LaTeX

Mingming Li

First Created: September 24, 2020
Last Modified: March 2, 2023

Contents

List of Figures

List of Tables

Chapter 1

LaTeX Base

1.1 What is LaTeX

LaTeX is a document markup language[1]. It separate format from content.

1.2 Reason to Use It

Because LaTeX is a markup language, you should learn it before you can use it. So why should you spend so much time to learn it while there is so much document creator like Word, Pages?

There is several reasons that push me to select it.

- ♠ It provides powerful edit ability. You can almost get whatever you want to show, especially the mathematical equations.
- ♠ Because it separate the format and the content, it is easy to do format alteration in the full document domain.
- ♠ Once you create your own template, it is easy to wreate document with the template applied, saving so much time in format.

1.3 Logical formatting

```
1  \documentclass{article}
2  \begin{document}
3  \title{Example}
4  \author{Mingming Li}
5  \date{2022/11/04}
6  \maketitle{}
7  \section{Logical Formatting}
8  This example show logical formatting.
9  \end{document}
```

We do not specify the font family, font size, and so on, instead we tell LaTeX the **class**, the **author**, or the **section** and let LaTeX to format it.

1.4 Command

LaTeX commands begin with a backslash, followed by big or small letters. LaTeX commands are usually named with small letters and in a descriptive way. There are exceptions: a backslash and just one special character. Commands may have arguments, given in curly braces or in square brackets.

Calling a command looks like the following:

[1] Just like HTML

```
\command
\command{argument}
\command[optional argument]{argument}
```

For example

```
{\large Title}
\include{preamble}
\documentclass[12pt]{article}
```

1.5 Comment

The percent sing(%) introduces a comment.

```
\include{preamble}  % include preamble tex file
```

1.6 Environment

```
% This is the environment syntax.
\begin{name}[optional argument]{argument}
  ...
\end{name}
```

1.7 Breaks and Empty Lines

LaTeX treats multiple spaces just like a single space. Also, a single line break has the same effect like a single space. It doesn't matter how you arrange your text in the editor using spaces or breaks, the output will stay the same. A blank line denotes a paragraph break. Like spaces, multiple empty lines are treated as one. Briefly said, spaces separate words, empty lines separate paragraphs.

1.8 Special Symbols

By putting a backslash before such a special symbol, we turned it into a LaTeX command. This command has the only purpose of printing out that symbol.

```
\%  % just print % symbol
\textbackslash % just print \ symbol
```

1.9 Create Your Own Commands

```
% This is the full definition of creating you own command.
\newcommand{command}[arguments][optional]{definition}
```

```
% With no arguments.
\newcommand{\TUG}{TeX Users Group\xspace}
\TUG

% With arguments.
\newcommand{\keyword}[1]{\textbf{#1}}
\keyword{declrations}
```

```
% With optional arguments.
\newcommand{\keyword2}[2][\bfseries]{{#1#2}}
\keyword2[\itshape]{declarations}
```

1.10 Get Help

Three ways to get help about the package:

1. `texdoc <package>`
2. `kpsewhich <package>.sty`
3. Visit the website: http://ctan.org/pkg

1.11 Install Extra Packages

The easy way is to use the terminal to install extra packages:

```
# Tex Live manager
tlmgr install <package>
```

1.12 Floats

LaTeX provides two floating environments, namely, `figure` and `table`. They are briefly called **floats**. Their content may float to a place where it's the optimum for the page layout.

Here's the float placement options:

- h: here. The float may appear where it's been written in the source code.
- t: top. Placing at the top of a page is permitted.
- b: bottom. The float may appear at the bottom of a page.
- p: page. The float is allowed to appear on a separate page, where only floats may reside but no normal text.
- !: tells LaTeX to try harder! Some constraints may be ignored, easing the placement.

The most flexible is using the placement `[!htbp]`, allowing a float everywhere.

1.13 Modes

LaTeX knows three general **modes**:

- The **paragraph mode**: The text is typeset as a sequence of words in lines, paragraphs, and pages.
- The **left-to-right mode**: The text is also considered to be a sequence of words, but LaTeX typesets it from left to right without breaking the line.
- The **math mode**: Letters are treated as math symbols. A lot of symbols can be used, most of them exclusively in this mode. Such symbols are roots, sum signs, relation signs, math accents, arrows, and various delimiters like brackets and braces. Space characters between letters and symbols are ignored.

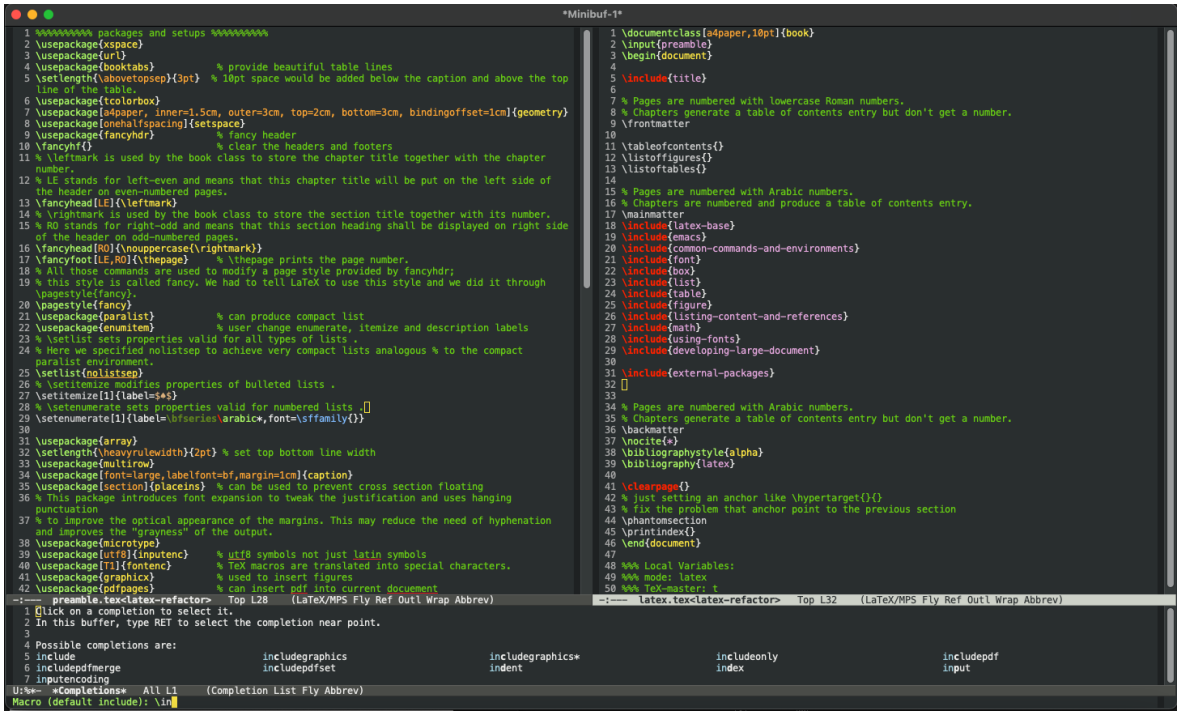
Chapter 2

Emacs

Type all the TEX commands is time consuming and error prone. **Emacs** comes to the rescue. It can auto-fill the commands.

Emacs comes with a package for editing TeX and LaTeX files. However, this package is extremely limited in its functionality. A far better package called **AUC TeX** can help you write your papers efficiently.

Here is the Figure of using Emacs:



Figures 2.1: Emacs

Chapter 3

Common Commands and Environments

3.1 Commands

```
% produces some space.
\quad

% ended a line.
\\ or \newline

% prevents a line break at the current position.
\nolinebreak

%  `` and '' is the quotation in latex.
``hello''

% ragged left
{\raggedleft Example text}
% ragged right
{\raggedright Example text}
% centering
{\centering Example text}


% tells LaTeX to produce a file with the extension .toc.
% This file will be used to generate a table of contents.
% We had to typeset twice: in the first run, the .toc file
% was written and in the second run, LaTeX read it and processed it.
\tableofcontents{}


% causes a page break. Furthermore, the text has been
% stretched to fill the page down to the bottom.
\pagebreak{}
% breaks the page as well, but it doesn't stretch the
% text: the remaining space of the page will stay empty.
\newpage{}
% forbids page breaking
\nopagebreak{}


% to squeeze more text onto a page.
\enlargethispage{2\baselineskip}
```

```latex


% placed a superscripted number at the current position.
% Further, it prints its argument text into the bottom of
% the page, marked by the same number
\footnote{text}
% modify the line that separates footnotes from the text
% is produced by the command \footnoterule.
\renewcommand{\footnoterule}{\noindent\smash{\rule[3pt]{\textwidth}{0.4pt}}}

\rule[raising]{width}{height}
% draw a line 1pt at thick, as wide as text, raised a bit by 3 pt
\rule[3pt]{\textwidth}{1pt}

% \smash, we let our line pretend to have a height and a depth of
zero, so it's occupying no vertical space at all


% when we use \footnote inside an argument, there will be an error.
% \protect simply prevents this processing error.
\protect{}
\section{Section \protect{\footnote{text}}}
% to avoid footnote appearing in heading and table of content
\section[title without footnote]{Section \protect{\footnote{text}}}



% ends the current page and causes all already defined figures and
tables to be printed out.
\clearpage{}

\cleardoublepage{}


% To be able to refer to a certain point, we have to mark it by a label.
% We can reference to the name of that label afterwards.
% notice , typeset twice to produce the cross reference
\label % marks the position
\ref % prints the number of the element we refer to
\pageref % prints the page number of that element
```

3.2 Environments

```latex
% quote long text
\begin{quotation}
\end{quotation}
```

```latex
% center environment
\begin{center}
```

```
\end{center}
```

Chapter 4

Font

4.1 Shape

Tables 4.1: Font Command

Command	Explaination	Output
`\textbf{Example}`	bold font	**Example**
`\textit{Example}`	italic	*Example*
`\textsl{Example}`	slated	*Example*
`\textsc{Example}`	small caps	EXAMPLE
`\textup{Example}`		Example
`\textmd{Example}`	medium	Example
`\textnormal{Example}`		Example
`\textsf{Example}`	sans-serif	Example
`\texttt{Example}`	typewritter	`Example`
`\textrm{Example}`	Roman	Example

Tables 4.2: Font Declaration

Declaration	Explaination	Output
`{\itshape Example}`	italic	*Example*
`{\bfseries Example}`	bold font	**Example**
`{\slshape Example}`	slated	*Example*
`{\scshape Example}`	small caps	EXAMPLE
`{\upshape Example}`		Example
`{\mdseries Example}`	medium	Example
`{\normalfont Example}`		Example
`{\sffamily Example}`	sans-serif	Example
`{\ttfamily Example}`	typewritter	`Example`
`{\rmfamily Example}`	roman	Example

Tables 4.3: Font Emphasized

Command	Explaination	Output
`\emph{Example}`	emphasized	*Example*

4.2 Size

Tables 4.4: Font Size

Command	Output
`\tiny{Example}`	Example
`\scriptsize{Example}`	Example
`\footnotesize{Example}`	Example
`\small{Example}`	Example
`\normalsize{Example}`	Example
`\large{Example}`	Example
`\Large{Example}`	Example
`\LARGE{Example}`	Example
`\huge{Example}`	Example
`\Huge{Example}`	Example

Chapter 5

Box

5.1 parbox

We used the command `\parbox` to create a column.

```
\parbox[alignment]{width}{text}
\parbox[alignment][height][inner alignment]{width}{text}
```

alignment Optional argument for the vertical alignment. State t to align at the top line of the box; write b to align at its bottom line. The default behavior is to place the box such that its center is in line with the center of the current text line.

height If this optional argument isn't given, the box will have just the natural height of the text inside. Use this argument if you want to change the height of the box to make it bigger or smaller.

inner alignment Especially, if the height of the box is different to the natural height of the contained text, you might want to adjust the text position. The argument means:

- ♠ c: vertically center the text in the box
- ♠ t: place text at the top of the box
- ♠ b: place text at its bottom
- ♠ s: stretch the text vertically if possible

```
Text line
\quad\fbox{\parbox[b]{1.8cm}{this parbox is aligned at its bottom line}}
\quad\fbox{\parbox{1.5cm}{center-aligned parbox}}
\quad\fbox{\parbox[t]{2cm}{another parbox aligned at its top line}}
```

Text line | this parbox is aligned at its bottom line | center-aligned parbox | another parbox aligned at its top line

5.2 fbox

This command show the frame out.

```
\fbox{Example}
```

Example

5.3 minipage

Parboxes are suitable for boxes with only a little text inside. In case of a box containing a large amount of text, the closing brace could easily be forgotten or overlooked. The minipage environment

would then be a better choice.

```
\begin{minipage}{3cm}
  Hello World!
\end{minipage}
```

Hello World!

5.4 mbox

```
\mbox{Hello World}
```

Hello World

5.5 tcolorbox

```
\begin{tcolorbox}
  \mbox{Hello World}
\end{tcolorbox}
```

Hello World

Chapter 6

Lists

6.1 Bulleted Lists

```
\begin{itemize}
\item geometry
\item amsmath
\end{itemize}
```

- ♠ geometry
- ♠ amsmath

6.2 Numbered Lists

```
\begin{enumerate}
\item geometry
\item amsmath
\end{enumerate}
```

1. geometry
2. amsmath

6.3 Definition Lists

```
\begin{description}
\item[paralist] provides compact lists and list versions that
  can be used within paragraphs, helps to customize labels and
  layout
\item[enumitem] gives control over labels and lengths
  in all kind of lists
\item[mdwlist] is useful to customize description lists, it
  even allows multi-line labels. It features compact lists and
  the capability to suspend and resume.
\item[desclist] offers more flexibility in definition list
\item[multenum] produces vertical enumeration in multiple
  columns
\end{description}
```

paralist provides compact lists and list versions that can be used within paragraphs, helps to customize labels and layout

enumitem gives control over labels and lengths in all kind of lists

mdwlist is useful to customize description lists, it even allows multi-line labels. It features compact lists and the capability to suspend and resume.

desclist offers more flexibility in definition list

multenum produces vertical enumeration in multiple columns

Chapter 7

Tables

7.1 tabular

LaTeX provides the tabular environment for typesetting simple and complex tables which can be nested.

```
\newcommand{\head}[1]{\textnormal{\textbf{#1}}}
\begin{tabular}{ccc}
  \hline                        % Draw a horizontal line over whole width of
      the table
  \head{Command} & \head{Declaration} & \head{Output} \\
  \hline
  \verb|\textrm| & \verb|\rmfamily| & \rmfamily Example text \\
  \verb|\textsf| & \verb|\sffamily| & \sffamily Example text \\
  \verb|\texttt| & \verb|\ttfamily| & \ttfamily Example text \\
  \hline
\end{tabular}
```

Command	Declaration	Output
\textrm	\rmfamily	Example text
\textsf	\sffamily	Example text
\texttt	\ttfamily	Example text

Within tabular, three types of lines may be used:

- `\hline`: draws a horizontal line over the whole width of the table
- `\cline{m-n}`: draws a horizontal line starting at the beginning of column `m` and ending at the end of column `n`
- `vline`: draws a vertical line over the full height and depth of the current row

The options understood by the tabular environment are as follows:

- `l`: for left alignment.
- `c`: for centered alignment.
- `r`: for right alignment.
- `p{width}`: for a "paragraph" cell of a certain width. . If you place several `p` cells next to each other, they will be aligned at their top line. It's equivalent to using `\parbox[t]{width}` within a cell.
- `@{code}` inserts `code` instead of empty space before or after a column. This might also be some text or it could be left empty to avoid this space.
- `|`: stands for a vertical line.
- `*{n}{options}` is equivalent to `n` copies of options, where n is a positive integer and options may consist of one or more column specifiers including * as well.

7.2 Spanning Entries Over Multiple Columns

```
\begin{tabular}{@{}l*2{>{\textbackslash\ttfamily}l}l<{Example text}@{}}
  \toprule
  & \multicolumn{2}{c}{\head{Input}} & \multicolumn{1}{c}{\head{Output}}\\
  & \normal{\head{Command}} & \normal{\head{Declaration}} & \normal{}\\
  \cmidrule(lr){2-3}\cmidrule(l){4-4}
  Family & textrm&rmfamily & \rmfamily\\
  & textsf & sffamily & \sffamily\\
  & texttt & ttfamily & \ttfamily\\
  \bottomrule
\end{tabular}
```

	Input		Output
	Command	**Declaration**	
Family	\textrm	\rmfamily	Example text
	\textsf	\sffamily	Example text
	\texttt	\ttfamily	Example text

7.3 Adding Captions to Tables

```
\begin{table}
  \centering
  \begin{tabular}{}

  \end{tabular}
  \caption{Caption}
  \label{tab:caption}
\end{table}
```

Chapter 8

Figure

graphicx is need to insert figure into you document.

LaTeX supports the following file types:

- PNG, JPG, and PDF if you directly compile to PDF (**pdfLaTeX**)
- EPS if you compile to DVI and convert to PS and PDF (traditional LaTeX)

> - PS: PostScript
> - EPS: Encapsulated PostScript
> - DVI: Device Independent Format

You don't need to specify a filename extension, it will be automatically added. Don't use blanks in the filename or path! Blanks and special characters may cause problems with `\includegraphics`. If such symbols in filenames are required, load the package grffile to try to fix it.

```
\usepackage[demo]{graphicx}
\begin{figure}
  \centering
  \includegraphics{test}
  \caption{Test figure}
\end{figure}
```

Because we specified the demo option, graphicx doesn't require a file test.png or any other file; instead it's just printing a black filled rectangle. This is useful for testing or if you would like to discuss a LaTeX problem in an online forum, but don't wish to publish your pictures.

```
% syntax
\includegraphics[k=v]{file name}
```

Here are the most popular ones:

- width: `width=0.8\textwidth`
- height
- scale: `scale=0.5`
- angle: `angle=90`

Chapter 9

Listing Content and References

9.1 Table of Content

Command	Level
\part	-1 (book and report class)
\chapter	0 (not available in article)
\section	1
\subsection	2
\subsubsection	3
\paragraph	4
\subparagraph	5

Tables 9.1: Depth of the TOC

There's a variable representing the level, namely, `\tocdepth`. It's an integer variable which we call a **counter**.

There are two basic ways to adjust a counter value:

```
% specifies an integer value of 'n' for the counter 'name'.
\setcounter{name}{n}
% adds the integer value of 'n' to value of the counter 'name'. 'n' may be
    negative.
\addtocounter{name}{n}


\setcounter{tocdepth}{3}
% you can raise or lower the level without knowing its number.
\addtocounter{tocdepth}{1}
```

9.1.1 Adding entries manually

```
% file extension:
% toc: table of contents file
% lof: list of figures file
% lot: list of tables file

% sectional unit: part, chapter, section, subsection, paragraph, subparagraph
\addcontentsline{file extension}{sectional unit}{text}

```

```
% In contrary to \addcontentsline, the argument entry is written directly to the file
% without any additional formatting. You may choose any formatting you like.
\addtocontents{file extension}{entry}


% Examples
\addcontentsline{toc}{chapter}{Preface}
\addcontentsline{toc}{part}{Appendix}

\addtocontents{toc}{\bigskip}
% extends the text height such that one additional line fits to the contents page.
\addtocontents{toc}{\protect\enlargethispage{\baselineskip}}
% causes a page break in the TOC.
\addtocontents{toc}{\protect\newpage}
% changes the page style of the current TOC page to fancy.
\addtocontents{toc}{\protect\thispagestyle{fancy}}
```

9.2 Creating and Customizing Lists of Figures

```
% renamed the figures and the list heading
\renewcommand{\figurename}{Diagram}
\renewcommand{\listfigurename}{List of Diagrams}
\listoffigures
```

9.3 Creating and Customizing Lists of Tables

```
\renewcommand{\tablename}{Diagram}
\renewcommand{\listtablename}{List of Diagrams}
\listoffigures
```

9.4 Generating an Index

Steps to generating index list:

1 load the index package and add the command to create the index

```
\usepackage{index}
\makeindex{}
```

2 mark index point

```
% simple entry
\index{entry}
% example \index{enterprise}

% subentry
\index{entry!subentry}
% example \index{enterprise!organization}

```

```
% subsubentry
\index{entry!subentry!subsubentry}
% example \index{enterprise!organization!operation}

This will be written to a file with the extension .idx.
```

3 create an entry for the index for the table of contents

```
\clearpage
\addcontentsline{toc}{chapter}{Index}
```

4 in the next line, typeset the index

```
\printindex{}
```

5 use shell command to typeset the tex document

```
xelatex latex.tex               # .tex is optional
# or
pdflatex latex.tex              # .tex is optional
```

6 use shell command to produce `.idx` file.

```
makeindex latex.idx             # .idx is optional
```

7 typeset the tex document again, refer to **5**

9.4.1 Specifying Page Ranges

```
% Example
\index{network|(}
...
\index{network|)}
```

9.4.2 Using Symbols in the Index

`makeindex` sorts the entries alphabetically. If you would like to include symbols in the index, for example, Greek letters, chemical formulas, or math symbols, you could face the problem of integrating them into the sorting. For this purpose, `\index` understands a sort key. Use this key as prefix for the entry, separated by an @ symbol, for instance:

```
\index{Gamma@$\Gamma$}
```

9.4.3 Referring to Other Index Entries

Different words may stand for the same concept. For such cases, it's possible to add a cross-reference to the main phrase without a page number. Adding the code `see{entry list}` achieves that, for example:

```
\index{network|see{WLAN}}
\index{WLAN}
```

As such references don't print a page number, their position in the text doesn't matter. You could collect them in one place of your document.

9.4.4 Fine-tuning Page Numbers

If an index entry refers to several pages, you might want to emphasize one page number to indicate it as the primary reference. You could define a command for emphasizing as follows:

```
\newcommand{\main}[1]{\emph{#1}}
\index{WLAN|main}
```

9.4.5 Designing the Index Layout

LaTeX provides some index styles called **latex** (the default), **gind, din**, and **iso**. To use another style, specify it using the `-s` option of the makeindex program, for example:

```
makeindex -s iso latex
```

9.5 Creating a Bibliography

```
\begin{thebibliography}{widest label}
   \bibitem[label]{key} author, title, year etc.
   \bibitem...
   ...
\end{thebibliography}
```

```
% Example
To study \TeX\ in depth, see \cite{DK86}.
For writing math texts, see \cite{DK89}.


\begin{thebibliography}{8}
\bibitem{DK86} D.E. Knuth, \emph{The {\TeX}book}, 1986
\bibitem{DK89} D.E. Knuth, \emph{Typesetting Concrete Mathematics}, 1989
\end{thebibliography}
```

Each item is specified using the command `\bibitem`. This command requires a mandatory argument determining the `key`. We may simply refer to this key by `\cite{key}` or `\cite{key1,key2}`. `\cite` accepts an optional argument stating a page range, for example, `\cite[p.\,18--20]{key}`. You may choose a label by the optional argument of `\bibitem`. If no label has been given, LaTeX will number the items consecutively in square brackets.

9.5.1 Using Bibliography Databases With Bibtex

1 Create a new document. For example `latex.bib`.

```
@book{DK86,
author = "D.E. Knuth",
title = "The {\TeX}book",
publisher = "Addison Wesley",
year = 1986
}

@article{DK89,
author = "D.E. Knuth",
```

```
title = "Typesetting Concrete Mathematics",
journal = "TUGboat",
volume = 10,
number = 1,
pages = "31--36",
month = apr,
year = 1989
}
```

2 Include the database in to your tex document. For example `latex.tex`.

```
\bibliographystyle{alpha}        % plain, unsrt, alpha, abbrv
\bibliography{latex}             % latex stands for latex.bib here
```

3 Typeset one time with **pdfLaTeX** or **xelatex**.

```
xelatex latex
```

4 `bibtex` document.

```
bibtex latex                     # here latex is the documentname
```

5 Typeset again the tex file.

9.6 Changing the Headings

You can use `\renewcommand` to change the headings.

List	Heading Command	Default heading
Table of contents	`\contentsname`	**Contents**
List of figures	`\listfigurename`	**List of figures**
List of tables	`\listtablename`	**List of tables**
Bibliography	`\bibname in book and report`	**Bibliography** in book and report
	`\refname in article`	**References** in article
Index	`\indexname`	**index**

Tables 9.2: Headings name

Name	Heading Command	Default heading
figure	`\figurename`	**Figure**
table	`\tablename`	**Table**
part	`\partname`	**Part**
chapter	`\chaptername`	**Chapter**
abstract	`\abstractname`	**Abstract**
appendix	`\appendixname`	**Appendix**

Tables 9.3: Macros name

Chapter 10

Math Formulas

10.1 Math Mode

Using math environments to enter math mode.

10.1.1 Embedding Math Expressions Within Text

LaTeX provides the math environment in-text formulas:

```
\begin{math}
  expression
\end{math}
```

Since it's very laborious to write this environment for each small expression or symbol, LaTeX offers an alias that's doing the same:

```
\(
expression
\)
% or
\(expression\)
```

A third way is by using a shortcut, coming from TeX:

```
$expression$
```

10.1.2 Displaying Formulas

For displayed formulas, which have to be centered, LaTeX offers the displaymath environment:

```
\begin{displaymath}
  expression
\end{displaymath}
```

The effect of this environment is that the paragraph will be ended, some vertical space follows, then the centered formula plus the following vertical space. As this math environment takes care of the spacing, don't leave empty lines before and after it! This would cause additional vertical space because of the superfluous paragraph breaks.

A shortcut is:

```
\[
expression
\]
```

10.1.3 Numbering Equations

Equations and formulas in general may be numbered. However, this applies only to displayed formulas. The equation environment is responsible for this:

```
\begin{equation}
  \label{key}
  expression
\end{equation}
```

10.2 Common Formulas

Source code	Output
`x^2`	x^2
`x_2`	x_2
`\sqrt[3]{x}`	$\sqrt[3]{x}$
`\frac{x}{y}`	$\frac{x}{y}$

Tables 10.1: Common

10.3 Dots

Source code	Output
`\ddots`	$\ddots$
`\cdot`	$\cdot$
`\ldots`	$\ldots$
`\vdots`	$\vdots$
`\dot{}`	$\dot{}$
`\cdots`	$\cdots$

Tables 10.2: Dots

10.4 Greek Letters

To get a lowercase Greek letter, just write the name with a backslash for the command.

Source code	Output	Source code	Output
\alpha	α	\beta	β
\gamma	γ	\delta	δ
\epsilon	ϵ	\zeta	ζ
\eta	η	\theta	θ
\iota	ι	\kappa	κ
\lambda	λ	\mu	μ
\nu	ν	\xi	ξ
o	o	\pi	π
\rho	ρ	\sigma	σ
\tau	τ	\upsilon	υ
\phi	ϕ	\chi	χ
\psi	ψ	\omega	ω
\varepsilon	ε	\vartheta	ϑ
\varpi	ϖ	\varrho	ϱ
\varsigma	ς	\varphi	φ
\Gamma	Γ	\Delta	Δ
\Theta	Θ	\Lambda	Λ
\Xi	Ξ	\Pi	Π
\Sigma	Σ	\Upsilon	Υ
\Phi	Φ	\Psi	Ψ
\Omega	Ω		

Tables 10.3: Greek Letters

10.5 Fonts

Source code	Package	Output
\mathrm{...}		abc 123
\mathit{...}		*abc 123*
\mathsf{...}		abc 123
\mathbb{...}	amsfonts	$\mathbb{ABC}$
\mathbbm{...}	bbm	$\mathbb{ABC}$
\mathds{...}	dsfont	$\mathbb{ABC}$
\mathfrak{...}	eufrak	$\mathfrak{ABC}$ 123
\mathnormal{...}		ABC 123

Tables 10.4: Fonts

10.6 Multi-line Formulas

```
% package amsmath needed
\begin{multline}
\sum = a + b + c + d + e \\
```

```
            + f + g + h + i + j \\
            + k + l + m + n
\end{multline}


\begin{gather}
x + y + z = 0 \\
y-z= 1
\end{gather}


\begin{align}
  x + y + z &= 0 \\
  y - z &= 1
\end{align}
```

$$\sum = a+b+c+d+e + f+g+h+i+j + k+l+m+n \quad (10.1)$$

$$x+y+z=0 \quad (10.2)$$
$$y-z=1 \quad (10.3)$$

$$x+y+z=0 \quad (10.4)$$
$$y-z=1 \quad (10.5)$$

10.7 Operators

Trigonometric functions, logarithm functions, and other analytic and algebraic functions are commonly written with upright Roman letters. Simply typing log would otherwise look like a product of the three variables, namely, l, o, and g. To ease the input, there are commands for many common functions or so called **operators**. Here's an alphabetical list of the predefined ones:

```
\arccos, \arcsin, \arctan, \arg, \cos, \cosh, \cot, \coth, \scs, \deg, \det,
    \dim, \exp, \gcd, \hom, \inf, \ker, \lg, \lim, \liminf, \limsup, \ln, \log
    , \max, \min, \Pr, \sec, \sin, \sinh, \sup, \tan, \tanh
```

10.8 Standard LaTeX Symbols

Source code	Output	Source code	Output
\circ	$\circ$	\bigcirc	$\bigcirc$
\star	$\star$	\ast	$\ast$
\cup	$\cup$	\cap	$\cap$
\ominus	$\ominus$	\oplus	$\oplus$
\oslash	$\oslash$	\otimes	$\otimes$
\times	$\times$	\div	$\div$
\pm	$\pm$	\mp	$\mp$
\odot	$\odot$	\bullet	$\bullet$
\approx	$\approx$	\equiv	$\equiv$
\propto	$\propto$	\sim	$\sim$
\simeq	$\simeq$		
\parallel	$\parallel$	\perp	$\perp$
\subset	$\subset$	\supset	$\supset$
\subseteq	$\subseteq$	\supseteq	$\supseteq$
\geq	$\geq$	\gg	$\gg$
\leq	$\leq$	\ll	$\ll$
\neq	$\neq$		
\prod	$\prod$	\sum	$\sum$
\coprod	$\coprod$	\int	$\int$
\oint	$\oint$		
\rightarrow	$\rightarrow$	\Rightarrow	$\Rightarrow$
\longrightarrow	$\longrightarrow$	\Longrightarrow	$\Longrightarrow$
\hookleftarrow	$\hookleftarrow$	\hookrightarrow	$\hookrightarrow$
\leftrightarrow	$\leftrightarrow$	\Leftrightarrow	$\Leftrightarrow$
\bot	$\bot$	\forall	$\forall$
\ni	$\ni$	\top	$\top$
\hbar	$\hbar$	\in	$\in$
\exists	$\exists$		
\langle	$\langle$	\lceil	$\lceil$
\lfloor	$\lfloor$	|	$\|$
\sharp	$\sharp$	\nabla	∇
\emptyset	$\emptyset$	\angle	$\angle$
\flat	$\flat$	\neg	$\neg$
\surd	$\surd$	\infty	∞
\prime	$\prime$	\triangle	$\triangle$

Tables 10.5: Standard latex sysmbols

10.9 Math Structures

```
\[
\binom{n}{k} = \frac{n!}{k!(n-k)!}
\]
```

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

```
\[
A =
\begin{pmatrix}
  a_{11} & a_{12} \\
  a_{21} & a_{22}
\end{pmatrix}
\]
```

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$$

These are amsmath's matrix environments:

Name	Delimiters of the matrix
matrix	no delimiters
pmatrix	parentheses()
bmatrix	square brackets[]
Bmatrix	braces{}
vamtrix	|
Vmatrix	|||
smallmatrix	without delimiters, add them if needed, more compact

Tables 10.6: Matrix

10.10 Stakcing Expressions

10.10.1 Underlining and Overlining

```
\[s = \overline{AB}\]
\[s = \underline{AB}\]
\[N = \underbrace{1 + 1 + \cdots + 1}_n\]
\[N = \overbrace{1 + 1 + \cdots + 1}_n\]
```

$$s = \overline{AB}$$

$$s = \underline{AB}$$

$$N = \underbrace{1 + 1 + \cdots + 1}_{n}$$

$$N = \overbrace{1 + 1 + \cdots + 1}^{n}$$

10.10.2 Setting Accents

Source code	Output	Source code	Output
\bar{a}	$\bar{a}$	\acute{a}	$\acute{a}$
\check{a}	$\check{a}$	\grave{a}	$\grave{a}$
\tilde{a}	$\tilde{a}$	\ddot{a}	$\ddot{a}$
\hat{a}	$\hat{a}$	\vec{a}	$\vec{a}$
\breve{a}	$\breve{a}$	\dot{a}	$\dot{a}$
\mathring{a}	$\mathring{a}$		
\widehat{a}	$\widehat{abc}$	\widetilde{a}	$\widetilde{a}$

Tables 10.7: Accents

10.10.3 Puting a Symbol Above Another

```
% package amsmath

\[\underset{A}{\equiv}\]
\[\overset{\equiv}{A}\]
```

$$\underset{A}{\equiv}$$

$$\overset{\equiv}{A}$$

Chapter 11

Using Fonts

11.1 Installing Additional Fonts

TeX distributions usually install a lot of fonts. A package manager allows the installation of further fonts, like mpm with MiKTeX or tlmgr with TeX Live.

TeX Live includes only freely licensed fonts. Non-free fonts may be installed using a separate program. It's called getnonfreefonts.

```
# Install getnonfreefonts
wget https://www.tug.org/fonts/getnonfreefonts/install-getnonfreefonts
texlua install-getnonfreefonts
```

Using the following commands to get the usage of commands:

```
getnonfreefonts --user -h
getnonfreefonts --sys -h
```

11.2 Choosing the Main Font

```
\usepackage{lmodern}
```

Chapter 12

Developing Large Documents

12.1 Spliting the Input

Using "divide and conquer" thought to develop large documents, i.e. break down a document into several sub-documents.

There are two common commands to combine the sub-documents into a large document:

```
\input{filename}
\include{filename}
```

When LaTeX encounters `\input` command, it reads in the file with the name `filename` exactly as if its contents have been typed at that point. Accordingly, all commands in this file would be processed by the LaTeX compiler. You can even nest `\input` — this command may be used inside an included file.

The argument is treated the same way as `\input`. However, there are some important differences:

1. `\include` implicitly starts new pages. `\include{filename}` behaves like:

```
\clearpage
\include{filename}
\clearpage
```

2. `\include` cannot be nested.
3. `\include` supports a mechanism of choosing which parts of the document you wish to compile (`\includeonly`).

```
\includeonly{file list}
```

The argument may be a comma-separated list of filenames. If a file, `name.tex`, is not specified within this argument, `\include{name}` would not insert this file but just behave like

instead. This allows excluding chunks or whole chapters from compiling. If you work on a huge document, this speeds up compilation if you choose to include just your current chapter while keeping the labels and references of the excluded chapter this way.

12.2 Creating Front and Back Matter

Books often begin with introductory material such as copyright information, a foreword, acknowledgements, or a dedication. This part, including the title page and the table of contents, is called the **front matter**. At the end, a book might include an afterword and supporting material like a bibliography, and an index. This part is called the **back matter**.

```
\documentclass{article}
\begin{document}

% Pages are numbered with lowercase Roman numbers.
% Chapters generate a table of contents entry but don't get a number.
\frontmatter
\include{dedication}
\tableofcontents
\listoftables
\listoffigures

% Pages are numbered with Arabic numbers.
% Chapters are numbered and produce a table of contents entry.
\mainmatter
\include{chapter1}
\include{chapter2}

% Pages are numbered with Arabic numbers.
% Chapters generate a table of contents entry but don't get a number.
\backmatter
\include{proofs}
\nocite{*}
\bibliographystyle{alpha}
\bibliography{tex}              % use tex.bib

\end{document}
```

12.3 Creating a Title Page

```

\begin{titlepage}

\newcommand{\HRule}{\rule{\linewidth}{0.5mm}} % Defines a new command for the
    horizontal lines, change thickness here

\center % Center everything on the page

%
    ----------------------------------------------------------------------------
```

```
%       HEADING SECTIONS
%
    ----------------------------------------------------------------------------------------

\includegraphics[width=0.5\textwidth]{images/logo.png}\\[1cm] % Include a
    department/university logo - this will require the graphicx package

%
    ----------------------------------------------------------------------------------------

%       TITLE SECTION
%
    ----------------------------------------------------------------------------------------

\HRule \\[0.4cm]
{ \huge \bfseries \LaTeX}\\[0.4cm] % Title of your document
\HRule \\[1.5cm]

%
    ----------------------------------------------------------------------------------------

%       AUTHOR SECTION
%
    ----------------------------------------------------------------------------------------

\begin{minipage}{0.4\textwidth}
\begin{center} \large
Mingming \textsc{Li}\\ % Your name
\end{center}

\end{minipage}\\[2cm]


%
    ----------------------------------------------------------------------------------------

%       DATE SECTION
%
    ----------------------------------------------------------------------------------------

{\large \today}\\[2cm] % Date, change the \today to a set date if you want to
     be precise

\vfill % Fill the rest of the page with whitespace

\end{titlepage}
```

```
43
44 %%% Local Variables:
45 %%% mode: latex
46 %%% TeX-master: "latex"
47 %%% End:
```

U

Chapter 13

Using Packages

13.1 listings

```
\usepackage{listings}
```

This package provides the following commands or environments:

```
% inline code
\lstinline

% external code file
\lstinputlisting
```

```
\begin{lstlisting}

\end{lstlisting}
```

13.2 xspace

```
  \usepackage{xspace}
```

This package provides the command that inserts a space depending on the following character: If a dot, a comma, an exclamation, or a quotation mark follows, it won't insert a space, but if a normal letter follows, then it will. Usually, that's exactly what we want.

```
  \newcommand{\TUG}{\textsc{\TeX\ Users Group}\xspace}
```

13.3 url

```
  \usepackage{url}
```

This package will provide the command `\url`. This command takes an address for the argument and will print it out with typewriter font. Furthermore, it is able to handle special characters in addresses like underscores and percent signs. It even enables hyphenation in addresses, which is useful for websites with a very long name.

13.4 microtype

```
  \usepackage{microtype}
```

This package introduces font expansion to tweak the justification and uses hanging punctuation to improve the optical appearance of the margins. This may reduce the need of hyphenation and improves the "grayness" of the output.

13.5 inputenc

```
\usepackage[utf8]{inputenc}
```

We loaded the inputenc package. The option utf8 tells the package to use Unicode input encoding, which provides many more symbols than just the ASCII code. Now we just need to find the symbol on the keyboard and to type it.

13.6 parskip

```
\usepackage{parskip}
```

It remove the paragraph indentation completely. At the same time, this package introduces a skip between paragraphs.

13.7 geometry

```
\usepackage[a4paper, inner=1.5cm, outer=3cm, top=2cm,
       bottom=3cm, bindingoffset=1cm]{geometry}
```

This package can be used to adjust margins.

The geometry package understands arguments of the form "key=value", separated by commas. If you load geometry without arguments, those arguments could alternatively be used by calling `\geometry{argument list}`.

13.8 setspace

```
\usepackage[onehalfspacing]{setspace}
```

It is used to adjust the line spacing. It understand 3 options: `singlespacing`, `onehalfspacing` and `doublespacing`.

13.9 fancyhdr

```
% fancy header
\usepackage{fancyhdr}
% clear the headers and footers
\fancyhf{}
% \leftmark is used by the book class to store the
% chapter title together with the chapter number.
% LE stands for left-even and means that this chapter
% title will be put on the left side of the header
% on even-numbered pages.
\fancyhead[LE]{\leftmark}
% \rightmark is used by the book class to store
% the section title together with its number.
% RO stands for right-odd and means that this section
% heading shall be displayed on right side of the
% header on odd-numbered pages.
```

```
\fancyhead[RO]{\nouppercase{\rightmark}}
% \thepage prints the page number.
\fancyfoot[LE,RO]{\thepage}
% All those commands are used to modify a page style
% provided by fancyhdr; this style is called fancy.
% We had to tell LaTeX to use this style and we did
% it through \pagestyle{fancy}.
\pagestyle{fancy}
```

```
\fancyhead[code]{text}
\fancyfoot[code]{text}
```

code may consist of one or more letters:

- L: left
- R: right
- C: center
- E: even page
- O: odd page
- H: header
- F: footer

LaTeX and its base classes provide four page styles:

- empty: Neither a header nor a footer is shown.
- plain: No header. The page number will be printed and centered in the footer.
- headings: The header contains titles of chapters, sections, and/or subsections, depending on the class and also the page number. The footer is empty.
- myheadings: The header contains a user-defined text and the page number; the footer is empty.

fancyhdr adds one page style:

- fancy: Both the header and footer may be customized by the user.

Two commands may be used to choose the page style:

- \pagestyle{name}: Switches to the page style name from this point onwards.
- \thispagestyle{name}: Chooses the page style name only or the current page; the following pages will have the style that's been used before.

We can introduce or delete lines between header and body text and body text and footer, respectively, with these two commands:

```
\renewcommand{\headrulewidth}{width}
\renewcommand{\footrulewidth}{width}
```

13.10 paralist

```
\usepackage{paralist}
```

paralist provides several new list environments designed to be typeset within paragraphs or in a very compact look. We loaded this package and replaced the standard environments with their compact counterparts.

For each standard environment, paralist adds three corresponding environments:

Numbered lists:

- compactenum: Compact version of the enumerate environment without any vertical space before or after the list or its items.

- inparaenum: An enumerated list typeset within a paragraph.
- asparaenum: Every list item is formatted like a separate common LaTeX paragraph, but numbered.

Bulleted lists:

- compactitem:
- inparaeitem:
- asparaitem

Description lists:

- compactdesc
- inparadesc
- asparadesc

13.11 enumitem

```
\usepackage{enumitem}
```

This package provide sophisticated features to define numbered and bulleted lists.

```
\usepackage{enumitem}
% \setlist sets properties valid for all types of lists.
% Here we specified nolistsep to achieve very compact lists analogous
% to the compact paralist environment.
\setlist{nolistsep}
% \setitemize modifies properties of bulleted lists.
\setitemize[1]{label=---}
% \setenumerate sets properties valid for numbered lists.
% \alph, \Alph, \arabic, \roman and \Roman
\setenumerate[1]{label=\textcircled{\scriptsize\Alph*},font=\sffamily{}}
```

All this three commands allow arguments of the form key=value. Some useful parameters are:

- font
- label
- align
- start
- resume
- noitemsep
- nolistsep

It also support:

```
\setdescription[level]{k=v}
```

13.12 array

```
\usepackage{array}
```

This package provide some options to tabular:

- m{width} is similar to \parbox{width}, the base line is at the middle.
- b{width} is similar to \parbox[b]{width}, the base line is at the bottom.
- !{code} can be used like | but inserts code instead of a vertical line.

- ♠ `>{code}` can be used before an `l, c, r, p, m`, or `b` option and inserts code right at the beginning of each entry of that column.
- ♠ `<{code}` can be used after an `l, c, r, p, m`, or `b` option and inserts code at the end of the entry of that column.

```
\begin{tabular}{@{}lp{1.2cm}m{1.2cm}b{1.2cm}@{}}
  \hline
  baseline & aligned at the top & aligned at the middle
  & aligned at the bottom\\
  \hline
\end{tabular}
```

baseline	aligned at the top	aligned at the middle	aligned at the bottom

The `array` package introduces a length called `\extrarowheight`. If it has a positive value, this will be added to the height of every row of the table.

```
\setlength{\extrarowheight}{4pt}
\begin{tabular}{@{}>{\itshape}ll!{:}l<{.}@{}}
  \hline
  Info: & Software & \LaTeX\\
        & Author & Leslie Lamport\\
        & Website & www.latex-project.org\\
  \hline
\end{tabular}
```

Info:	Software	: LaTeX.
	Author	: Leslie Lamport.
	Website	: www.latex-project.org.

13.13 booktabs

```
\usepackage{booktabs}
```

This package provides commands to beauty the table lines.

- ♠ `\toprule[thickness]` may be used to draw a horizontal line at the top of the table. If desired, a thickness may be specified, like 1pt or 0.5mm.
- ♠ `\midrule[thickness]` draws a horizontal dividing line between rows of a table.
- ♠ `\bottomrule[thickness]` draws a horizontal line to finish off a table.
- ♠ `\cmidrule[thickness](trim){m-n}` draws a horizontal line from column `m` to column `n`. (`trim`) is option, it could be (`l` or `r`) to trim the line at its left or right end.

The package does not define vertical lines.

```
\setlength{\heavyrulewidth}{2pt} % set top bottom line width
  \begin{tabular}{ccc}
    \toprule % British typesetters call a line a rule
    \head{Command} & \head{Declaration}& \head{Output}\\
    \midrule %
```

```
    \verb|\textrm| & \verb|\rmfamily| & \rmfamily Example text \\
    \verb|\textsf| & \verb|\sffamily| & \sffamily Example text \\
    \verb|\texttt| & \verb|\ttfamily| & \ttfamily Example text \\
    \bottomrule %
  \end{tabular}
```

Command	Declaration	Output
\textrm	\rmfamily	Example text
\textsf	\sffamily	Example text
\texttt	\ttfamily	Example text

13.14 multirow

```
\usepackage{multirow}
```

```
\begin{tabular}{@{}l*2{>{\textbackslash\ttfamily}l}l<{Example text}@{}}
  \toprule
  & \multicolumn{2}{c}{\head{Input}} & \multicolumn{1}{c}{\head{Output}}\\
  & \normal{\head{Command}} & \normal{\head{Declaration}} & \normal{}\\
  \cmidrule(lr){2-3}\cmidrule(l){4-4}
  \multirow{3}{*}{Family} & textrm&rmfamily & \rmfamily\\
  & textsf & sffamily & \sffamily\\
  & texttt & ttfamily & \ttfamily\\
  \bottomrule
\end{tabular}
```

	Input		Output
	Command	Declaration	
Family	\textrm	\rmfamily	Example text
	\textsf	\sffamily	Example text
	\texttt	\ttfamily	Example text

13.15 caption

```
\usepackage[font=large,labelfont=bf,margin=1cm]{caption}
```

Through this package, you could enhance the visual appearance of all of your captions.

13.16 graphicx

```
\usepackage{graphicx}
```

This package is used to insert figure into your document.

13.17 pdfpages

```
\usepackage{pdfpages}
```

It provides a command, `\includepdf`, which is able to include a complete page and even a multi-page PDF document at once.

13.18 eso-pic

```
\usepackage{eso-pic}
```

This package makes it easy to add some picture commands to every page at absolute positions. So it can be used for watermarks, background images.

13.19 textpos

```
\usepackage{textpos}
```

This package facilitates placing boxes at absolute positions on the LATEX page. It provides the following environment:

```
\begin{textblock}{hsize}(hpos,vpos)
text...
\end{textblock}
```

So it can be used for watermarks, background images.

placeins

It may happen that tables and figures float far away, perhaps even into another section. The placeins package provides a useful command to restrict the floating. If you load placeins with `\usepackage{placeins}` and write `\FloatBarrier` somewhere in your document, no table or figure could float past it. This macro keeps floats in their place.

A very convenient way to prevent floats from crossing section boundaries is stating the section option:

```
\usepackage[section]{placeins}
```

This option causes an implicit `\FloatBarrier` to be used at the beginning of each section.

13.20 float

```
\usepackage{float}
```

This package introduces the placement option H causing the float to appear right there.

13.21 wrapfig

```
\usepackage{wrapfig}
```

This package provides environments wrapfigure and wraptable to let text flow around a table or a figure.

```
\begin{wrapfigure}[number of lines]{placement}[overhang]{width}

\end{wrapfigure}
```

13.22 subfig

```
\usepackage{subfig}
```

It is a sophisticated package supporting inclusion of small figures and tables. It takes care of positioning, labeling, and captioning within single floats.

13.23 varioref

```
\usepackage{varioref}
```

This package defines the commands `\vref, \vpageref, \vrefrange`, and `\vpagerefrange`. `\vref` is similar to `\ref` but adds an additional page reference, like 'on the facing page' or 'on page 27' whenever the corresponding `\label` is not on the same page. The command `\vpageref` is a variation to **??**ith a similar functionality. The `\vpagerefrange` commands take two labels as arguments and produce strings which depend on whether or not these labels fall onto a single page or on different pages. Generated strings are customizable so that these commands are usable with various languages.

13.24 xr

```
\usepackage{xr}
\externaldocument[A-]{aaa}
```

This package implements a system for eXternal References.

If one document needs to refer to sections of another, say `aaa.tex`, then this package may be loaded in the main file, and the command `\externaldocument{aaa}` given in the preamble. Then you may use `\ref` and `\pageref` to refer to anything which has been given a `\label` in either `aaa.tex` or the main document. You may declare any number of such external documents. If any of the external documents, or the main document, use the same `\label` then an error will occur as the label will be multiply defined. To overcome this problem `\externaldocument` has an optional argument. If you declare `\externaldocument[A-]{aaa}` Then all references from `aaa` are prefixed by `A-`. So for instance, if a section of `aaa` had `\label{intro}`, then this could be referenced with `\ref{A-intro}`.

13.25 hyperref

```
\usepackage{hyperref}
```

This package provides hyperlink capability. It provides the following the link commands:

```
% makes text to a hyperlink, which points to the URL address
\href{URL}{text}
% prints the URL and links it
\url{URL}
% prints the URL without linking it
\nolinkurl{URL}
```

```
% changes text to a hyperlink, which links to the place
% where the label has been set, thus to the same place
% \ref{label} would point to
\hyperref{label}{text}
% creates a target name for potential hyperlinks
% with text as the anchor
\hypertarget{name}{text}
% makes text to a hyperlink, which points to the target name
\hyperlink{name}{text}
```

Sometimes you might need just an anchor, for instance, if you use `\addcontentsline`, which creates a hyperlinked TOC entry, but there hasn't been a sectioning command setting the anchor. The TOC entry would point to the previously set anchor, thus to the wrong place! The command `\phantomsection` comes to the rescue; it's just setting an anchor like `\hypertarget{}{}` would do. It's mostly used this way for creating a TOC entry for the bibliography while linking to the correct page as follows:

```
\cleardoublepage
\phantomsection
\addcontentsline{toc}{chapter}{\bibname}
\bibliography{name}
```

It also provides metadata property.

```
\hypersetup{
  colorlinks=true,
  linkcolor=red,
  pdfauthor={Mingming Li},
  pdftitle={Latex},
  pdfsubject={Latex},
  pdfkeywords={Latex,Emacs}
}
```

13.26 tocloft

```
\usepackage{tocloft}
```

This package provides means of controlling the typographic design of the Table of Contents, List of Figures and List of Tables. New kinds of 'List of ...' can be defined.

13.27 minitoc

```
\usepackage{minitoc}

\dominitoc
\dominilof
\dominilot


\chapter{Chapter}
\label{cha:chapter}

\minitoc
```

```
\mtcskip
\minilof
\mtcskip
\minilot
```

This package can create small TOCs for each part, chapter, or section.

13.28 tocbibind

```
\usepackage{tocbibind}
```

It can automatically add bibliography, index, TOC, LOF, and LOT to the table of contents.

13.29 index

```
\usepackage{index}
\makeindex{}

...
\index{network}
...

\clearpage{}
\addcontentsline{toc}{chapter}{Index}
\printindex
```

This package improves LaTeX's built-in indexing capabilities.

13.30 fontenc

```
\usepackage[T1]{fontenc}
```

This package is responsible for the output encoding: TeX macros are translated into special characters.

13.31 titlesec

```
\usepackage{titlesec}
```

It provide a consistent way to modify the headings.

```
\titleformat{cmd}[shape]{format}{label}{sep}{before}[after]
\titlespacing*{cmd}{left}{beforesep}{aftersep}[right]

% example
\titleformat{\chapter}[display]
{\normalfont\sffamily\Large\bfseries\centering}
{\chaptertitlename\ \thechapter}{0pt}{\Huge}
% section heading
\titleformat{\section}
{\normalfont\sffamily\large\bfseries\centering}
```

```
{\thesection}{1em}{}
% adjust the chapter headings spacing
% with star(*), the indentation of the following paragraph would be removed as you
know of sections. With drop, wrap and run-in the starred version has no meaning.
\titlespacing*{\chapter}{0pt}{30pt}{20pt}
```

The meaning of the arguments of `\titleformat` is as follows:

- ♠ `cmd` tands for the sectioning command we redefine, that is, `\part, \chapter, \section, \subsection, \subsubsection, \paragraph`, or `\subparagraph`
- ♠ `shape` specifies the paragraph shape. The effect of the possible values is:
  - `display` puts the label into a separate paragraph
  - `hang` creates a hanging label like in standard sections and is the default option
  - `runin` produces a run-in title like `\paragraph` does by default
  - `leftmargin` sets the title into the left margin
  - `rightmargin` puts the title into the right margin
  - `drop` wraps the text around the title, requires care to avoid overlapping
  - `wrap` works like drop but adjusts the space for the title to match the longest text line
  - `frame` works like display and additionally frames the title
- ♠ `format` may contains commands which will be applied to label and text of the title.
- ♠ `label` prints the label, that is, the number.
- ♠ `sep` is a length which specifies the separation between label and title text. With `display` option, it's the vertical separation, with `frame` option it means the distance between text and frame, otherwise it's the horizontal separation between label and title.
- ♠ `before` can contain code which comes before the title body. The last command of it is allowed to take an argument, which should then be the title text.
- ♠ `after` can contain code which comes after the title body.

13.32 color

```
\usepackage{color}
```

It provides the following commands:

```
% declaration that switches to the color name
\color{name}
% like {\color{name}}
\textcolor{name}{text}
% define your own color
\definecolor{name}{model}{color specification}
% \definecolor{light-blue}{rgb}{0.8,0.85,1}
```

13.33 xcolor

```
\usepackage{xcolor}
```

It extends the color facilities. It offers a lot of readily mixed colors; you just need to call it by its name and it has powerful capabilities regarding color definition

13.34 tikz

```
\usepackage{tikz}
```

It is an enormously capable package for creating graphics.

13.35 amsmath

```
\usepackage{amsmath}
```

It provides some math commands or environments. For example

```
\begin{multline}
\sum = a + b + c + d + e \\
           + f + g + h + i + j \\
           + k + l + m + n
\end{multline}


\begin{gather}
x + y + z = 0 \\
y-z= 1
\end{gather}


\begin{align}
  x + y + z &= 0 \\
  y - z &= 1
\end{align}
```

$$\sum = a + b + c + d + e + f + g + h + i + j + k + l + m + n \tag{13.1}$$

$$x + y + z = 0 \tag{13.2}$$
$$y - z = 1 \tag{13.3}$$

$$x + y + z = 0 \tag{13.4}$$
$$y - z = 1 \tag{13.5}$$

It also provide two commands to insert text into formulas:

```
% inserts text within a math formula.
\text{words}
```

```
% suspends the formula, the text follows in a separate paragraph, then the multi-line formula is resumed, keeping the alignment. Use it for longer text.
\intertext{text}
```

13.36 longtable

```
\usepackage{longtable}
```

```
\begin{center}
  \begin{longtable}[H]{l>{\bfseries}lp{0.6\textwidth}}
    \toprule
    \head{Group} & \head{Binding} & \head{Meaning}\\
    \midrule
    \endfirsthead

    \toprule
    \head{Group} & \head{Binding} & \head{Meaning}\\
    \midrule
    \endhead

    \midrule
    \multicolumn{3}{c}{{Continued on next page}}\\
    \bottomrule
    \endfoot

    \endlastfoot

    content
    ...
    content

    \bottomrule
    \caption{Dired commands}
    \label{tab:dired-commands}
  \end{longtable}
\end{center}
```

It is a popular package for creating multi-page table.

13.37 xsavebox

```
\usepackage{xsavebox}
\newsavebox{\lstbox}
```

This package provides some box environments to define box that can be used in footnote.

```
\begin{lrbox}{\lstbox}
\begin{lstlisting}[language=elisp, basicstyle=\footnotesize]
(with-eval-after-load 'org
  (define-key org-mode-map (kbd "M-n") #'org-next-link)
  (define-key org-mode-map (kbd "M-p") #'org-previous-link))
\end{lstlisting}
\end{lrbox}

\footnote{\usebox{\lstbox}}
```

13.38 tablefootnote

This package provides the `\tablefootnote` command to add footnote in a table.

13.39 fncychap

```
% Options: Sonny, Lenny, Glenn, Conny, Rejne, Bjarne, Bjornstrup
\usepackage[Lenny]{fncychap}
```

This package provides some predefined chapter settings.

13.40 fontawesome

This package provides some awesome social icons.

```
\faGithub, \faLinkedin, \faStackExchange, \faStackOverflow, \faHome
```

, , , ,

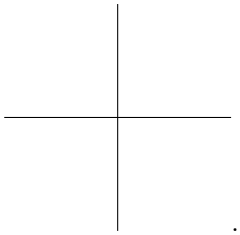
Chapter 14

TikZ

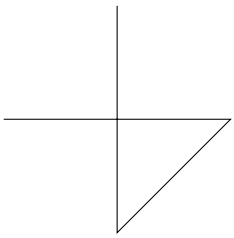
14.1 Basics

14.1.1 Setting up the endironment

```
\documentclass{article} % say
\usepackage{tikz}
\begin{document}
We are working on
\begin{tikzpicture}
  \draw (-1.5,0) -- (1.5,0);
  \draw (0,-1.5) -- (0,1.5);
\end{tikzpicture}.
\end{document}
```

We are working on  .

```
\tikz \draw (-1.5,0) -- (1.5,0) -- (0,-1.5) -- (0,1.5);
```



\tikz either takes one argument (starting with an opening braces) or collects everything up to the next semicolon and puts it inside a tikzpicture endironment.

14.1.2 Straight path

```
\draw (0,0) -- (1.5,0);
```

The coordinates are used to locate the positions and – is used for drawing.

14.1.3 Curved path

```
\filldraw [gray] (0,0) circle (2pt)
                 (1,1) circle (2pt)
                 (2,1) circle (2pt)
                 (2,0) circle (2pt);
 \draw (0,0) .. controls (1,1) and (2,1) .. (2,0);
```

You can leave out the **and** (second control point), which causes the first one to be used twice.

14.1.4 Circle path

```
\draw (0,0) circle (10pt);
```

```
\draw (0,0) ellipse (20pt and 10pt);
```

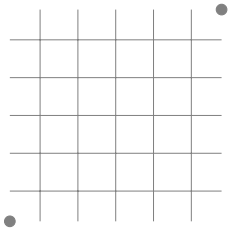
14.1.5 Rectangle path

```
\filldraw [gray] (0,0) circle (2pt);
\filldraw [gray] (0.5,0.5) circle (2pt);
  \draw (0,0) rectangle (0.5,0.5);
```

14.1.6 Grid path

```
  \filldraw [gray] (-1.4,-1.4) circle (2pt);
  \filldraw [gray] (1.4,1.4) circle (2pt);

  \draw[step=.5cm,gray,very thin] (-1.4,-1.4) grid (1.4,1.4);
```

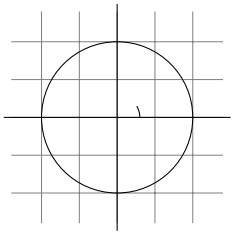


14.1.7 Arc path

```
  \filldraw [gray] (0,0) circle (2pt);
  \draw (0mm,0mm) arc (0:30:3cm);
  % (center) arc (angle1:angle2:radius)
  % an arc from angle1 to angle2 on a circle of radius
```

14.1.8 Clipping a path

```
\draw[step=.5cm,gray,very thin] (-1.4,-1.4) grid (1.4,1.4);
\draw (-1.5,0) -- (1.5,0);
\draw (0,-1.5) -- (0,1.5);
\draw (0,0) circle (1cm);
\draw (3mm,0mm) arc (0:30:3mm);
```



```
\clip (-0.1,-0.2) rectangle (1.1,0.75);
\draw[step=.5cm,gray,very thin] (-1.4,-1.4) grid (1.4,1.4);
\draw (-1.5,0) -- (1.5,0);
\draw (0,-1.5) -- (0,1.5);
\draw (0,0) circle (1cm);
\draw (3mm,0mm) arc (0:30:3mm);
```

In reality, \draw is just a shorthand for \path[draw] and \clip is a shorthand for \path[clip] and you could also say \path[draw,clip].

14.1.9 Filling

```
\fill[green!20!white] (0,0) -- (3cm,0cm) arc (0:30:3cm) -- cycle;
```


The –cycle causes the current path to be closed.

You can also fill and draw a path at the same time using the \filldraw command.

14.1.10 Shading

\shade and \shadedraw are used for shading and drawing at the same time.

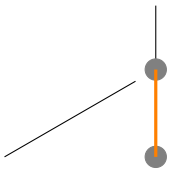
```
\shade (0,0) rectangle (2,1);
\shade[top color=yellow,bottom color=black] (3,0) rectangle +(2,1);
\shade[left color=yellow,right color=black] (6,0) rectangle +(2,1); %
    relative coordinate
\shadedraw[inner color=yellow,outer color=black,draw=yellow] (9,0)
    rectangle +(2,1);
\shade[ball color=green] (12,.5) circle (.5cm);
```

The default shading is a smooth transition from gray to white. To specify different colors, you can use options.

14.1.11 Specifying coordinates

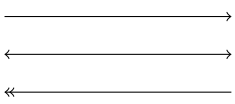
- If you leave out the unites, the default are set to cm and for angle to degree.
- + means a relative coordinate from the previous specified position and ++ means a relative coordinate from the previous specified position, making this the new specified position.
- You can use **intersection** to specify a coordinate.

```
\begin{tikzpicture}[scale=2]
  \draw (1,0) -- (1,1);
  \draw (0,0) -- (30:1cm);
  \filldraw [gray] (1,0) circle (2pt);
  \filldraw [gray] (intersection of 1,0--1,1 and 0,0--30:1cm) circle (2pt);
  \draw[very thick,orange] (1,0) -- (intersection of 1,0--1,1 and 0,0--30:1cm
      );
\end{tikzpicture}
```



14.1.12 Adding arrow tips

```
\begin{tikzpicture}
  \draw[->] (-1.5,0) -- (1.5,0);
  \draw[<->] (-1.5,-0.5) -- (1.5,-0.5);
  \draw[<<-] (-1.5,-1) -- (1.5,-1);
\end{tikzpicture}
```



```
\begin{tikzpicture}[>=stealth]  % >= right arrow tip kind
  \draw[->] (-1.5,0) -- (1.5,0);
\end{tikzpicture}
```


14.1.13 Scoping

Scope can let you apply graphic options to a local group.

```
\begin{tikzpicture}[ultra thick]
  \draw (0,0) -- (0,1);
  \begin{scope}[thin]
    \draw (1,0) -- (1,1);
    \draw (2,0) -- (2,1);
```

```
  \end{scope}
  \draw (3,0) -- (3,1);
\end{tikzpicture}
```

14.1.14 Transformations

When you specify a coordinate, TikZ applies certain transformations to the given coordinate in order to determine the finally position on the page.

```
\begin{tikzpicture}[even odd rule,rounded corners=2pt,x=10pt,y=10pt]
  % x=10pt set the x unit to 10pt
  \filldraw (0,0)   rectangle (1,1)
  [xshift=5pt,yshift=5pt]   (0,0)   rectangle (1,1)
  [rotate=30]   (-1,-1) rectangle (2,2);

\end{tikzpicture}
```

Options to do transformations:

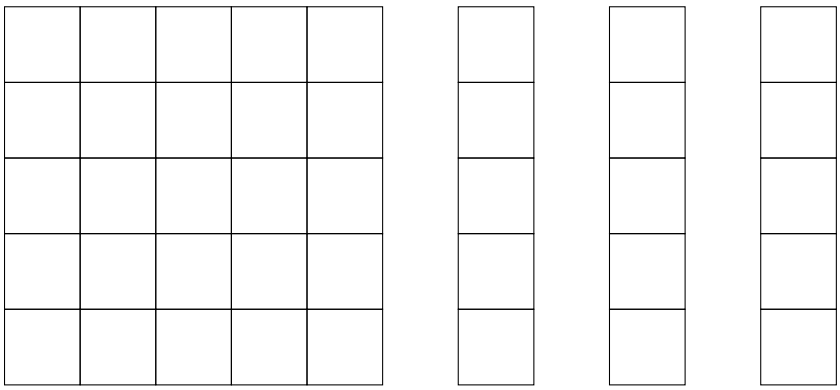
- **xshift** and **yshift**
- `shift={(1,0)}` for shifting to a given point
- **rotate** for rotating by a certain angle
- **rotate around** for rotating around a given point
- **scale** for scaling by a certain factor
- **xscale** and **yscale** (**xscale=-1** is a flip)
- **xslant** and **yslant** for slanting

14.1.15 For-loops

PGF introduces a command called **\foreach**. The general syntax is

```
\foreach variable in {list of values} command
```

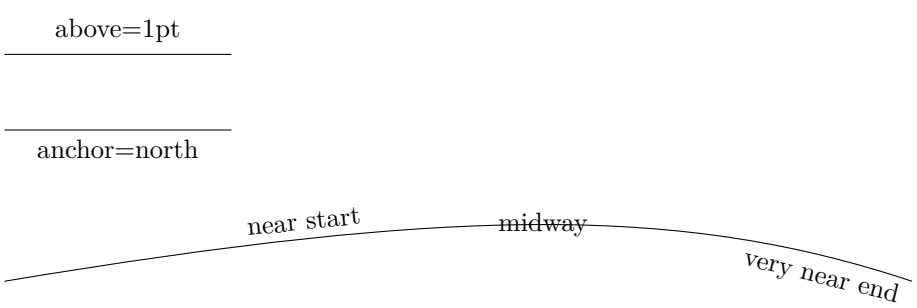
```
\begin{tikzpicture}
  \foreach \x in {1,2,...,5,7,9,...,12}
    \foreach \y in {1,...,5}
    {
      \draw (\x,\y) +(-.5,-.5) rectangle ++(.5,.5);
    }
\end{tikzpicture}
```



If you provide two numbers before the **...**, the **\foreach** statement will use their difference for the stepping.

14.1.16 Adding text

```
\begin{tikzpicture}
  \draw (0,0) -- node[above=1pt] {above=1pt} (3,0)
  (0,-1) -- node[anchor=north] {anchor=north} (3,-1);
  \draw (0,-3) .. controls (6,-2) and (9,-2) ..
  node[near start,sloped,above] {near start}
  node {midway}
  node[very near end,sloped,below] {very near end} (12,-3);
\end{tikzpicture}
```



When TikZ is constructing a path and encounters the keyword **node** in the middle of a path, it reads a "node specification". The keyword **node** is typically followed by some options and then some text between curly braces. This text is put inside a normal TEX box. All nodes are drawn only after the path has been completely drawn. You can determine the direction to the position with the **anchor** option. And there are simplified writing for the **anchor** option. **below** does the same as **anchor=south east**. You can also position labels on curves and, by adding the **sloped** option, have them rotated such that they match the line's slope.

14.1.17 Load library packages

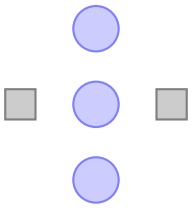
```
\usetikzlibrary{arrows,snakes,backgrounds}
```

14.1.18 Node

A node have a `position` and can have a `shape` and `name`.

The **\node** command is an abbreviation for **\path node**.

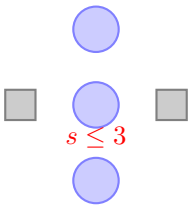
```
% shape (circle), style (blue!50,fill=blue!20,thick), size (inner sep=0pt,
    minimum size=6mm)
\tikzstyle{place}=[circle,draw=blue!50,fill=blue!20,thick,
                   inner sep=0pt,minimum size=6mm]
\tikzstyle{transition}=[rectangle,draw=black!50,fill=black!20,thick,
                        inner sep=0pt,minimum size=4mm]
\begin{tikzpicture}
  % option   name   coordinate  text
  \node[place]      (waiting 1)      at ( 0,2) {};
  \node[place]      (critical 1)     at ( 0,1) {};
  \node[place]      (semaphore)      at ( 0,0) {};
  \node[transition] (leave critical) at ( 1,1) {};
  \node[transition] (enter critical) at (-1,1) {};
\end{tikzpicture}
```



We can use relative coordinates and add label to a node.

```
\begin{tikzpicture}
  \tikzstyle{every label}=[red]
  \node[place]   (waiting)      {};
  \node[place]   (critical)     [below of=waiting]  {};
  \node[place]   (semaphore)    [below of=critical,
                                 label=above:$s\le3$] {};

  \node[transition] (leave critical) [right of=critical] {};
  \node[transition] (enter critical) [left of=critical]  {};
\end{tikzpicture}
```

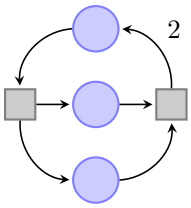


We can use **edge** to draw connection lines.

```
\tikzstyle{pre}=[<-,shorten <=1pt,>=stealth,semithick]
\tikzstyle{post}=[->,shorten >=1pt,>=stealth,semithick]
\begin{tikzpicture}[bend angle=45]
  \node[place]    (waiting)                          {};
  \node[place]    (critical)       [below of=waiting]  {};
  \node[place]    (semaphore)      [below of=critical] {};

  \node[transition] (leave critical) [right of=critical] {}
  edge [pre]                                  (critical)
  edge [post,bend right] node[auto,swap] {2} (waiting)
  edge [pre, bend left]                       (semaphore);
  \node[transition] (enter critical) [left of=critical]  {}
```

```
  edge [post]                                (critical)
  edge [pre, bend left]                      (waiting)
  edge [post,bend right]                     (semaphore);
\end{tikzpicture}
```

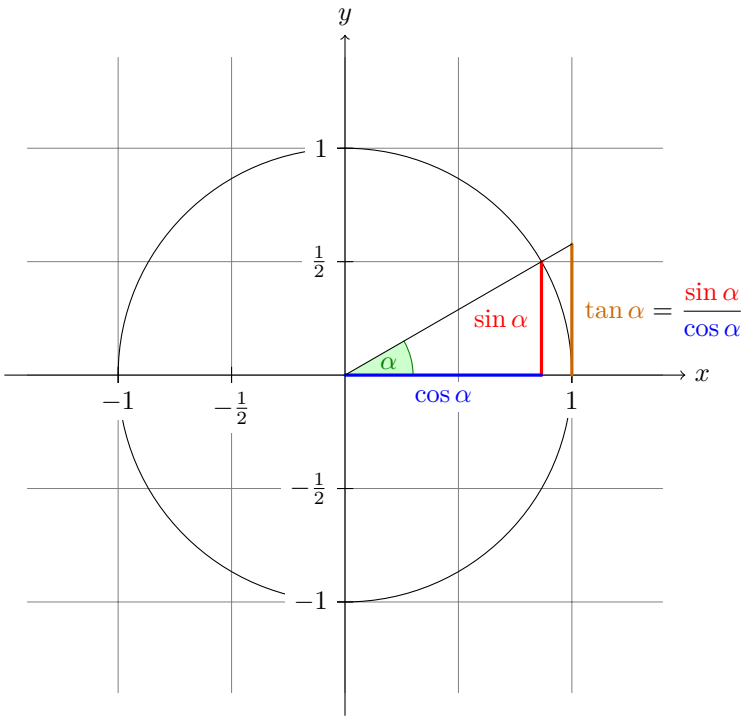


14.2 Examples

14.2.1 A picture for Karl's students

```
\begin{tikzpicture}[scale=3,cap=round]
% Local definitions
  \def\costhirty{0.8660256}
% Colors
  \colorlet{anglecolor}{green!50!black}
  \colorlet{sincolor}{red}
  \colorlet{tancolor}{orange!80!black}
  \colorlet{coscolor}{blue}
% Styles
  \tikzstyle{axes}=[]
  \tikzstyle{important line}=[very thick]
  \tikzstyle{information text}=[rounded corners,fill=red!10,inner sep=1ex]
% The graphic
  \draw[style=help lines,step=0.5cm] (-1.4,-1.4) grid (1.4,1.4);
  \draw (0,0) circle (1cm);
  \begin{scope}[style=axes]
    \draw[->] (-1.5,0) -- (1.5,0) node[right] {$x$} coordinate(x axis);
    \draw[->] (0,-1.5) -- (0,1.5) node[above] {$y$} coordinate(y axis);
    \foreach \x/\xtext in {-1, -.5/-\frac{1}{2}, 1}
      \draw[xshift=\x cm] (0pt,1pt) -- (0pt,-1pt) node[below,fill=white] {$\
          xtext$};
    \foreach \y/\ytext in {-1, -.5/-\frac{1}{2}, .5/\frac{1}{2}, 1}
      \draw[yshift=\y cm] (1pt,0pt) -- (-1pt,0pt) node[left,fill=white] {$\
          ytext$};
\end{scope}
  \filldraw[fill=green!20,draw=anglecolor] (0,0) -- (3mm,0pt) arc(0:30:3mm);
  \draw (15:2mm) node[anglecolor] {$\alpha$};
  \draw[style=important line,sincolor]
    (30:1cm) -- node[left=1pt,fill=white] {$\sin \alpha$} (30:1cm |- x axis);
  \draw[style=important line,coscolor]
    (30:1cm |- x axis) -- node[below=2pt,fill=white] {$\cos \alpha$} (0,0);
  \draw[style=important line,tancolor] (1,0) -- node[right=1pt,fill=white] {
    $\displaystyle \tan \alpha \color{black}=
    \frac{{\color{sincolor}\sin \alpha}}{\color{coscolor}\cos \alpha}$}
    (intersection of 0,0--30:1cm and 1,0--1,1) coordinate (t);
```

```
  \draw (0,0) -- (t);
  \draw[xshift=1.85cm]
    node[right,text width=6cm,style=information text]
    {
      The {\color{anglecolor} angle $\alpha$} is $30^\circ$ in the
      example ($\pi/6$ in radians). The {\color{sincolor}sine of
        $\alpha$}, which is the height of the red line, is
      \[
      {\color{sincolor} \sin \alpha} = 1/2.
      \]
      By the Theorem of Pythagoras ...
    };
\end{tikzpicture}
```



The angle α is 30° in the example ($\pi/6$ in radians). The sine of α, which is the height of the red line, is

$$\sin\alpha = 1/2.$$

By the Theorem of Pythagoras we have $\cos^2\alpha + \sin^2\alpha = 1/alpha = 1$. Thus the length of the blue line, which is the cosine of α, must be

$$\cos\alpha = \sqrt{1 - 1/4} = \frac{1}{2}\sqrt{3}$$

This shows that $\tan\alpha$, which is the height of the orange line, is

$$\tan\alpha = \frac{\sin\alpha}{\cos\alpha} = 1/\sqrt{3}$$

Chapter 15

Reference

```
% \usepackage{amssymb}
\checkmark{}
```

✓