace command generates text with a brace underneath.

See

Math Formulae

.

See

Math Miscellany

.

1.205 LaTeX.guide/\underline

\underline

\underline{text}

The \underline command causes the argument text to be underlined. This command can also be used in paragraph and LR modes.

See
 Math Formulae
 .

See
 Math Miscellany
 .

1.206 LaTeX.guide/\vdots

 \vdots

The `\vdots` command produces a vertical ellipsis.

See
 Math Formulae
 .

See
 Math Miscellany
 .

1.207 LaTeX.guide/Index

 Index

Appendix, creating
 \appendix

Arrays
 array

article style
 Document Styles

Author, for titlepage
 \author

Bibliography, creating
 thebibliography

book style
 Document Styles

Centering text

	<code>\centering</code>
Centering text	<code>center</code>
chapter, Sectioning	Sectioning
Characters, special	Special Characters
Commands, defining new ones	<code>\newcommand</code>
Counters, a list of	Counters
Counters, adding to	<code>\addtocounter</code>
Counters, creating	<code>\newcounter</code>
Counters, setting	<code>\setcounter</code>
Counters, use of	<code>\value</code>
Counters, use of	<code>\usecounter</code>
Creating Letters	Letters
Creating pictures	<code>picture</code>
Cross referencing	<code>\pageref</code>
Cross referencing	<code>\ref</code>
Cross referencing	Cross References
Cross referencing	<code>\label</code>
Date, for titlepage	<code>\date</code>
Defining a new command	<code>\newcommand</code>
Defining new environments	

	<code>\newenvironment</code>
Defining new fonts	<code>\newfont</code>
Defining new theorems	<code>\newtheorem</code>
description Environments	<code>description</code>
Document Styles	Document Styles
Ending & Starting	Starting & Ending
enumerate Environment	<code>enumerate</code>
Environment	<code>verbatim</code>
Environment	<code>tabular</code>
Environments	Environments
Environments	<code>description</code>
Environments	<code>eqnarray</code>
Environments	<code>enumerate</code>
Environments	<code>tabbing</code>
Environments	<code>itemize</code>
Environments	<code>titlepage</code>
Environments	<code>list</code>
Environments	<code>equation</code>
Environments	<code>minipage</code>
Environments	

	figure
Environments	theorem
Environments	quote
Environments	flushleft
Environments	\raggedright
Environments	table
Environments	quotation
Environments	thebibliography
Environments	picture
Environments	verse
Environments, defining	\newenvironment
eqnarray Environment	eqnarray
equation Environment	equation
figure Environment	figure
Flushing a page	\clearpage
Flushing a page	\cleardoublepage
Font Sizes	Sizes
Font Styles	Styles
Fonts	Typefaces
Fonts, new commands for	

	<code>\newfont</code>
Footnotes, creating	Footnotes
Formatting Text	<code>\centering</code>
Formatting text	<code>center</code>
Formulae, maths	Math Formulae
Hyphenation	<code>\hyphenation</code>
Hyphenation	<code>\- (hyphenation)</code>
Input file, splitting	Splitting the Input
Input/Output	Terminal Input-Output
Inserting figures	<code>figure</code>
itemize Environment	<code>itemize</code>
Left-to-right mode	Modes
Lengths, defining and using	<code>\newlength</code>
Lengths, defining and using	<code>\settowidth</code>
Lengths, defining and using	Lengths
Lengths, defining and using	<code>\addtolength</code>
Lengths, defining and using	<code>\setlength</code>
letter style	Document Styles
Letters	<code>\signature</code>
Letters	

	Declarations
Letters	<code>\opening</code>
Letters	<code>\closing</code>
Letters	<code>\location</code>
Letters	<code>\telephone</code>
Letters	<code>\address</code>
Letters	Letters
Line Breaking	Line & Page Breaking
Line Breaks	<code>\linebreak</code>
list Environment	<code>list</code>
Lists of items	<code>list</code>
Lists of items	<code>itemize</code>
Lists of items	<code>enumerate</code>
LR mode	Modes
Making a title page	<code>titlepage</code>
Making Paragraphs	Making Paragraphs
Math Formulae	Math Formulae
Math mode	Math Symbols
Math mode	Math Miscellany
Math mode	

	Math Formulae
Math mode	Modes
Math mode	Subscripts & Superscripts
Math mode, spacing	Spacing in Math Mode
Maths Miscellany	Math Miscellany
Maths symbols	Math Symbols
minipage Environment	minipage
Modes	Modes
Multicolumn text	\twocolumn
New Page	\newpage
Newline	\newline
Newline	\
Page Breaking	Line & Page Breaking
Page breaking	\pagebreak
Page Formatting	Line & Page Breaking
Page Styles	\pagestyle
Page Styles	\pagenumbering
Page styles	Page Styles
Page Styles, changing	\pagestyle
Pagenumbering	

	<code>\pagenumbering</code>
Paragraph mode	Modes
paragraph, Sectioning	Sectioning
Paragraphs	<code>\par</code>
Paragraphs	<code>\noindent</code>
Paragraphs	<code>\indent</code>
Paragraphs	Making Paragraphs
picture Environment	<code>picture</code>
Poetry, an environment for verse	
quotation Environment	<code>quotation</code>
report style	Document Styles
Sectioning	Sectioning
Sizes of text	Sizes
Spaces & Boxes	<code>\hspace</code>
Spaces & Boxes	<code>\parbox</code>
Spaces & Boxes	<code>\fbox</code>
Spaces & Boxes	<code>\dotfill</code>
Spaces & Boxes	<code>\bigskip</code>
Spaces & Boxes	<code>\vspace</code>
Spaces & Boxes	

	<code>\vfill</code>
Spaces & Boxes	<code>\usebox</code>
Spaces & Boxes	<code>\smallskip</code>
Spaces & Boxes	<code>\savebox</code>
Spaces & Boxes	<code>\rule</code>
Spaces & Boxes	<code>\raisebox</code>
Spaces & Boxes	<code>\newsavebox</code>
Spaces & Boxes	<code>\makebox</code>
Spaces & Boxes	<code>\hrulefill</code>
Spaces & Boxes	<code>\addvspace</code>
Spaces & Boxes	<code>\medskip</code>
Spaces & Boxes	<code>\mbox</code>
Spaces & Boxes	Spaces & Boxes
Spaces & Boxes	<code>\framebox</code>
Spaces & Boxes	<code>\hfill</code>
Spacing, within Math mode	Spacing in Math Mode
Special Characters	Special Characters
Splitting the input file	Splitting the Input
Starting & Ending	Starting & Ending
Styles of document	

Document Styles

Styles of text

Styles

Styles, pages

Page Styles

subparagraph, Sectioning

Sectioning

Subscript

Subscripts & Superscripts

subsection, Sectioning

Sectioning

subsubparagraph, Sectioning

Sectioning

subsubsection, Sectioning

Sectioning

subsubsubparagraph, Sectioning

Sectioning

Superscript

Subscripts & Superscripts

Symbols

Math Symbols

tabbing Environment

tabbing

table Environment

table

Table of Contents, creating

Table of Contents

tabular Environment

tabular

Terminal Input/Output

Terminal Input-Output

Thanks, for titlepage

\thanks

thebibliography Environment

thebibliography

theorem Environment

theorem

Theorems, defining

`\newtheorem`

Title making `\maketitle`

Title pages, creating `titlepage`

Title, for titlepage `\title`

titlepage Environment `titlepage`

Typeface Sizes `Sizes`

Typeface Styles `Styles`

Typefaces `Typefaces`

Using Counters `\value`

Using Counters `\usecounter`

Variables, a list of... (see also Counters)
Counters

verbatim Environment `\verb`

verbatim Environment `verbatim`

verse Environment `verse`

`\maketitle` Style `\maketitle`

`\quote` Environment `quote`
