

1 What is L^AT_EX?

The short and simple answer is that L^AT_EX is a typesetting program and is an extension of the original program T_EX written by *Donald Knuth*. At the beginning of 1980s, *Leslie Lamport* started work on a document preparation system called L^AT_EX based on the T_EX formatter. L^AT_EX is a system for typesetting documents. It was introduced in 1985. It is mainly used to produce mathematically oriented documents like thesis, books, articles, journals, LCD power points etc.

L^AT_EX's first widely available version, mysteriously numbered 2.09, appeared in 1985. L^AT_EX is now extremely popular in the scientific and academic communities, and it is used extensively in industry. Scientists send their papers electronically to colleagues around the world in the form of L^AT_EX input. Over the years, various non standard enhancements were made to L^AT_EX2.09 to overcome some of its limitations. L^AT_EX input that made use of these enhancements would not work properly at all sites. A new version of L^AT_EX was needed to keep a Tower of Babel from rising. The current version of L^AT_EX, with the some what less mysterious number 2 ϵ and was released in 1994. L^AT_EX2 ϵ contains an improved method for handling different styles of type, commands for including graphics and producing colors, and many other new features.

In L^AT_EX we prepare our document using a plain text editor, and the formatting is specified by commands embedded in our document. The appearance of our document is decided by L^AT_EX, but we need to specify it using some commands.

1.1 Merits of \LaTeX over Word Processors

1. Mathematical formulas can be produced quite easy.
2. Equation, table, figures, etc can be label so that cross referecing is automated.
3. \LaTeX enforces proper typesetting, especially for inexperienced writers. The math symbols are italicized, equations are numbered, and figures are properly captioned because the templates take care of it.
4. Automatic figure positioning is far more efficient in \LaTeX than in Microsoft Word.
5. Users need only to learn a few simple commands, which specify the logical structure of a document.
6. Complex structures such as footnotes, references, table of contents, and bibliographies can be generated easily.
7. Highly portable and FREE.

1.2 Demerits of \LaTeX

1. The design of a whole new layout is difficult and takes a lot of time.
2. \LaTeX is not a 'What You See Is What You Get '(WYSIWYG) system while most of the word processors are WYSIWYG systems.
3. Very hard to write unstructured and disorganized documents.
4. Although some parameters can be adjusted within a predefined document layout, the design of a whole new layout is difficult and takes a lot of time.
5. It lacks the obvious attraction of a real-time display of the formatted output.



1.3 Running of L^AT_EX

1. Create and save the file with .tex extension *eg.* maths.tex
2. The latex source document (the .tex file) is compiled by the latex program to generate a device independent (the .dvi file) output. From that you can generate postscript or PDF versions of the document.
3. Running of maths.tex (by the command `pdflatex maths.tex`) gives the output as pdf file.

1.4 L^AT_EX input File structure

When L^AT_EX process an input file, it requires us to follow a certain structure. Thus every input file must start with the command

```
\documentclass{class}
```

When all the set up work is done, you start the body of the text with the command

```
\begin{document}
```

Now you enter the text mixed with some useful L^AT_EX commands. At the end of the document you add the following command

```
\end{document}
```

which tells L^AT_EX the end of the file. Any thing which follows this command will be ignored by L^AT_EX.

2 Document structure

2.1 Preamble

The first command in any \LaTeX file normally determines the global processing format for the entire document. The syntax for this command :

`\documentclass[options]{class}` The possible values of class, of which one and only one may be given, are : article, book, report or letter. The options available allow various modifications to be made to the formatting, like selecting font size 10pt, 11pt, 12pt, specifying paper size letterpaper, legalpaper, executivepaper, page formats onecolumn, twocolumn etc. The standard \LaTeX class used for ordinary documents is the report class. The article class is generally used for shorter documents than the report class. The book class is for real books and letter class for formatting letters.

Preamble is the portion between `\documentclass` and `\begin{document}`. This can contain package loading command like `\usepackage{packagename}`. Any number of `\usepackage` command can be issued or alternatively you can give the package names as a comma separated list in a single `\usepackage` command. Preamble can also contain the header/footer style chosen, the command for which take the following form : `\pagestyle{style option}`.

A typical preamble of a \LaTeX document will look like :

```
\documentclass[a4paper,11pt,twocolumn]{article}
\usepackage{amsmath,times}
\pagestyle{headings}
\begin{document}
```

2.2 Page Numbering and Headings

The command `\pagestyle` controls page numbering and headings. It should always go between the `\documentclass{article}` and the `\begin{document}` commands. It can take the following forms :

1. `\pagestyle{plain}` is the default, which puts the page number at the center of the bottom of the page and provides no headings.



2. `\pagestyle{empty}` provides neither page numbers nor headings.
3. `\pagestyle{headings}` will provide page numbers and headings from any `\section`'s that you are using.
4. `\pagestyle{myheadings}` will provide page numbers and custom headings. These commands can also be applied to a single page using `\thispagestyle` instead of `\pagestyle`.

You can also define your own custom headers and footers with fancied text, boxes, graphic elements, etc.

2.3 Creating a Title Page

The title, author, and date of your document are information that various \LaTeX commands can make use of, if you provide it. It is a good habit to get into to provide this information in the preamble of your document. The commands are

1. `\title{your title here}`
2. `\author{your name here}`
3. `\date{current date}`

Given that you have provided this information in the preamble, you may or may not want a title heading. If you do, place a `\maketitle` command immediately after the `\begin{document}` command.

2.4 Preparing an Input

To get a desired output the following must be followed while typing the input.

- All commands should precede with a `\`
- **Sentences**

To denote the end of a word or sentence we leave one or more space. It doesn't matter how many spaces we type ; one is as good as five.



- **Paragraphs**

One or more **blank lines** denote the end of a paragraph.

- **Special Symbols**

To get the following 7 symbols in the output those symbols must be preceded by a backslash in front of the input file.

`$ & % # - { }`. That is `\$` will give \$ and `\%` will give %.

- **Logos of T_EX and L^AT_EX**

To get the logos of T_EX and L^AT_EX in the output, we type `\TeX` and `\LaTeX` in the input.

- **etc.**

The three dots ... to denote etc. can be obtained in the output document by typing `\ldots` in the input file.

- **Footnotes**

Footnotes are produced with a `\footnote` command having the text of the footnote as its argument.

- **Preventing Line Breaks**

- Typing `~` (a tilde character) produces an ordinary interword space at which T_EX will never break a line.
- The `\mbox` command tells T_EX to print its entire argument on the line. For example, by typing `\mbox{Mathematics}` T_EX will never split Mathematics across lines.

3 Modifying Text

We can use `\newline` or `\\` to generate a line break. A **blank line** will start a new paragraph. In a line, anything after a `%` sign is taken as a comment.

Now we will demonstrate different types of text.

| Command | Output |
|--------------------------------------|--------------------|
| <code>\textit{Mathematics}</code> | <i>Mathematics</i> |
| <code>\textbf{Mathematics}</code> | Mathematics |
| <code>\tiny{Mathematics}</code> | Mathematics |
| <code>\large{Mathematics}</code> | Mathematics |
| <code>\Large{Mathematics}</code> | Mathematics |
| <code>\underline{Mathematics}</code> | <u>Mathematics</u> |

4 Usepackages

Add-on features for \LaTeX are known as packages. Dozens of these are pre-installed with \LaTeX and can be used in your documents immediately. A *package* is a file or collection of files containing extra \LaTeX commands and programming which add new styling features or modify those already existing. Packages allow us to use extra commands without having to include tons and tons of code in the preamble of a document. Instead, we just add the line `\usepackage{package name}` and we have access all the commands, symbols, environments, etc. that are in the package.

4.1 AMS Math packages

The American Mathematical Society has produced several packages for use with \LaTeX . These packages allow much of the mathematical formatting we have introduced on these pages, as well as much, much more. The packages are included with the installation of MiKTeX. In order to be able to use the commands and symbols in these packages, they must be included in each document in which you require them. To include the packages, include the following in the preamble of your document:

```
\usepackage{amsmath}
\usepackage{amssymb}
\usepackage{amsthm}
```



4.2 geometry

To adjust the margins of pages, the geometry package comes in handy. To include this package include `\usepackage[a4paper]{geometry}` in the preamble of your document. The default page margins of the entire document can be altered with package options (the syntax between square brackets after `\usepackage`). We use this package most of the time to create a4 paper margins, which is done with the a4paper option. With this package it is also possible to change the margins of one particular page.

4.3 graphicx

Nothing special about the graphicx package, but it probably is the most important of all. This package introduces the `\includegraphics` command, which is needed for inserting figures.