Document Creation with LaTeX

Shaohao Chen and

Alexander B. Pacheco

HPC User Services

LSU HPC & LONI

sys-help@loni.org

HPC Training Fall 2013

Louisiana State University

October 30, 2013

Presented by Shaohao Chen

Slides made by Alexander B. Pacheco

Introduction

LaTeX Basics

Document Structure

Typesetting

User Customization

Bibiliography

Wrap Up

Introduction
    What is TeX?
    What is LaTeX?
    Advantages of LaTeX?
    Disadvantages of LaTeX?

- TeX is a low-level markup and programming language created by Donald Knuth to typeset documents attractively and consistently.

- TeX is a programming language in the sense that it supports the if-else construct: you can make calculations with it (that are performed while compiling the document), etc., but you would find it very hard to do anything else but typesetting with it.

- The fine control TeX offers over document structure and formatting makes it a powerful and formidable tool.

- TeX is renowned for being extremely stable, for running on many different kinds of computers, and for being virtually bug free.

- TeX is a popular means by which to typeset complex mathematical formulae; it has been noted as one of the most sophisticated digital typographical systems in the world.

- Programming in TeX generally progresses along a very gradual learning curve, requiring a significant investment of time to build custom macros for text formatting.

- Document preparation systems based on TeX, consisting of collections of pre-built macros, exist making it easier for the user to create documents without the need to learn the TeX language.

What is LaTeX?

- LaTeX is a macro package based on TeX created by Leslie Lamport.

- Its purpose is to simplify TeX typesetting, especially for documents containing mathematical formulae.

- Popular in academia, especially in mathematics, computer science, economics, engineering, physics, statistics, and quantitative psychology.

- Many of the academic publishing houses such as American Institute of Physics, Elsevier, etc provide templates to prepare manuscripts in LaTeX.

- Since LaTeX comprises a group of TeX commands, LaTeX document processing is essentially programming.

- Using LaTeX to create documents is a WYSIWYM (What You See Is What You Mean) approach rather than

  WYSIWYG (What You See Is What You Get) approach of Microsoft Word and Libre Office.

- In LaTeX, you create a text file in LaTeX markup, which then needs to be compiled to produce the final document, most commonly is postscript (ps) or portable document format (pdf).

- The final document can be viewed uniformly on any Operating System using any version of the document viewer.

- Document sources can be read with any text editor.

- You can concentrate purely on the structure and contents of the document, not get caught up with superficial layout issues.

- You don't need to manually adjust fonts, text sizes, line heights, or text flow for readability, as LaTeX takes care of them automatically.

- In LaTeX the document structure is visible to the user, and can be easily copied to another document.

- The layout, fonts, tables and so on are consistent throughout the document.

- Mathematical formulae can be easily typeset.

- Indexes, footnotes, citations and references are generated easily.

- Since the document source is plain text, tables, figures, equations, etc. can be generated programmatically with any language.

- You are forced to structure your documents correctly.

- LATEX is WYSIWYM and not WYSIWYG approach

  i.e. you can't see what the final version will look like while typing.

- You need to know the necessary commands for the markup language.

  i.e. there is no drop-down menu to create the document content such as equations, tables, inserting figures etc, you need to know how to enter those in a text editor.

- It can sometimes be difficult to obtain a certain look for the document.

- A document preparation system such as LaTeX is the combination of the TeX language and the macros.

  LaTeX : designed by Leslie Lamport, it is actually a set of macros for TeX. It aims at taking care of the formatting process.

  ConTEXT : has a very consistent and easy syntax and support for pdfTeX, XeTeX and LuaTeX engines.

  TeX : The original language designed by Donald Knuth

- Engines

  ◆ An engine is an executable that can turn your source code into a printable output format.

  ◆ The engine by itself only handles the syntax, it also needs to load fonts and macros to fully understand the source code and generate output properly.

  ◆ The engine will determine what kind of source code it can read, and what format it can output (usually DVI or PDF).

  pdftex,pdflatex : PDF compilers

  tex,latex : DVI compilers

  luatex,lualatex : A TeX engine with Lua scripting engine embedded

- Distributions

  - ◆ TeX distributions are collections of packages and programs (compilers, fonts, and macro packages) that enable you to typeset without having to manually fetch files and configure things.

    TeX Live : A cross platform TeX distribution `http://www.tug.org/texlive`

    MacTeX : A TeX Live based distribution targeting Mac OS X `http://www.tug.org/mactex`

    MikTeX : A TeX distribution for Windows `http://www.miktex.org`

- Installation:

  ❶ On Mac OSX, download the zip file and follow the instructions

  ❷ On Windows, MikTeX has an easy installer that takes care of setting up the environment and downloading core packages.

  ❸ On Linux, use the package manager (apt-get, yum, zypper etc) to download and install texlive and other additional packages.

  texlive should be present in main repositories. If not, your distribution web-site will have additional information.

● TeX and LaTeX source documents (as well as related files) are all text files, and can be opened and modified in almost any text editor.

● You should use a text editor (e.g. Notepad), not a word processor (Word, OpenOffice).

Type	Editor	Windows	Mac OS X	Linux
Text Editor	Vim	✓	✓	✓
	Emacs	✓	✓	✓
	gedit			✓
	kate			✓
	Notepad	✓		
WYSIWYG Editor	LyX	✓	✓	✓
	Bakoma TeX	✓	✓	✓
	Gummi	✓		✓
LaTeX Editor	TeXmaker	✓	✓	✓
	TeXstudio	✓	✓	✓
	TeXworks	✓	✓	✓
	Kile			✓
	TeXshop		✓	
	TeXnicle		✓	
	WinEdt	✓		
	TeXnicCenter	✓		
Web Based	Authorea	https://authorea.com		
	writeLATEX	https://www.writelatex.com		

LaTeX Basics
Syntax
Reserved Characters
Environments
Groups
Commands
Comments
Type Fonts
Type Size
My First LaTeX Document
Compiling LaTeX Document
Ancillary Files

- LaTeX uses a markup language in order to describe document structure and presentation.
- LaTeX converts your source text, combined with the markup, into a high quality document.
- For the purpose of analogy, web pages work in a similar way: the HTML is used to describe the document, but it is your browser that presents it in its full glory - with different colors, fonts, sizes, etc.
- "Whitespace" characters such as space or tab are treated uniformly as "space" by LaTeX.
- Several consecutive whitespace characters are treated as one "space".

```
Several␣consecutive␣whitespace␣characters␣such␣as␣␣␣␣␣␣␣␣␣␣are␣treated␣as␣one␣space
```

Several consecutive whitespace characters such as are treated as one space

Reserved Characters

- LaTeX has special characters or symbols that either have a special meaning or are bit available in all fonts.

- If you enter them directly in your text, they will normally not print.

- To print these symbols, you need to be escape with a \ except \ itself since \\ is reserved for line break.

Symbol	Command	Usage
#	\#	reference arguments
%	\%	Comment
$	\$	Math Mode
^	\^	Superscript
_	_	Subscript
&	\&	Alignment
{ }	\{ \}	wrap arguments
~	\~	produce non-breakable space
\	\textbackslash	escape character
\\	\\	line break

- Environments in LaTeX have a role that is quite similar to commands, but they usually have effect on a wider part of the document.

- Their syntax is:

```
\begin{environmentname}
text to be influenced
\end{environmentname}
```

- Between the \begin and the \end you can put other commands and nested environments.

- The internal mechanism of environments defines a group, which makes its usage safe (no influence on the other parts of the document).

- In general, environments can accept arguments as well, but this feature is not commonly used and so it will be discussed in more advanced parts of the document.

- Anything in LaTeX can be expressed in terms of commands and environments.

- A group is basically defined by a pair of braces.

- The range of commands put between braces is limited to them.

- Example

```
{\bf This is in bold}\\
{\em This is in italics}
This is normal text
```

This is in bold
This is in italics This is normal text

- LaTeX commands are case sensitive, and take one of the following two formats:

  - They start with a backslash \and then have a name consisting of letters only. Command names are terminated by a space, a number or any other "non-letter".

  - They consist of a backslash \and exactly one non-letter.

- Some commands need an argument, which has to be given between curly braces { } after the command name.

- Some commands support optional parameters, which are added after the command name in square brackets [ ].

- The general syntax is: `\commandname[option1,option2,...]{argument1}{argument2}...`

- Most standard LaTeX commands have a switch equivalent.

- Switches have no arguments but apply on the rest of the scope, i.e. the current group or environment.

- A switch should (almost) never be called outside of any scope, otherwise it will apply on the rest of the document.

```
{\bf This is in bold}\\
\em This is in italics
This is normal text
```

This is in bold
This is in italics This is normal text

- When LaTeX encounters a % character while processing an input file, it ignores the rest of the current line, the line break, and all whitespace at the beginning of the next line.

- This can be used to write notes into the input file, which will not show up in the printed version.

```
This␣is␣an␣%␣stupid
%␣Better:␣instructive␣<----
example:␣Supercal%
␣␣␣␣␣␣␣␣␣␣␣␣ifragilist%
icexpialidocious
```

> This is an example: Supercalifragilisticexpialidocious

- Note that the % character can be used to split long input lines that do not allow whitespace or line breaks, as with Supercalifragilisticexpialidocious above.

- The core LaTeX language does not have a predefined syntax for commenting out regions spanning multiple lines.

- The actual letters and symbols (collectively called type) that LaTeX produces are characterized by their style and size.

- A type style is specified by family, series and shape.

- Default font type is roman family, medium series and upright shape.

	STYLE	COMMAND	ALTERNATE DECLARATION	
FAMILY	roman	\textrm{roman}	{\rmfamily roman}	{\rm roman}
	sans serif	\textsf{sans serif}	{\sffamily sans serif}	{\sf sans serif}
	typewriter	\texttt{typewriter}	{\ttfamily typewriter}	{\tt typewriter}
SERIES	medium	\textmd{medium}	{\mdseries medium}	
	boldface	\textbf{boldface}	{\bfseries boldface}	{\bf boldface}
SHAPE	upright	\textup{upright}	{\upshape upright}	
	italics	\textit{italics}	{\itshape italics}	{\it italics}
	slanted	\textsl{slanted}	{\slshape slanted}	{\sl slanted}
	SMALL CAP	\textsc{small cap}	{\scshape small caps}	{\sc small caps}

- Type size is traditional measured in (printer) points.
- The default type produced by LaTeX documents is 10pt size.
- To change the type size, LaTeX has ten declarations available

size	{\tiny size}	size	{\large size}
size	{\ scriptsize size}	size	{\Large size}
size	{\footnotesize size}	size	{\LARGE size}
size	{\small size}	size	{\huge size}
size	{\normalsize size}	size	{\Huge size}

My First LaTeX Document

- Using your favorite text editor, create a file `hello.tex` that contains the following lines.

```
%My First LaTeX Document
\documentclass{article}
\begin{document}
Hello World!
\end{document}
```

- The first line is a comment. All comments begin with % symbol.
- The second line tells LaTeX to use the article document class.
- The \begin{document} command begins the actual document, while
- \end{document} command ends the document.
- The document content goes within \begin{document} and \end{document} commands.

- Run one of the following two sets of commands from the command prompt
  1. `latex hello`
  2. `dvips hello` (If you can view postscript files)
  3. `dvipdf hello`

OR

  1. `pdflatex hello`

- Open the pdf file created i.e. `hello.pdf`
- If your computer has a DVI or postscript viewer, you can view the `hello.dvi` or `hello.ps` files.

● LaTeX compilation creates a bunch of files with various extensions to store temporary data to use for next compilation

.aux : A file that transports information from one compiler run to the next. Among other things, the .aux file is used to store information associated with cross-references.

.log : Gives a detailed account of what happened during the last compiler run.

.toc : Stores all your section headers. It gets read in for the next compiler run and is used to produce the table of contents.

.lof : This is like .toc but for the list of figures.

.lot : And again the same for the list of tables.

.bbl : Bibliography file output by BiBTeX and used by LaTeX

.blg : BiBTeX log file. (errors are logged here)

.dvi : Device Independent File. This is the main result of a LaTeX compile run with latex. You can look at its content with a DVI previewer program or you can send it to a printer with dvips or a similar application.

Hello World!

Document Structure
    Preamble
    Document Types
    documentclass options
    Creating a Title Page
    Structuring a LaTeX Document
    Section Numbering
    Appendix
    Cross Referencing
    Table of Contents
    Structuring a LaTeX Document
    Adding Packages
    Modular Document

- When LaTeX processes an input file, it expects it to follow a certain structure.
- Every LaTeX input file must contain the commands,

```
\documentclass{...}

\begin{document}
...
\end{document}
```

- The area between \documentclass{...} and \begin{document} is called the **Preamble**.
- The document content goes between the \begin{document} and \end{document} commands,

```
\begin{document}
   ...
\end{document}
```

- The Preamble is anything that comes before the main document.
- It is used for
  - Defining the type of document.
  - Defining the top matter i.e. title, author, etc.
  - Applying global formatting including changing page layout from the default.
  - Including packages to add functionality.

- The first uncommented line of the LaTeX document needs to describe the type of document that you are creating using

  \documentclass[options]{documenttype}

- LaTeX can be used to create documents of various types

  ❶ article

  ❷ report

  ❸ book

  ❹ letter

  ❺ beamer[1], powerdot[2], prosper or seminar[3] for Presentations

- The difference between article, report and book is in the document structure and presentation:

- In article type, there is no chapter and the title page and document content can appear on the first page.

- In report and book, the title page is the first page and document content begins on the second page onwards.

- In article and report, there is an abstract environment to write the abstract of the article or report that you are writing.

---

[1] For Tutorial, visit http://www.hpc.lsu.edu/training/archive/tutorial.php

[2] THIS PRESENTATION, style file included in downloads

[3] Not popular anymore, use beamer or powerdot

documentclass options

- The options to documentclass are used to define a predetermined structure for the document.
- The most commonly used options are defining
  - ◆ font size: 10pt (default), 11pt or 12pt
  - ◆ paper size: letterpaper (default), legalpaper, executivepaper, a4paper, a5paper or b5paper
  - ◆ orientation: portrait (default) or landscape
  - ◆ page format: onecolumn (default) or twocolumn
- Options that depend on document type
  - ◆ Where to print page numbers for book, report and article

  oneside page numbers are printed the same on even and odd pages, default for article & report
  twoside page number appears on the right side for odd pages and on the left for even pages, default for book

  - ◆ Where new chapters begin in the book and report class

  openright chapters start on the right hand insert blank page if necessary i.e. odd numbered page
  openany chapters always start on the next page

  - ◆ Where the title appears
    - ★ In book and report classes, the title appears on the first page separate from the document content
    - ★ In article class, the title appears on the first page followed by the document content
    - ★ Use **titlepage** and **notitlepage** to override this standard behavior.

- Other options commonly used

  leqno : display equation numbers on the left rather than the default right

  fleqn : displayed formulas are flushed left instead of default centered

  draft : mark lines that are too wide with a thick black bar

  final : default, do not mark lines that are too wide.

- Add some options to documentclass to create your second document.

```
\documentclass[12pt,twocolumn,fleqn]{article}
```

- To create a title page LaTeX has three commands.
  - ◆ \ title {Title} where Title is the title of your article, book or report.
  - ◆ \author{FirstName LastName}, if there are multiple authors, list them all delimited by a comma (,) or and.
  - ◆ \date{\today} to set the date when the article was created i.e. today
    If the date required is different from today, add the date that you need as in \date{Feb. 29, 2016}
- If you are publishing a journal article, please see their LaTeX templates and style files. Most of their style and class files define additional commands such as \affiliation{...}, \institution{...}, etc.
- To create the actual page, you need to add \maketitle in your document i.e. after the \begin{document} command.
- The \maketitle is almost always the first line of your document content.

```
\documentclass[12pt,twocolumn,fleqn]{article}
\ title {Simple \LaTeX{} Document}
\author{Alex Pacheco, Bhupender Thakur, Feng Chen and Le Yan}
\date{\today}
\begin{document}
\maketitle
\end{document}
```

- Document Content i.e. everything between the \begin{document} and \end{document} is partitioned into

Section	Command	Level	Comment
part	\part{title}	-1	not in letters
chapter	\chapter{title}	0	only in book and report
section	\section{title}	1	not in letters
subsection	\subsection{title}	2	not in letters
subsubsection	\subsubsection{title}	3	not in letters
paragraph	\paragraph{title (optional)}	4	not in letters
subparagraph	\subparagraph{title (optional)}	5	not in letters

- LaTeX provides 7 levels of depth for defining sections. The depth levels for the various commands are listed in column three in the above table.

- The depth level of a section affects whether that section appears in the table of content or not. This can however be changed as we will see in the next few slides.

- Since LaTeX is used very often for writing scientific articles and reports, there are environments defined to create Abstract, Appendices and Bibliographies.

- Numbering of the sections is performed automatically by LaTeX.

- Parts get roman numerals (Part I, Part II, etc.);

- chapters and sections get decimal numbering, and

- appendices (which are just a special case of chapters, and share the same structure) are lettered (A, B, C, etc.).

- You can change the depth to which section numbering occurs, so you can turn it off selectively. By default it is set to 2.

- To change the depth level, use the \setcounter command.

- For example, to change depth to only include chapters: \setcounter{secnumdepth}{1}

- You can change the numbering mechanism of the sectioning commands as well as lists, captions, equations, tables, figures etc. We'll discuss more about this when we get to user defined commands.

- As most research papers have an abstract, there are predefined commands for telling LaTeX which part of the content makes up the abstract.

  - This should appear in its logical order, therefore, after the top matter, but before the main sections of the body.

  - *This command is available for the document classes article and report, but not book.*

  - In document class report, the abstract appears on a separate page without a page number.

  - In document class article, the abstract comes after the title heading on the first page.

```
\begin{abstract}
In this article we discuss how to create simple LaTeX documents. Topics include
    structuring a document, list environment, inserting equations and figures,
    creating tables and more.
\end{abstract}
```

Abstract

In this article we discuss how to create simple LaTeX documents. Topics include structuring a document, list environment, inserting equations and figures, creating tables and more.

- The following commands are available for producing automatic, sequential sectioning

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

- Except for \part, these commands form a sectioning hierarchy.

- In document class report and book, the highest sectioning level is \chapter while in article class, it is \section.

- The chapters are divided into sections using the \section command, which is further divided into subsections using the \subsection command and so on.

- The syntax for these commands is \command[short title]{title} or \command*{title}

- In the first form, the section is given the next number in the sequence which is then printed together with a heading using the text "title".

- The text "short title" becomes the entry in the table of contents and page head. If "short title" is omitted, then the "title" is used.

- In the second form (with *), no section number is printed and no entry is created in the table of contents.

- The highest sectioning command is given a single number (1,2,$\cdots$), the second highest command then creates a double number (1.1, 2.3, $\cdots$) and so on.

- The paragraph and subparagraph commands are not numbered.

- For each sectioning command, there is an internal counter that is incremented by one every time that command is called and reset to zero on every call to a higher sectioning command.

- The sectioning command, \part is a special case and does not affect the numbering of other sectioning commands.

- The \part are usually numbered with Uppercase Roman Numerals as in Part I, Part IV, etc.

- The \part is used to divide your document into multiple parts which can be independent of each other.

```
\documentclass{article}

\ title {Report of Projects I'm working on}
\author{Alex Pacheco}
\date{Oct 30, 2013}
\begin{document}
\maketitle
\part{First Project}
\section{Introduction}
Some stuff I'm doing
\section{Results}
Some useless results that I got
\section{Conclusion}
Unsuccessful Project ... keep working
\part{Second Project}
\section{Introduction}
Some stuff I'm doing
\section{Results}
More useless results
\section{Conclusion}
Unsuccessful Project ... keep working
\part{Third Project}
\section{Introduction}
Some stuff I'm doing
\section{Results}
Finally got some useful results
\section{Conclusion}
Successful Project ... boss says no, can't graduate ... keep
      working
\end{document}
```

```
\documentclass{report}

\ title {Report of Projects I'm working on}
\author{Alex Pacheco}
\date{Oct 30, 2013}
\begin{document}
\maketitle
\part{First Project}
\chapter{Project Title}
\section{Introduction}
Some stuff I'm doing
\section{Results}
Some useless results that I got
\section{Conclusion}
Unsuccessful Project ... keep working
\part{Second Project}
\chapter{Project Title}
\section{Introduction}
Some stuff I'm doing
\section{Results}
More useless results
\section{Conclusion}
Unsuccessful Project ... keep working
\part{Third Project}
\chapter{Project Title}
\section{Introduction}
Some stuff I'm doing
\section{Results}
Finally got some useful results
\section{Conclusion}
Successful Project ... boss says no, can't graduate ... keep
      working
\end{document}
```

Report of Projects I'm working on

Alex Pacheco

Oct 30, 2013

Part I
First Project

1 Introduction

Some stuff I'm doing

2 Results

Some useless results that I got

3 Conclusion

Unsuccessful Project ... keep working

Part II
Second Project

4 Introduction

Some stuff I'm doing

5 Results

More useless results

6 Conclusion

Unsuccessful Project ... keep working

Part III
Third Project

7 Introduction

Some stuff I'm doing

8 Results

Finally got some useful results

9 Conclusion

Successful Project ... boss says no, can't graduate ... keep working

Report of Projects I'm working on

Alex Pacheco

Oct 30, 2013

Part I

First Project

Chapter 1

Project Title

1.1 Introduction

Some stuff I'm doing

1.2 Results

Some useless results that I got

1.3 Conclusion

Unsuccessful Project ... keep working

Part II

Second Project

Chapter 2

Project Title

2.1 Introduction

Some stuff I'm doing

2.2 Results

More useless results

2.3 Conclusion

Unsuccessful Project ... keep working

Part III

Third Project

Chapter 3

Project Title

3.1 Introduction

Some stuff I'm doing

3.2 Results

Finally got some useful results

3.3 Conclusion

Successful Project ... boss says no, can't graduate ... keep working

- An appendix is introduced with the declaration `\appendix`

- The `\appendix` resets the section counter in article and chapter counter in book and report.

- The numbering for the sectioning commands is also changed from numerals to capital letters, A, B, ···

- The word "Chapter" is replaced by "Appendix" so that subsequent chapter headings are preceded by "Appendix A", "Appendix B", etc.

- The numbering of lower sectioning commands contain the letter in place of chapter number, for e.g. A.2.1

```
\appendix
\section{My First Appendix}
...
\subsection{Subsection in My First Appendix}
...
```

- Since the various sectioning commands are numbered automatically, the chapter, section, etc numbers may not be known at the time of writing the document and may change as more content is added.

- LaTeX has a cross-reference system, which allows you to label the various sectioning commands to refer to them at point in the document.

- To label a command, use \label{name} as in \chapter{Introduction}\label{chap:intro} or \section{My First document}\label{first}

- To reference the labeled section, use \ref{name} as in

```
\chapter{Introduction}\label{chap:intro}
\section{My First document}\label{sec:first}
In section \ref{sec:first} of Chapter \ref{chap:intro}, we wrote our first LaTeX document
```

- The cross-reference commands \label{name} and \ref{name} can also be used for other content such as tables, figures and equations.

- To get the cross-referencing to show up correctly, you need to compile your document i.e. run latex filename or pdflatex filename two times.

- The first time, the compiler stores the labels with the right number to be used for referencing.

- The second time, it replaces \ref{name} with the right number.

- The name that you use in the label command must be unique else the compiler will complain that there are multiply defined references.

- All auto-numbered headings get entered in the Table of Contents (ToC) automatically.

- Just add the command `\tableofcontents` at the point where you want it printed (usually after the title page).

- Entries for the ToC are recorded each time you process your document, and reproduced the next time you process it, so you need to re-run LaTeX one extra time to ensure that all ToC pagenumber references are correctly calculated.

- The commands `\listoffigures` and `\listoftables` work in exactly the same way as `\tableofcontents` to automatically list all your tables and figures, usually created after the TOC.

- The `\tableofcontents` commands normally shows only numbered section headings.

- To add extra entries, use the `\addcontentsline` command

```
\subsection*{Preface}
\addcontentsline{toc}{subsection}{Preface}
```

- This will format an unnumbered ToC entry for "Preface" in the "subsection" style.

- To change the title of the TOC, you have to use this command `\renewcommand{\contentsname}{New table of contents title}` in your document preamble.

- The default ToC will list headings of level 3 and above. Use the `\setcounter` command to change this depth. For e.g. `\setcounter{tocdepth}{4}`.

```latex
\documentclass[12pt,fleqn]{article}

\title{Simple \LaTeX{} Document}
\author{Alex Pacheco}
\date{\today}

\begin{document}
\maketitle
\tableofcontents

\begin{abstract}
  In this article we discuss how to create simple \LaTeX{} documents. Topics include structuring a document, list environment,
        inserting equations and figures, creating tables and more.
\end{abstract}

\section{Section 1}
\subsection{Section 1.1}
\subsection{Section 1.2}
\subsection*{Section nonumber}
\addcontentsline{toc}{subsection}{Section nonumber}
  blah blah blah

\section{Section 2}
\subsection{Section 2.1}
\subsubsection{Section 2.1.1}
\subsubsection{Section 2.1.2}
\paragraph{Paragraph} no numbering for this paragraph
\subparagraph{Subparagraph} no numbering for this subparagraph
\subsection{Section 2.2}
  blah blah blah

\appendix
\section{Appendix A}
\subsection{Appendix A.1}
  blah blah blah

\section{Appendix B}
\subsection{Appendix B.1}
\subsubsection{Appendix B.1.1}
  blah blah blah

\end{document}
```

Simple LaTeX Document

Alex Pacheco

October 25, 2013

Contents

Abstract

In this article we discuss how to create simple LaTeX documents. Topics include structuring a document, list environment, inserting equations and figures, creating tables and more.

1 Section 1

1.1 Section 1.1

1.2 Section 1.2

Section nonumber

blah blah blah

2 Section 2

2.1 Section 2.1

2.1.1 Section 2.1.1

2.1.2 Section 2.1.2

Paragraph no numbering for this paragraph

    Subparagraph no numbering for this subparagraph

2.2 Section 2.2

blah blah blah

A Appendix A

A.1 Appendix A.1

blah blah blah

B Appendix B

B.1 Appendix B.1

B.1.1 Appendix B.1.1

blah blah blah

- In LaTeX, the document type determines its overall general properties, such as layout and sectioning.
- However, it is possible to change the way certain commands work by invoking specific packages which may define new commands to add features that are not part of standard LaTeX.
- A LaTeX packages is nothing more than a set of LaTeX or TeX commands stored in a file with an extension .sty.
- To use a package, add `\usepackage[options]{packagename}` in the preamble of the document. The `[options]` is optional and some packages do not provide options at all.
- There are hundreds of useful packages and listing them all is beyond the scope of this tutorial.
- Some of the most commonly used packages are:

  amsmath   contains the advanced math extensions for LaTeX

  graphicx   manage external pictures.

  color   adds support for colored text.

  geometry   easy management of document margins and the document page size.

  inputenc   choose the encoding of the input text.

  babel   provides the internationalization of LaTeX. It has to be loaded in any document, and you have to give as an option the main language you are going to use in the document. e.x. `\usepackage[english]{babel}`

  hyperref   It gives LaTeX the possibility to manage links within the document or to any URL when you compile in PDF.

  cite   assists in citation management.

  natbib   gives additional citation options and styles.

- As your work grows, your LaTeX file can become unwieldy and confusing, especially if you are writing a long article with substantial, discrete sections, or a full-length book.

- In such cases it is good practice to split your work into several files.

- For example, if you are writing a book, it makes a lot of sense to write each chapter in a separate .tex file.

- LaTeX makes this very easy thanks to two commands:

  \input{filename}

  and

  \include{filename}

- Both these commands process the contents of filename.tex.

- When the compiler processes your base file (the file that contains these statements) and reaches the command \input or \include, it reads filename.tex and processes its content in accordance with the formatting commands specified in the base file.

- This way you can put all the formatting options in your base file and then \input or \include the files which contain the actual content of your work.

● There are some differences between these two commands:

❶ You cannot nest \include statements within a file added via \include, whereas \input, on the other hand, allows you to call files which themselves call other files, ad infinitum (well, nearly!).

You can, however, \include a file which contains one or more \input commands.

❷ \include will force a page break (which makes it ideal for a book's chapters), whereas the \input command does not.

```latex
\documentclass{article}
\begin{document}
\input{Section_1}
\input{Section_2}
\input{Section_3}
\input{Section_4}
\input{Section_5}
\end{document}
```

● The \includeonly{filename1,filename2} allows you to compile your document by including only the files listed in the curly braces.

```latex
\documentclass{book}
\includeonly{Chapter_1,Chapter_4} % compile just chapters 1 and 4, space characters not permitted
\begin{document}
\include{Chapter_1}              % omit the '.tex' extension
\include{Chapter_2}
\include{Chapter_3}
\include{Chapter_4}
\end{document}
```

Typesetting
  Foreign Letters & Accents
  Mathematics
  Math Symbols
  Power and Indices
  Fractions and Binomials
  Sums, Products, Integrals and Roots
  Matrices
  Creating Lists
  Creating Tables
  Table Environment and Captions
  Inserting Figures
  Floating Tables and Figures

- Special letters that exist in European languages can be generated with TeX.

{\oe}	œ	{\o}	ø	{\ae}	æ	{\l}	ł	{\aa}	å	{\ss}	ß
{\OE}	Œ	{\O}	Ø	{\AE}	Æ	{\L}	Ł	{\AA}	Å	{\SS}	SS

- In non-English languages, there is a multiplicity of diacritic marks or accents, most of which can be printed with TeX.

- The examples below are shown for the letter "o" but can be used with any letter.

\'{o}	ò	\'{o}	ó	\^{o}	ô	\"{o}	ö	\~{o}	õ
\={o}	ō	\.{o}	ȯ	\u{o}	ŏ	\v{o}	ǒ	\H{o}	ő
\t{oo}	o͡o	\c{o}	ǫ	\d{o}	ọ	\b{o}	o̲	\r{o}	ů

- When using these accents with the letters i and j, the dot must be removed using the commands \i and \j respectively to yield ı and ȷ.

- \u{\i} and \H{\j} now yield ĭ and j̋ instead of ĭ and j̋.

- One of the greatest motivating forces for Donald Knuth when he began developing the original TeX system was to create something that allowed simple construction of mathematical formulas, while looking professional when printed.

- Typesetting mathematics is one of LaTeX's greatest strengths.

- If your document requires only a few simple mathematical formulas, plain LaTeX has most of the tools that you will need.

- If you are writing a scientific document that contains numerous complicated formulas, the amsmath package introduces several new commands that are more powerful and flexible than the ones provided by LaTeX.

- The mathtools package fixes some amsmath quirks and adds some useful settings, symbols, and environments to amsmath.

- To use either package, include \usepackage{amsmath} or \usepackage{mathtools} in the preamble of the document.

- The mathtools package loads the amsmath package and hence there is no need to add \usepackage{amsmath} in the preamble if mathtools is used.

- LaTeX needs to know beforehand that the subsequent text does indeed contain mathematical elements.

- This is because LaTeX typesets maths notation differently from normal text. Therefore, special environments have been declared for this purpose.

- They can be distinguished into two categories depending on how they are presented:

  text   text formulas are displayed inline, that is, within the body of text where it is declared, for example, I can say that $a + a = 2a$ within this sentence.

  displayed   displayed formulas are separate from the main text.

- As maths require special environments, there are naturally the appropriate environment names you can use in the standard way.

- Unlike most other environments, however, there are some handy shorthands to declaring your formulas.

Type	Inline	Displayed	Displayed & automatically numbered
Environment	math	displaymath	equation
Requires			amsmath
LaTeX shorthand	\(... \)	\[... \]	
TeX shorthand	$ ... $	$$ ... $$	
Comment			starred version suppresses numbering

- Using the $$...$$ should be avoided, as it may cause problems, particularly with the AMS-LaTeX macros.

- Mathematics has many symbols!

- One of the most difficult aspects of learning LaTeX is remembering how to produce symbols.

- The following set of symbols can be accessed directly from the keyboard

  `+ - = ! / ( ) [ ] < > | ' :`

- Beyond those listed above, distinct commands must be issued in order to display the desired symbols.

- There are a great deal of examples such as Greek letters, set and relations symbols, arrows, binary operators, etc.

```
$\ forall x \in X, \quad \exists y \leq \epsilon$
```

$$\forall x \in X, \quad \exists y \leq \epsilon$$

- Fortunately, there's a tool that can greatly simplify the search for the command for a specific symbol.

  - ◆ Detexify: applet for looking up LaTeX symbols by drawing them
    `http://detexify.kirelabs.org/classify.html`

  - ◆ The Comprehensive LaTeX Symbol List
    `http://www.ctan.org/tex-archive/info/symbols/comprehensive`

- Mathematical formulas often contains exponents (power) and indices, characters that are either raised or lowered relative to the main line of the formula.

- Superscripts and subscripts are typographically the same things as exponents and indices respectively.

- The character command caret (^) set the next character as an exponent (superscript).

- The character command underscore (_) set the next character as an index (subscript).

- If the exponent or index contains more than one character, the group of characters must be enclosed in braces { }.

```
\begin{math}x^2, a_n, x^{10}, b_{i,j}, x^n_i\end{math}
```

$$x^2, a_n, x^{10}, b_{i,j}, x_i^n$$

- When exponents and indices occur together, their order is unimportant i.e. `x^n_i` and `x_i^n` will produce the same result as above.

- Multiple raisings and lowerings are generated by applying ^ and _ to the exponents and indices.

```
\begin{displaymath}
x^{y^2}, x^{y_1}, A^{x_i^2}_{j^{2n}_{n,m}}
\end{displaymath}
```

$$x^{y^2}, x^{y_1}, A^{x_i^2}_{j^{2n}_{n,m}}$$

- The raising and lowering commands ^ and _ are only permitted in math mode.

- By convention, all text in math mode is in italics or slanted text.

- If you need to write normal text with superscripts or subscripts, you need to use special commands to typeset the fonts correctly

```
The HPC Training on LaTeX will be held on Oct. $30^{th}$, 2013.\\
A better way to write this is to set the superscript th in roman font using Oct. $30^{\mathrm{th}}$
```

The HPC Training on LaTeX will be held on Oct. 30th, 2013.

A better way to write this is to set the superscript th in roman font using Oct. 30th

- Other available font typesets in math modes are

```
\mathrm{Roman} & \mathsf{sanserif} \\
\mathnormal{normal} & \mathtt{typewriter} \\
\mathit{italic} & \mathbf{boldface} \\
\mathcal{CAL} & \\
```

Roman	sanserif
$normal$	typewriter
$italic$	**boldface**
$\mathcal{CAL}$	

- A fraction is created using the \frac{numerator}{denominator} command.

- The binomial function can be written using the \binom command.

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

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

- Another way to write fractions is using the \over command while binomials can also be written with the \choose command,

```
\[ {n! \over k!(n-k)!} = {n \choose k} \]
```

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

- For relatively simple fractions, it may be more aesthetically pleasing to use powers and indices,

```
\[ {}^3/_7 \]
```

$${}^3/_7$$

- You can embed fractions within fractions,

```
\[ \frac{\frac{1}{x}+\frac{1}{y}}{y-z} \]
```

$$\frac{\frac{1}{x}+\frac{1}{y}}{y-z}$$

- Inline fractions can be typeset using the \ displaystyle {math command} or \dfrac{numerator}{denominator} command.

- Similarly, inline binomials can be typeset using the \dbinom{numerator}{denominator} command.

```
For example: $\frac{n!}{k!(n-k)!} = \binom{n}{k}$ looks crummy but $\dfrac{n!}{k!(n-k)!} = \dbinom{n}{k}$ or
        $\displaystyle{\frac{n!}{k!(n-k)!} = \binom{n}{k}}$ looks pleasing.
```

For example: $\frac{n!}{k!(n-k)!} = \binom{n}{k}$ looks crummy but $\dfrac{n!}{k!(n-k)!} = \dbinom{n}{k}$ or $\dfrac{n!}{k!(n-k)!} = \dbinom{n}{k}$ looks pleasing.

- Alternatively you can also use \tfrac, \tbinom or \ textstyle {math command} commands.

- Continued fractions should be written using \cfrac command,

```
x = a_0 + \cfrac{1}{a_1
        + \cfrac{1}{a_2
        + \cfrac{1}{a_3 + \cfrac{1}{a_4} } } }
```

$$x = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cfrac{1}{a_4}}}}$$

- Summation, Product and Integral signs are made with the commands \sum, \prod and \int respectively.

- Sums, products and integrals very often occur with upper and lower limits.

- These are printed using the power and index commands ^ and _ respectively.

```
2\sum^{i=1}_{n}\int^{b}_{a}f_i(x)g_i(x)dx \\
P^m_n = \prod^{m-1}_{i=0}(n-i)
```

$$2 \sum_{n}^{i=1} \int_{a}^{b} f_i(x) g_i(x) dx$$

$$P_n^m = \prod_{i=0}^{m-1} (n-i)$$

- Roots are printed using the command \sqrt[n]{arg} where n is the order. Default is $n = 2$ which can be omitted.

```
\[\sqrt[3]{8} = 2\quad\sqrt{4}=2\]
```

$$\sqrt[3]{8} = 2 \quad \sqrt{4} = 2$$

- Roots can be nested,

```
\[\sqrt[3]{-q + \sqrt{q^2 + p^3}}\]
```

$$\sqrt[3]{-q + \sqrt{q^2 + p^3}}$$

Greek letters are commonly used in mathematics, and they are very easy to type in math mode.

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

Special Symbols

```
$\dag$ & $\S$ & $\copyright$ & $\ddag$ & $\P$ & $\pounds$ \\
```

†	§	©	‡	¶	£

Miscellaneous Symbols

```
$\ partial$ & $\imath$ & $\Re$ & $\nabla$ & $\aleph$ \\
$\eth$    & $\jmath$ & $\Im$ & $\Box$ & $\beth$ \\
$\hbar$   & $\ell$  & $\wp$ & $\infty$ & $\gimel$ \\
$\angle$    & $\prime$ & $\forall$ & $\exists$ & $\nexists$ \\
$\measuredangle$ & $\sphericalangle$ & $\emptyset$ & $\eth$ & $\mho$ \\
```

∂	ι	ℜ	∇	ℵ
ð	ȷ	ℑ	□	ℶ
ℏ	ℓ	℘	∞	ℷ
∠	′	∀	∃	∄
∡	◁	∅	ð	℧

Binary Operation Symbols

```
$\pm$ & $\mp$ & $\times$ & $\cdot$ & $\oplus$ & $\ominus$ \\
$\otimes$ & $\odot$ & $\ast$ & $\star$ & $\dagger$ & $\ddagger$ \\
```

±	∓	×	·	⊕	⊖
⊗	⊙	∗	⋆	†	‡

Function Names

```
$\sin$ & $\arcsin$ & $\sinh$ & $\sec$ \\
$\cos$ & $\arccos$ & $\cosh$ & $\csc$ \\
$\tan$ & $\arctan$ & $\tanh$ & \\
$\cot$ &          & $\coth$ & \\
$\arg$ & $\deg$    & $\det$ & $\inf$ \\
$\exp$ & $\ln$     & $\log$ & $\lim$ \\
$\max$ & $\min$    & &\\
```

sin	arcsin	sinh	sec
cos	arccos	cosh	csc
tan	arctan	tanh	
cot		coth	
arg	deg	det	inf
exp	ln	log	lim
max	min		

Delimiters

```
$|$       & $\|$       & $\lfloor$   & $\rfloor$ & $\lceil$  & $\rceil$ \\
$\{$      & $\}$       & $($         & $)$       & $\langle$ & $\rangle$ \\
```

|	||	⌊	⌋	⌈	⌉
{	}	(	)	⟨	⟩

Arrows

```
$\leftarrow$       & $\Leftarrow$       & $\rightarrow$   & $\Rightarrow$ \\
$\uparrow$         & $\Uparrow$         & $\downarrow$    & $\Downarrow$ \\
$\leftrightarrow$  & $\Leftrightarrow$  & $\updownarrow$  & $\Updownarrow$ \\
```

←	⇐	→	⇒
↑	⇑	↓	⇓
↔	⇔	↕	⇕
↦	⇝	⊸	

Mathematical Symbols

```
$\sum$   & $\prod$  & $\coprod$  & $\int$ & $\oint$ \\
$\ldots$ & $\cdots$ & $\vdots$   & $\ddots$ & \\
```

∑	∏	∐	∫	∮
…	⋯	⋮	⋱	

Relational Symbols

```
$<$ & $>$ & $\not<$ & $\not>$ \\
$\le$ & $\ge$ & $\not\le$ & $\not\ge$ \\
$\ll$ & $\gg$ & $\lll$ & $\ggg$ \\
$\subset$ & $\supset$ & $\not\subset$ & $\not\supset$ \\
$\subseteq$ & $\supseteq$ & $\not\subseteq$ & $\not\supseteq$ \\
$\in$ & $\ni$ & $\not\in$ & $\not\ni$ \\
$\neq$ & $\sim$ & $\not\sim$ & $\doteq$ \\
$\approx$ & $\cong$ & $\not\approx$ & $\not\cong$ \\
$\equiv$ & $\asymp$ & $\not\equiv$ & $\not\asymp$ \\
$\propto$ & $\simeq$ & $\not\simeq$ & $\perp$ \\
$\parallel$ & $\mid$ & $\nparallel$ & $\nmid$ \\
```

Math Accents

```
$\hat{a}$ & $\check{a}$ & $\dot{a}$ & $\breve{a}$ & $\acute{a}$ & $\ddot{a}$ \\
$\grave{a}$ & $\tilde{a}$ & $\mathring{a}$ & $\bar{a}$ & $\vec{a}$ \\
```

$\hat{a}$	$\check{a}$	$\dot{a}$	$\breve{a}$	$\acute{a}$	$\ddot{a}$
$\grave{a}$	$\tilde{a}$	$\mathring{a}$	$\bar{a}$	$\vec{a}$	

Math Alphabet Commands

```
$\mathrm{Roman}$ & $\mathsf{sanserif}$ & $\mathnormal{normal}$ \\
$\mathtt{typewriter}$ & $\mathit{italic}$ & $\mathbf{boldface}$ \\
$\mathcal{CAL}$ \\
```

Roman	sanserif	$normal$
typewriter	$italic$	**boldface**
$\mathcal{CAL}$		

```
Note that in math mode as in the normal text i.e. anything other than numbers or
    commands is $slanted or italics\,and\quad there\qquad are no spaces in the text
    $. Thats the reason for the mathxx commands allows you to write text which is
    in non-math mode i.e. upright in the same font as your document.
```

Note that in math mode as in the normal text i.e. anything other than numbers or commands is *slantedoritalics and there arenospacesinthetext*. Thats the reason for the mathxx commands allows you to write text which is in non-math mode i.e. upright in the same font as your document.

If you need your text in math mode to be slanted and you do not use the mathit command, you can use the commands \, \quad \qquad to add space wherever you need to in your text.

- There are several environments available to write equations.

  1. equation

  2. eqnarray

  3. align

  4. gather

- The equation environment can be used to enter one equation at a time.

- The equations are automatically numbered in sequence. If you do not want equations to be numbered, use the starred version of the environment for e.g. equation∗

- You can also add \nonumber at the end of the equation to skip the numbering.

```
\begin{equation}
\imath\hbar\frac{\partial }{\partial t}\Psi = \mathcal{H}\Psi
\end{equation}
\begin{equation*}
\mathcal{H}\Psi = E\Psi
\end{equation*}
\begin{equation}
\Psi = \sum_i^n c_i\psi_i\nonumber
\end{equation}
```

$$i\hbar\frac{\partial}{\partial t}\Psi = \mathcal{H}\Psi \tag{1}$$

$$\mathcal{H}\Psi = E\Psi$$

$$\Psi = \sum_i^n c_i\psi_i$$

- To enter multi-line equations, use the eqnarray, align or gather environment.

- The individual lines of the equation are separated by \\.

- In eqnarray each line has the form

  `left formula` & `mid formula` & `right formula`

  where mid formula is centered and is single math character such as an assignment operators, =, >, etc.,

  the left formula is left justified and

  the right formula is right justified

- align is similar to eqnarray with the form `left formula` & `right formula` with the whole equation centered by default.

- eqnarray and align require the alignment marker & to align the equations while gather center aligns all equations.

- By default, all equations are centered. To left justify all equations add fleqn option to documentclass.

```
\begin{eqnarray}
\imath\hbar\frac{\ partial }{\ partial  t}|\Psi\rangle &=& \hat{\mathcal{H
    }}\mid\Psi\rangle\\
\mid\Psi_i\rangle &=& \sum_p|\psi_p\rangle c_{pi}
\end{eqnarray}
```

$$\imath\hbar\frac{\partial}{\partial t}|\Psi\rangle \quad = \quad \hat{\mathcal{H}} \mid \Psi\rangle \qquad (2)$$

$$\mid \Psi_i\rangle \quad = \quad \sum_p |\psi_p\rangle c_{pi} \qquad (3)$$

- The starred version of these environments i.e. eqnarray*, align* and gather* suppress printing of equation numbers.

- You can use the \setcounter{counter}{value} to set the equation number to value where counter=equation.

- You can also use the \addtocounter{counter}{value} to increment the equation by the value.

```
\addtocounter{equation}{4}
\begin{align}
\hat{\rho} &= \sum\mid\Psi\rangle\langle\Psi\mid\\
\imath\hbar\frac{\partial }{\partial  t}\hat{\rho} &= [\hat{\mathcal{H}},\hat{\rho}]
\end{align}
```

$$\hat{\rho} = \sum \mid \Psi \rangle \langle \Psi \mid \qquad (8)$$

$$\imath\hbar\frac{\partial}{\partial t}\hat{\rho} = [\hat{\mathcal{H}}, \hat{\rho}] \qquad (9)$$

```
\setcounter{equation}{2}
\begin{gather}
\hat{\rho} = \sum_{pq}|\psi_p\rangle c_{pi}c^{\ast}_{iq}\langle\psi_q| \\
  = \sum_{pq}|\psi_p\rangle{\boldsymbol\Gamma}^{i}_{pq}\langle\psi_q|
\end{gather}
```

$$\hat{\rho} = \sum_{pq} |\psi_p\rangle c_{pi} c^*_{iq} \langle\psi_q| \qquad (3)$$

$$= \sum_{pq} |\psi_p\rangle \boldsymbol{\Gamma}^i_{pq} \langle\psi_q| \qquad (4)$$

- LaTeX has a multline environment for writing long equations that need to split across multiple lines.

- The multline environment switches to math mode at the start and back to text mode at the end.

- The line break occurs when \\ is encountered.

- By default, the first line is left justified, the last line is right justified while all others are centered.

- You can use the commands \shoveleft{formula} and \shoveright{formula} to shift the lines to the left and right respectively.

```
\begin{multline}
|\Psi_{II}(t+\delta t)\rangle = \exp\left[-\frac{i}{\hbar} H_{II}\delta t\right] \\
\times\ left [|\Psi_{II}(t)\rangle + \frac{\delta t}{2\hbar}V_{I-II}|\Psi_I(t)\rangle\right]\\
 + \frac{\delta t}{2\hbar}V_{I-II}|\Psi_I(t+\delta t)\rangle
\end{multline}
```

$$
|\Psi_{II}(t + \delta t)\rangle = \exp\left[-\frac{i}{\hbar} H_{II}\delta t\right]
$$

$$
\times \left[|\Psi_{II}(t)\rangle + \frac{\delta t}{2\hbar} V_{I-II}|\Psi_I(t)\rangle\right]
$$

$$
+ \frac{\delta t}{2\hbar} V_{I-II}|\Psi_I(t + \delta t)\rangle \qquad (5)
$$

- Like multline, the split environment is meant for single equations that does not fit on a single line.

- Line breaks are forced with the \\ command.

- In the split environment, the alignment marker & is required to align the multi line equation.

- The split environment doesn't switch to math mode and needs to be used within an equation environment.

```latex
\begin{equation}\begin{split}
H_c &= \frac{1}{2n}\sum_{l=0}^n (-1)^l (k-1)^{p-2} \sum_{l_1+\dots+l_p=l} \prod_{i=1}^p \binom{n_i}{l_i} \\
  & \times[(k-l)-(k_i-l_i)]^{k_i-l_i} \times\left[(k-l)^2 - \sum_{j=1}^p (k_i-l_i)^2\right]
\end{split}\end{equation}
```

$$H_c = \frac{1}{2n} \sum_{l=0}^{n} (-1)^l (k-1)^{p-2} \sum_{l_1+\cdots+l_p=l} \prod_{i=1}^{p} \binom{n_i}{l_i}$$
$$\times [(k-l) - (k_i - l_i)]^{k_i - l_i} \times \left[(k-l)^2 - \sum_{j=1}^{p} (k_i - l_i)^2 \right] \tag{6}$$

- To number subordinate equations in a numbered equation environment, place the part of document containing them in a subequations environment:

```
\begin{subequations}
Maxwell's equations:
\begin{align}
B'&=-\nabla \times E,\\
E'&=\nabla \times B - 4\pi j,
\end{align}
\end{subequations}
```

Maxwell's equations:

$$B' = -\nabla \times E, \tag{7a}$$

$$E' = \nabla \times B - 4\pi j, \tag{7b}$$

- A basic matrix (or determinant) can be created using the matrix or array environment.

- Entries specified by row, with columns separated using an ampersand (&) and a new rows separated with a double backslash (\\).

- By default, all columns are center aligned in matrix environment but need to be specified in the array environment.

```
\begin{equation*}
\begin{matrix}
a & b & c \\
d & e & f \\
ghi & jkl & mno
\end{matrix}
=
\left (
\begin{array}{lcr}
a & b & c \\
d & e & f \\
ghi & jkl & mno
\end{array}
\right )
\end{equation*}
```

$$
\begin{matrix}
a & b & c \\
d & e & f \\
ghi & jkl & mno
\end{matrix}
=
\left (
\begin{array}{lcr}
a & b & c \\
d & e & f \\
ghi & jkl & mno
\end{array}
\right)
$$

- The matrix and array environments do not contain any delimiters.

- In general, matrices are enclosed in different delimiters such as ( ), { }, [ ], | |, || ||.

- You can add these delimiters explicitly as above or

- use the predefined LaTeX environments.

Environment name	Surrounding delimiter
matrix	
pmatrix	( )
bmatrix	[ ]
Bmatrix	{ }
vmatrix	| |
Vmatrix	|| ||

- If you need to align the columns differently i.e. either left or right aligned, use the starred version of these environment with column alignment as option to the environment command.

```
\begin{equation*}
\begin{Bmatrix*}[r]
a & b & c \\
ghi & jkl & mno
\end{Bmatrix*}
=
\begin{Vmatrix*}[l]
a & b & c \\
ghi & jkl & mno
\end{Vmatrix*}
\end{equation*}
```

$$\begin{Bmatrix*}[r] a & b & c \\ ghi & jkl & mno \end{Bmatrix*} = \begin{Vmatrix*}[l] a & b & c \\ ghi & jkl & mno \end{Vmatrix*}$$

- **amsmath** has a case environment to write piecewise functions,

```
\[ u(x) =
  \begin{cases}
  \exp{x} & \text{if } x \geq 0 \\
  1       & \text{if } x < 0
  \end{cases} \]
```

$$u(x) = \begin{cases} \exp x & \text{if } x \geq 0 \\ 1 & \text{if } x < 0 \end{cases}$$

- If the purpose is to make comments on particular parts of an equation, use the \overbrace and \underbrace commands,

```
\[  z = \overbrace{
   \underbrace{x}_\text{real} +
   \underbrace{iy}_\text{imaginary}
 }^\text{complex number} \]
```

$$z = \overbrace{\underbrace{x}_{\text{real}} + \underbrace{iy}_{\text{imaginary}}}^{\text{complex number}}$$

- The \xleftarrow and \xrightarrow commands produce arrows which extend to the length of the text. The optional argument [ ] contains the subscript while { } contains the superscript which can be empty.

```
\[  A \xleftarrow{\text{this way}} B \xrightarrow[\
    text{or that way}]{} C \]
```

$$A \xleftarrow{\text{this way}} B \xrightarrow[\text{or that way}]{} C$$

- Very often mathematical features will differ in size, in which case the delimiters surrounding the expression should vary accordingly. This can be done automatically using the \left, \right, and \middle commands.

```
\[ \left(\frac{x^2}{y^3}\right) \]
\[ P\left(A=2\middle|\frac{A^2}{B}>4\right) \]
```

$$\left(\frac{x^2}{y^3}\right)$$

$$P\left(A = 2\,\middle|\,\frac{A^2}{B} > 4\right)$$

- If a delimiter on only one side of an expression is required, then an invisible delimiter on the other side may be denoted using a period (.).

```
\[\left.\frac{x^3}{3}\right|_0^1\]
```

$$\left.\frac{x^3}{3}\right|_0^1$$

- LaTeX has several environments to create bulleted or numbered lists.

- itemize environment, creates a bulleted list.

- enumerate environment, creates a numbered list.

- description environment, creates a list with text instead of bullets and numbers.

```
\begin{itemize}
\item first level is a filled circle
  \begin{itemize}
  \item second level is a hyphen
    \begin{itemize}
    \item third level is an asterix
      \begin{itemize}
      \item fourth level is a period
      \end{itemize}
    \end{itemize}
  \end{itemize}
\end{itemize}
\begin{enumerate}
\item first level arabic numeral
  \begin{enumerate}
  \item second level is a letter in parenthesis
    \begin{enumerate}
    \item third level is a lowercase roman numeral
      \begin{enumerate}
      \item fourth level is an uppercase letter
      \end{enumerate}
    \end{enumerate}
  \end{enumerate}
\end{enumerate}
\begin{description}
\item[Header 1] Item 1
\item[Header 2] Item 2
\end{description}
```

Lists in LaTeX

1 itemize environment

\begin{itemize}
  \item item 1
  \item item 2
    \begin{itemize}
      \item subitem 1 of item 2
        \begin{itemize}
          \item subsubitem 1 of subitem 1
            \begin{itemize}
            \item subsubsubitem 1 of subsubitem 1
            \end{itemize}
        \end{itemize}
    \end{itemize}
\end{itemize}

- item 1

- item 2

    - subitem 1 of item 2

        * subsubitem 1 of subitem 1

            · subsubsubitem 1 of subsubitem 1

2 enumerate environment

\begin{enumerate}
  \item item 1
  \item item 2
    \begin{enumerate}
      \item subitem 1 of item 2
        \begin{enumerate}
          \item subsubitem 1 of subitem 1
            \begin{enumerate}
            \item subsubsubitem 1 of subsubitem 1
            \end{enumerate}
        \end{enumerate}
    \end{enumerate}
\end{enumerate}

1. item 1

2. item 2

    (a) subitem 1 of item 2

        i. subsubitem 1 of subitem 1

            A. subsubsubitem 1 of subsubitem 1

3 description environment

\begin{description}
  \item[Description]: The description environment is slightly   different .
  \item[item label]: You can specify the item label by passing it as an optional argument
  \item[] (although optional, it would look odd if you didn't include it !).
  \item[Use]: Ideal for a series of definitions , such as a glossary .
\end{description}

Description : The description environment is slightly different.

item label : You can specify the item label by passing it as an optional argument

(although optional, it would look odd if you didn't include it!).

Use : Ideal for a series of definitions, such as a glossary.

- Tables are created within the tabular environment.

- The arguments to the tabular environment is the positional alignment of all columns in the table.

- Every column entry needs to be separated by the alignment character &, a blank entry will be bounded by & except if it the last column entry.

- Each row needs to have \\ at the end signifying the end of row.

- To create a horizontal line between rows, use \hline. Note that \hline is not followed by \\.

- To create a horizontal line between columns m and n, use \cline{m-n}.

- To add border to each column add | to the position arguments of the tabular environment.

- The tabular environment is in text mode. To add column entries in math mode, your column entries should be within $...$.

  Recall all the math symbols in previous slides, they were created within a tabular environment.

```
\begin{tabular}{|l|c|r}
R1C1 & R1C2 & R1C3 \\
\hline
R2C1 & R2C2 & \\
\cline{2-3}
R3C1 & & R3C3 \\
\end{tabular}
```

R1C1	R1C2	R1C3
R2C1	R2C2	
R3C1		R3C3

- Creating fancy tables with one column spanning multiple columns is possible with LaTeX.

- use the command `\multicolumn{num}{col}{text}` where

  - num columns are merged into one,

  - col is the alignment of the column, either l, c, or r for left, center and right, and

  - text is the column entry.

- If you need one row to span multiple rows, you need to include `\usepackage{multirow}` in the preamble and

- use the command `\multirow{num}{row}{text}` where

  - num and text have the same meaning as in multicolumn, and

  - row is location of the text, by default centered using $*$.

  - You can use the `\raisebox{lift}{text}` command to reposition the text in both the multicolumn and multirow environments.

- To write long tables in landscape mode use the lscape package,

```
\usepackage{lscape} % this goes in the preamble
\begin{landscape} % to print a table in landscape mode
\begin{table} ... \end{table}
\end{landscape}
```

```latex
\begin{center}
\begin{tabular}{ |l|l|l| }
\hline
\multicolumn{3}{ |c| }{Team sheet} \\
\hline
Goalkeeper & GK & Paul Robinson \\ \hline
\multirow{4}{*}{Defenders} & LB & Lucus Radebe \\
 & DC & Michael Duberry \\
 & DC & Dominic Matteo \\
 & RB & Didier Domi \\ \hline
\multirow{3}{*}{Midfielders} & MC & David Batty \\
 & MC & Eirik Bakke \\
 & MC & Jody Morris \\ \hline
Forward & FW & Jamie McMaster \\ \hline
\multirow{2}{*}{\raisebox{-10pt}{Strikers}} & ST &
    Alan Smith \\
 & ST & Mark Viduka \\
\end{tabular}
\end{center}
```

Team sheet		
Goalkeeper	GK	Paul Robinson
Defenders	LB	Lucus Radebe
	DC	Michael Duberry
	DC	Dominic Matteo
	RB	Didier Domi
Midfielders	MC	David Batty
	MC	Eirik Bakke
	MC	Jody Morris
Forward	FW	Jamie McMaster
	ST	Alan Smith
Strikers	ST	Mark Viduka

Table Environment and Captions

- The tabular environment does not create a caption.

- If you need captions, the tabular environment needs to be within a table environment.

- The table environment is basically a floating table and LaTeX will place it at the earliest location without causing excessive space.

  i.e. if the table cannot fit at the location of content where entered, then LaTeX will put the table on the next page and add content that follows the table at the current location.

- Usage: `\begin{table}[loc] ... \end{table}`

- The caption can be either at the top i.e. before the table contents or at the bottom i.e. after the table contents.

- The caption is specified using `\caption[short title]{title}`

- The short title is the text that will appear in the list of contents (TOC for tables) if present else title will be used.

- To display a list of tables in the table of contents, add the command `\listoftables` at the location you want it to appear usually after the table of contents.

- The List of Tables (LoT) name can be changed by using the command `\renewcommand{\listtablenamename}{New List of Tables Title}`

- **You need to have a caption to cross reference tables in your document** for e.g. in Table **??** we show an example of a floating table.

```latex
\begin{center}
\begin{table}[ht]
\caption{First example of table with caption}\label{
    table1}
\begin{tabular}{ |l|l|l| }
\hline
\multicolumn{3}{ |c| }{Team sheet} \\
\hline
Goalkeeper & GK & Paul Robinson \\ \hline
\multirow{4}{*}{Defenders} & LB & Lucus Radebe \\
 & DC & Michael Duberry \\
 & DC & Dominic Matteo \\
 & RB & Didier Domi \\ \hline
\multirow{3}{*}{Midfielders} & MC & David Batty \\
 & MC & Eirik Bakke \\
 & MC & Jody Morris \\ \hline
Forward & FW & Jamie McMaster \\ \hline
\multirow{2}{*}{Strikers} & ST & Alan Smith \\
 & ST & Mark Viduka \\
\hline
\end{tabular}
\end{table}
\end{center}
```

Table 1: First example of table with caption

Team sheet		
Goalkeeper	GK	Paul Robinson
Defenders	LB	Lucus Radebe
	DC	Michael Duberry
	DC	Dominic Matteo
	RB	Didier Domi
Midfielders	MC	David Batty
	MC	Eirik Bakke
	MC	Jody Morris
Forward	FW	Jamie McMaster
Strikers	ST	Alan Smith
	ST	Mark Viduka

- LaTeX cannot manage pictures directly.
- We need to load the graphicx package in the preamble of the document.

  \usepackage[options]{graphicx}

- This package accepts as an optional argument the external driver to be used to manage pictures,

  dvips : (default) if you are compiling with latex to get a DVI and you want to see your document with a DVI or PS viewer.

  dvipdfm : if you are compiling with latex to get a DVI that you want to convert to PDF using dvipdfm.

  pdftex : (default if compiling with pdflatex), if you are compiling with pdftex to get a PDF that you will see with any PDF viewer.

- Supported Image Formats if compiling with

  latex : EPS (Encapsulated PostScript)

  pdflatex : JPG, PNG, PDF. You can use EPS if using the epstopdf package with compiler command `pdflatex -shell-escape file.tex`

- You can include images using the command

  `\includegraphics[attr1=val1, attr2=val2, ..., attrn=valn]{imagename}`

  where the attributes can be

  width=xx   Specify the preferred width of the imported image to xx cm (or in, pt).

  height=xx   Specify the preferred height of the imported image to xx.

  keepaspectratio   This can be set to either true or false. When true, it will scale the image according to both height and width, but will not distort the image, so that neither width nor height are exceeded.

  scale=xx   Scales the image by the desired scale factor. e.g, 0.5 to reduce by half, or 2 to double.

  angle=xx   This option can rotate the image by xx degrees (counter-clockwise).

  trim=l b r t   This option will crop the imported image by l from the left, b from the bottom, r from the right, and t from the top. Where l, b, r and t are lengths.

  clip   For the trim option to work, you must set clip=true.

- On Page 42, the simple latex document was inserted using the command
  `\includegraphics[height=\slidewidth,angle=90]{./asimple.ps}`

- If you need to include a caption, then you need use the figure environment,

```
\begin{figure}[loc]
\includegraphics[attr1=val1, attr2=val2, ..., attrn=valn]{imagename}
\caption{Figure Caption}
\end{figure}
```

- In WYSIWYG document processors, it is common to put tables and figures in the middle of the text.

- Professional documents, however, often make it a point to print tables and figures on a dedicated page so that they do not disrupt the flow.

- From the point of view of the source code, one has no idea on which page the current text is going to lie, so it is hardly possible to guess which page may be appropriate for our table.

- LaTeX can automate this task by abstracting objects such as tables and pictures, and decide for us where they might fit best. This abstraction is called a float.

- The table and figure environment create a table and figure as float respectively.

- For Example: \begin{table}[loc] ... \end{table} OR
\begin{figure}[loc] ... \end{figure}

- where loc is the position of the table (or figure) and can be one of the following

  h : Print the table (or figure) on the current page with content.

  t : Print the table (or figure) at the top of the page.

  b : Print the table (or figure) at the bottom of the page.

  p : Print the table (or figure) as a float on pages along with other floating tables and figures in sequence.

- Default location is tbp i.e. LaTeX will try to put the floating table or figure first at the top of the page, bottom of the page or on a separate page depending on the accompanying content.

User Customization
  User Defined Counters
  Printing counter values
  User Defined Commands
  Page Layout

- LaTeX manages a number of counters by giving them initial values at the start and changing these values when certain commands are called

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

- A user may create a new counter with the command `\newcounter{counter name}[in counter]`

  where "counter name" is the name of the newly established counter, and

  "in counter", an optional argument is the name of an already established counter.

- Whenever "in counter" is incremented using the commands `\stepcounter` or `\refstepcounter`, the "counter name" counter is reset to zero.

- Changing counter values:

  - `\setcounter{counter}{num}`: "counter" is assigned the integer value "num".
  - `\addtocounter{counter}{num}`: value of "counter" is increased by integer value (positive or negative) "num".
  - `\stepcounter{counter}`: value of "counter" is incremented by one and all its subcounters are reset to zero.
  - `\refstepcounter{counter}`: same effect as `\stepcounter{counter}`.
  - `\value{counter}`: Treat the value of the "counter" as a number mostly used with `\stepcounter` or `\addtocounter`.

- The numerical value in a counter can be printed with the commands

\arabic{counter}	Arabic number	$1, 2, 3, \cdots,$
\Roman{counter}	Uppercase Roman numeral	$I, II, III, IV, \cdots,$
\roman{counter}	Lowercase Roman numeral	$i, ii, iii, iv, \cdots,$
\alph{counter}	Lowercase letter	a,b,c,d,$\cdots,$
\Alph{counter}	Uppercase letter	A,B,C,D,$\cdots,$
\fnsymbol{counter}	footnote symbol	$*, \dagger, \ddagger, \S, \P, \|, **, \dagger\dagger, \ddagger\ddagger$

- For each counter, a command of the form \thecounter is also available such as \thepage.

- This type of command is initially set to \arabic{counter} but may be redefined to be composed of several counter commands.

  For e.g., the command \thesection in book and report is defined as \arabic{chapter}.\arabic{section} and \thesection will print say 7.1

- To change the counter commands, the user need to redefine the counter commands.

- New commands may be defined or redefined using the commands:

  `\newcommand{\com_name}[narg][opt]{def}`

  `\renewcommand{\com_name}[narg][opt]{def}`

- The first version defines a command `\com_name` which does not exist yet while

- the second version redefines an already existing command `\com_name`

- "narg" is a number between 1 and 9 specifying how many arguments the new or already command is to have,

- "opt" gives the default value for an optional argument that the new command may take, and

- "def" is the actual definition of the new command.

- Example:

  ◆ To redefine $x_1, \ldots, x_n$ which may occur at regular intervals

  ```
  \newcommand{\xvec}{$x_1,\ldots,x_n$}
  Redefine $x_1,\ldots,x_n$ as \xvec
  ```

  Redefine $x_1, \ldots, x_n$ as $x_1, \ldots, x_n$

  ◆ Redefine chapter, section and subsection numbering as I.i.a instead of 1.1.1

  ```
  \newcommand{\thechapter}{\Roman{chapter}}
  \newcommand{\thesection}{\thechapter.\roman{section}}
  \newcommand{\thesubsection}{\thesection.\alph{subsection}}
  ```

- Define commands with arguments, say $x_1, \ldots, x_n, y_1, \ldots, y_n$ and $z_1, \ldots, z_n$

```
\newcommand{\avec}[1]{\ensuremath{#1_1,\ldots,#1_n}}
Now $x_1,\ldots,x_n$, $y_1,\ldots,y_n$ and $z_1,\ldots,z_n$ can be written as \avec{x}, \avec{y} and \avec{z} in text mode or $\
    avec{x}$, $\avec{y}$ and $\avec{z}$ in math mode.
```

> Now $x_1, \ldots, x_n$, $y_1, \ldots, y_n$ and $z_1, \ldots, z_n$ can be written as $x_1, \ldots, x_n$, $y_1, \ldots, y_n$ and $z_1, \ldots, z_n$ in text mode or $x_1, \ldots, x_n, y_1, \ldots, y_n$ and $z_1, \ldots, z_n$ in math mode.

- The \ensuremath command permits using the newly defined command in both math and text mode.
- Define commands with optional arguments

```
\newcommand{\subvec}[3][x]{\ensuremath{#1_{#2},\ldots,#1_{#3}}}
\subvec{i}{j} prints $x_i,\ldots,x_j$ while \subvec[a]{1}{n} prints $a_1,\ldots,a_n$.
```

> $x_i, \ldots, x_j$ prints $x_i, \ldots, x_j$ while $a_1, \ldots, a_n$ prints $a_1, \ldots, a_n$.

- To change the first level of itemize from filled circle to asterix, and second level of enumerate to be similar to the section numbering i.e. 1.i, 2.iii, etc

```
\renewcommand{\labelitemi}{$\ast$}
\renewcommand{\theenumii}{\roman{enumii}}
\renewcommand{\labelenumii}{\theenumi.~\theenumii}
```

- The page layout, twoside, oneside, paper type can be set as options to `\documentclass`.

- One of the most versatile packages for page layout is the geometry package.

- The immediate advantage of this package is that it lets you customize the page size even with classes that do not support the options.

- Usage: `\usepackage[options]{geometry}`

- The geometry package has many pre-defined pages sizes such as a0paper, a1paper, · · · , a6paper, b0paper, b1paper, · · · , b6paper, letterpaper, legalpaper and executive paper.

- To explicitly change the paper dimensions using the geometry package, the paperwidth and paperheight options can be used.

  For example: `\usepackage[paperwidth=5.5in, paperheight=8.5in]{geometry}`

- Changing size manually: Use the `\setlength` command in the preamble to adjust the parameters to the appropriate dimensions.

  `\setlength{\paperwidth}{5.5in}` and `\setlength{\paperheight}{8.5in}`

- All pages in a LaTeX document have a header and footer.
- The header usually contains the document title or chapter title.
- The footer contains the page number.
- There are two commands to change to the headers in plain LaTeX.

  \pagestyle{"style"} will apply the specified style to the current and all subsequent pages, and

  \thispagestyle {"style"} will only affect the current page.
- The possible styles are:

  empty   Both header and footer are cleared.

  plain   Header is clear, but the footer contains the page number in the center.

  headings   Footer is blank, header displays information according to document class (e.g., section name) and page number top right.

  myheadings   Page number is top right, and it is possible to control the rest of the header.
- For example, if you put \thispagestyle {empty}, then no headers and footers (page number) for that page will not be printed.
- You can begin a newpage with the command \newpage.
- The commands \clearpage and \cleardoublepage 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, \cleardoublepage also makes the next page a right-hand (odd-numbered) page, producing a blank page if necessary.

Bibiliography
    Bibliography Management
    Embedded system
    Bibliographic Database
    BibTeX File
    natbib package

- For any academic/research writing, incorporating references into a document is an important task.

- Fortunately, LaTeX has a variety of features that make dealing with references much simpler, including built-in support for citing references.

- However, a much more powerful and flexible solution is achieved thanks to an auxiliary tool called BibTeX (which comes bundled as standard with LaTeX).

- BibTeX provides for the storage of all references in an external, flat-file database.

- This database can be referenced in any LaTeX document, and citations made to any record that is contained within the file.

- This is often more convenient than embedding them at the end of every document written; a centralized bibliography source can be linked to as many documents as desired (write once, read many!).

- bibliographies can be split over as many files as one wishes, so there can be a file containing sources concerning topic A (a.bib) and another concerning topic B (b.bib).

- When writing about topic AB, both of these files can be linked into the document (perhaps in addition to sources ab.bib specific to topic AB).

- LaTeX provides an environment called thebibliography that you have to use where you want the bibliography; that usually means at the very end of your document, just before the \end{document} command.

  Example

  ```
  \begin{thebibliography}{9}
  \bibitem{lamport94}
    Leslie Lamport,
    \emph{\LaTeX: A Document Preparation System}.
    Addison Wesley, Massachusetts,
    2nd Edition,
    1994.
  \end{thebibliography}
  ```

- thebibliography is a keyword that LaTeX recognizes as everything between the begin and end tags as being data for the bibliography.

- The mandatory argument is telling LaTeX how wide the item label will be when printed.

- In the above example, reference label with only one digit i.e. upto 9 references will be printed.

- To actually cite a given document, go to the point where you want the citation to appear, and use the following: \cite{cite_key}, where the cite_key is that of the bibitem you wish to cite.

- To cite the above example, type \cite{lamport94}.

- To cite a sequence of multiple references, use a single cite command with keys separated by commands.

  \cite{citation1,citation2,citation3}

- If you only want a reference to appear in the bibliography, but not where it is referenced in the main text, then the nocite command can be used,

  \nocite{citation1,citation2,citation3}

- A special version of the command, \nocite{*}, includes all entries from the database, whether they are referenced in the document or not.

- To compile a latex document and get the bibliographies listed correctly, you need to run latex filename or pdflatex filename two times.

- If you do not run latex or pdflatex two times after the bibtex command, your citation references will not show up in the text and you will see warnings such as ,

  "LaTeX Warning: There were undefined references."

- Instead of writing the bibitems at the end of each document, it would be convenient if one can create a database of such bibliographic entries which will then be available for all documents.

- BIBTeX is an auxiliary program to LaTeX that automatically constructs a bibliography by searching one or more databases.

- To this end, the LaTeX file must contain the command
  \bibliography{database1,database2,...} at the point where the bibliography is to appear.

- The argument database1, database2 is the root name of the database that are to be searched and has an extension .bib.

- The reference is again made with the \cite{key} or \nocite{key} command.

- The style of the bibliography can be selected using the command \bibliographystyle{style} where style can one of the following values,

  plain : The entries in the bibliography are ordered alphabetically, each is assigned a running number in square brackets.

  unsrt : The entries are ordered according to their first references by the cite and nocite commands.

  alpha : Same as plain but the markers are an abbreviation of the author's name plus year of publication.

  abbrv : Same as plain but bibliography listing is shortened by abbreviating first names, months and journal names.

- To compile a LaTeX document, you now need to run the following sequence of commands,
  1. latex filename (or pdflatex filename)
  2. bibtex filename
  3. latex filename (or pdflatex filename) $\times 2$

- The bibliography database is a plain text file with a .bib extension,

```
@article{greenwade93,
    author = ``George D. Greenwade'',
    title  = ``The {C}omprehensive {T}ex {A}rchive {N}etwork ({CTAN})'',
    year   = ``1993'',
    journal = ``TUGBoat'',
    volume = ``14'',
    number = ``3'',
    pages  = ``342--351''
}
@book{goossens93,
    author   = ``Michel Goossens and Frank Mittelbach and Alexander Samarin'',
    title    = ``The LaTeX Companion'',
    year     = ``1993'',
    publisher = ``Addison-Wesley'',
    address  = ``Reading, Massachusetts''
}
```

- Common types for entries in a BibTeX file are

  @article : An article from a magazine or a journal.

  @book : A published book.

  @proceedings : The proceedings of a conference. Can also use conference.

  @phdthesis : Ph.D. thesis.

  @manual : Technical manual.

  @inbook : A section of a book without its own title.

  @inproceedings : An article in a conference proceedings.

  @techreport : Technical report from educational, commercial or standardization institution.

  @unpublished : An unpublished article, book, thesis, etc.

- Using the standard LaTeX bibliography support, you will see that each reference is numbered and each citation corresponds to the numbers.

- The numeric style of citation is quite common in scientific writing.

- In other disciplines, the author-year style, e.g., (Roberts, 2003), such as Harvard is preferred.

- The natbib package is used to get such an output and it can supersede LaTeX's own citation commands.

- To use the natbib citation style, you need to add \usepackage[options]{natbib} to the document preamble.

- The options to the natbib package are

  round   : Parenthesis ( ) which is the default i.e. citation reference will be included within ( )

  square   : Square Brackets [ ]

  curly   : Curly Braces { }

  angle   : Angle brackets < >

  colon   : multiple citations are separated by semi-colons (default)

  comma   : multiple citations are separated by commas

  authoryear   : author year style citations (default)

  numbers   : numeric citations

  super   : superscripted numeric citations

  sort   : multiple citations are sorted into the order in which they appear in the references section

  sort&compress   : as sort, compressing multiple numeric citations where possible

The natbib package gives access to more citation commands as well as additional bibliography styles that are commonly used in scientific journals.

Natbib Commands	
Citation Command	Output
\cite{goossens93}	Goossens et al. (1993)
\citep{goossens93}	(Goossens et al., 1993)
\citet*{goossens93}	Goossens, Mittlebach, and Samarin (1993)
\citep*{goossens93}	(Goossens, Mittlebach, and Samarin, 1993)
\citeauthor{goossens93}	Goossens et al.
\citeauthor*{goossens93}	Goossens, Mittlebach, and Samarin
\citeyear{goossens93}	1993
\citeyearpar{goossens93}	(1993)
\citealt{goossens93}	Goossens et al. 1993
\citealp{goossens93}	Goossens et al., 1993
\citetext{priv.\ comm.}	(priv. comm.)

Natbib-compatible styles	
Style	Description
plainnat	natbib-compatible version of plain
abbrvnat	natbib-compatible version of abbrv
unsrtnat	natbib-compatible version of unsrt
apsrev	natbib-compatible style for Physical Review journals
rmpaps	natbib-compatible style for Review of Modern Physics journals
IEEEtranN	natbib-compatible style for IEEE publications
achemso	natbib-compatible style for American Chemical Society journals
rsc	natbib-compatible style for Royal Society of Chemistry journals

Wrap Up
References
Additional Help

[1] Helmut Kopka and Patrick W. Daly. *A Guide to LaTeX*. Addison-Wesley, 1999.

[2] Andrew Roberts. *Getting to Grips with LaTeX*. `http://www.andy-roberts.net/writing/latex`.

[3] *LaTeX*. Wikibooks.org. `http://en.wikibooks.org/wiki/LaTeX`.

[4] E. Krishnan, editor. *LaTeX Tutorial: A Primer*. Indian TeX Users Group, 2003. http://www.tug.org/twg/mactex/tutorials/ltxprimer-1.0.pdf.

```
This slide content was generated using
  \nocite{kopka,roberts,latex,ltxprimer}
  \ bibliographystyle {unsrt}
  \ bibliography {asv}
The BibTeX file, asv.bib is included in the download tar file
```

- LONI Moodle Course on LaTeX: `https://docs.loni.org/moodle`
- Contact Us
  - ✉ Ticket System: sys-help@loni.org
  - ☎ Help Desk: 225-578-0900
  - ☺ Instant Messenger (AIM, Yahoo Messenger, Google Talk)
    - ★ Add "lsuhpchelp"
- HPC staff does not support users with their LaTeX needs but feel to contact us regarding this tutorial or the LONI Moodle Course