

L^AT_EX Tutorial for UNIX Systems

You can either print this document or follow it on line.

About L^AT_EX

L^AT_EX (pronounced either “Lay-tech” or “Lah-tech”) is a portable document formatting system based on T_EX (pronounced “tech”), a typesetting language originally designed especially for math and science. It is also a programming language, which means you can create your own commands to simplify and customize it. T_EX and L^AT_EX use by default a font family called “Computer Modern,” which includes a variety of styles such as serif, sans serif, typewriter, and a particularly rich set of mathematical symbols.

Getting Started

You will first need to copy some files for use while going through this tutorial.

Create a new directory (e.g., `mkdir latex`), then `cd` to it and copy the following example files from the [L^AT_EX training page](#). (Go to the training page and scroll down to the Examples. Right click on the name of each file, choose “Save Link As...” .)

For exercises

`basic.tex`

`exart.tex`

`ex2.tex`

`ex3.tex`

Graphics info

`graphics.tex`

`graphics.pdf`

`exrotating.tex`

`exrotating.pdf`

Image files

`cat.eps`

`cat.pdf`

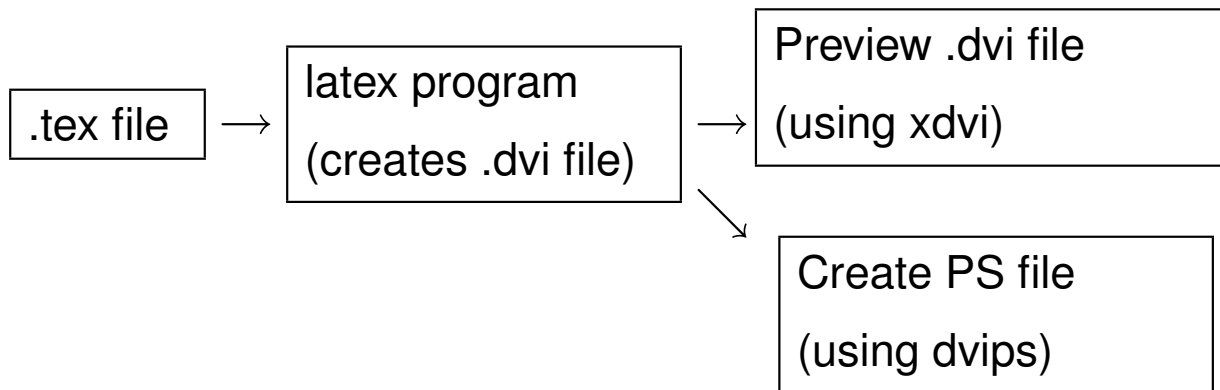
`smokeblk.eps`

`smokeblk.pdf`

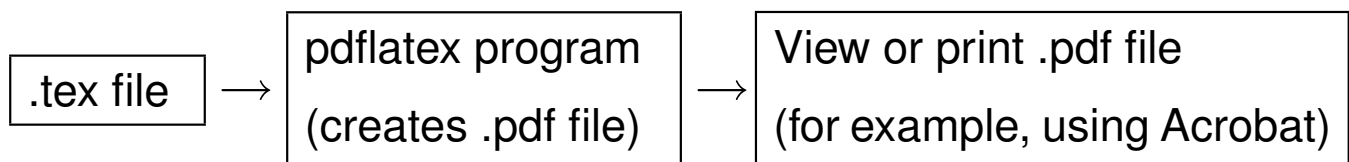
How L^AT_EX Works

To use L^AT_EX, you first create a file using a plain text editor (such as `emacs`, `vi` or `gedit`) and give it a name ending with `.tex`. In this file, you type both the text of your document and the commands to format it. Then there are two ways to process and print your `.tex` file:

1. The traditional way is to run the `latex` program, which creates a DVI (Device Independent) file. This file is in binary format and not viewed directly. You then run a previewing program for viewing on screen and/or the `dvips` program to create a PostScript file for viewing or for printing via Ghostview or GV.



2. Alternatively you can run the relatively recent `pdflatex` program to create a PDF file for viewing or printing.



The second method is more direct but not as quick or convenient for previewing as the first.

Try it!

Before getting into the details of the L^AT_EX language, you can get a feel for what it's like to process and view a L^AT_EX file by trying out both methods on the simplest of the files you copied, `basic.tex`. *

Try out method 1

1. Run L^AT_EX by issuing the UNIX command:

```
latex basic.tex    (you may omit the ".tex")
```

2. Preview the resulting DVI (device independent) file:

```
xdvi basic.dvi &  (you may omit the ".dvi")
```

3. If/when you want to print (probably not now), do so by running `dvips` and then using Ghostview or GV:

```
dvips basic -o     (output goes into the file basic.ps)
```

Open `basic.ps` with Ghostview (or GV) and either print from there or convert to PDF (File → Convert) and print using a program such as Acrobat Reader.

Try out method 2

1. Run PDFL^AT_EX by issuing the UNIX command:

```
pdflatex basic.tex    (you may omit the ".tex")
```

2. View the resulting file, `basic.pdf`, with Acrobat Reader:

```
acroread basic.pdf &
```

To print, select **Print...** from Acrobat's File menu. Before printing, be sure Page Scaling is set to "None" in the Print window.

* For more information on UNIX commands, see [Using LaTeX on UNIX Systems](#).

Structure of a L^AT_EX File

```
\documentclass [options] {article}
```

Preamble (for L^AT_EX commands only)

```
\begin{document}
```

Document text (text with embedded L^AT_EX commands)

```
\end{document}
```

The **Document class** determines the overall layout of the document. In addition to **article** class, which is a good all-purpose class, other commonly-used classes are:

report – for longer documents containing chapters

thesis – for writing an RPI thesis (see [Preparing a Thesis with L^AT_EX](#)).

book – for books

letter – for letters

slides – for making transparencies

Among other things, the classes provide heading commands, such as `\part`, `\chapter`, `\section`.

Document Class Options & Packages

A document class may be modified by using options:

```
\documentclass[options]{article}
```

Commonly-used options include:

11pt Prints document in 11pt type (default 10pt)

12pt Prints document in 12pt type

Example:

```
\documentclass[11pt]{article}
```

L^AT_EX **Packages** contain extra definitions that provide additional formatting features. To load a package, include in the preamble the command:

```
\usepackage{packagename}
```

Some commonly-used packages are:

setspace Provides easy way to change linespacing

graphicx Provides commands to include graphics files

fancyhdr Customizes headers and footers

rotating Provides rotations, especially for figures & tables

color Provides a way to use colors

L^AT_EX Basics

The backslash “\” is used to begin all L^AT_EX commands.

In the input file (.tex file), words are separated by one or more blank spaces, paragraphs are separated by one (or more) blank lines.

Commands are case-sensitive. Commands are all lowercase unless there’s a good reason to use uppercase. For example: `\Delta` → Δ
`\delta` → δ

Some commands take *arguments*, which are enclosed in braces:

`\textbf{this text will be bold}`

Certain characters have special meaning to L^AT_EX. The complete list is in [Text Formatting with L^AT_EX](#); the most common are listed below.

<u>Char</u>	<u>Input</u>	<u>Special T_EX meaning</u>
#	\#	Parameter in a macro
\$	\\$	Used to begin and end math mode
%	\%	Used for comments in the input file
&	\&	Tab mark, used in alignments
-	\-	Used in math mode for subscripts

Some L^AT_EX Vocabulary

Commands produce text or space:

`\hspace{2in}` and `\textit{some italic words}`

Declarations affect the following text:

`\large` prints the following text in a larger font.

Grouping `{ }` is often used to limit the scope of a declaration:

`{\large only this text is big}`

Environments receive special processing and are defined by

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

Example: `\begin{quote} ... \end{quote}`

Mandatory arguments are included in braces : `\hspace{2in}` needs the information provided by the argument to generate the space.

Optional arguments are enclosed in brackets `[]`:

`\documentclass[11pt]{article}` gives you 11-point type.

(The default is 10-point type.)

* indicates a variation on a command or environment.

`\\` indicates a line break

`*` indicates a line break where a page cannot be broken.

Exercise 1

1. Close `xdvi` or Acrobat Reader if you haven't already.
2. In your `latex` directory, use your editor to open `exart.tex` and note the following:
 - use of the heading commands
 - use of `\textit{..}`, `\textbf{..}`, `{\small..}`
 - how to get various dashes and quotes
3. Run either `latex` or `pdflatex` (note you don't need the `.tex` extension):

```
latex exart
```

```
pdflatex exart
```

4. Preview the result:

```
xdvi exart & (if you ran latex)
```

```
acroread exart.pdf & (if you ran pdflatex)
```

5. If you used `xdvi`, leave it open. If you used `acroread`, close the file but not the program. Return to your editor and make the following changes to the `exart.tex` file:

- add the option `[12pt]` to `\documentclass`
- add to the preamble the command:

```
\pagestyle{empty} % no pagenumbers
```
- change `\section` and `\subsection` headings to `\section*` and `\subsection*`

6. Save the changes, run `latex` again, and bring the `xdvi` window to the front. Or run `pdflatex` again, and in Acrobat, go to the File menu and in the list of files at the bottom, click on `exart.pdf` to reopen the file. Note the differences in the formatted result.

Responding to Error Messages

When L^AT_EX finds an error, it emits an error message ending with a “?” prompt, such as:

```
! Undefined control sequence.  
1.9 \section  
      {Introducing LaTeX}  
?
```

This means the `\section` command was misspelled, and the error occurred on line 9 of the input file.

Other common errors include unmatched braces or a special character (e.g., \$, #, %) in the text. You can respond with:

h for help

x for exit

press the **Return** key to ignore it, hope for the best.

If it stops with a * prompt, it often means you have forgotten `\end{document}`. Enter it at the prompt (and fix the file later).

If you mistyped the file name or for some other reason L^AT_EX cannot find a file, it will ask for another filename. If you don't want to enter a new filename, quit the program by typing Ctrl-d. Another handy “Emergency stop sequence” is Ctrl-c.

Exercise 2

1. Quit `xdvi` or Acrobat Reader.
2. Still in your `latex` directory, edit the file `ex2.tex`.
3. Note the following:
 - How to use the `center` environment
 - How to use the `quote` environment
 - How to make 3 kinds of lists
4. Run `latex` or `pdflatex` by issuing one of the commands:

```
latex ex2
```

```
pdflatex ex2
```

5. Preview the result:

```
xdvi ex2 &      (if you ran latex)
```

```
acroread ex2.pdf &      (if you ran pdflatex)
```

6. Return to your editor and add a section at the end illustrating the `itemize` environment.

HINT: See the comments at the end of the file.

Tables

To make a table, use the `tabular` environment. This environment requires an additional parameter that specifies the alignment of each column:

```
\begin{center}
\begin{tabular}{ccc} % 3 centered columns
Apples & Oranges & Pears \\
Bananas & Mangos & Melons
\end{tabular}
\end{center}
```

Apples	Oranges	Pears
Bananas	Mangos	Melons

-
- l Left-justified column entry
 - c Centered column entry
 - r Right-justified column entry
 - p Paragraph column entry
 - | Vertical rule column
 - || Double vertical rule column

The width of each column is determined automatically from the widest entry.

Inside the `tabular` environment:

- & (the tab character) moves to the next column
- \\ is used to end each line (except the last one)

Example: A Ruled Table

```

\begin{tabular}{|l|p{2.5in}|}
\multicolumn{2}{c}{Various Column Alignments}\
\hline
llll & 4 columns, all left justified
      (yes, those are ‘‘L’’s not ‘‘1’’s)\
lcr   & 3 columns with the first entry
      left justified, the second centered,
      and the third right justified.\
lp\{2.5in\} & 2 columns, the second is
              a paragraph 2.5 inches wide.\
\hline
\end{tabular}

```

Various Column Alignments

llll	4 columns, all left justified (yes, those are “L”s not “1”s)
lcr	3 columns with the first entry left justified, the second centered, and the third right justified.
lp{2.5in}	2 columns, the second is a paragraph 2.5 inches wide.

Mathematics

L^AT_EX has several modes for formatting equations. When in a math mode, L^AT_EX follows different rules:

- All letters are set in the math italic font.
- There are many commands to print special symbols (e.g., `\pi`). Most work only in math mode.
- All spaces in the input are ignored.
- new paragraphs are not allowed. (No blank lines!)

To use roman type or retain spaces put the text in an “mbox”:

`\mbox{this is normal text}`.

The simplest mode is **in-line math**. The formulas are about the same size as the text they’re in. To use it:

`$ math expression $`

The equation `$ax^2+bx+c = 0$` has 2 roots.

produces:

The equation $ax^2 + bx + c = 0$ has 2 roots.

Displayed Equations

There are four environments for formatting equations that are to be set off from surrounding text.

1. **Display Math** (for unnumbered equations)

It can be invoked using any of the following:

`$$... $$` (*plain T_EX notation, convenient but frowned on in L^AT_EX*)

`\[... \]` (*This and the following are approved L^AT_EX notation*)

`\begin{displaymath} ... \end{displaymath}`

2. **Equation Environment** (for numbered equations)

Just like `Display Math` except it numbers the equation. It is invoked with

`\begin{equation}... \end{equation}`

3. **Eqnarray Environment** (for multiline equations)[†]

Formats a series of equations, aligning them on the “=” or some other point of your choosing. It is invoked with

`\begin{eqnarray}... \end{eqnarray}`

4. **Array Environment** (for matrices, etc.)

Builds rectangular arrays of numbers, matrices, etc. It is invoked with

`\begin{array}... \end{array}`

NOTE: *The array environment must be enclosed in another math environment, such as displaymath or equation.*

[†] The AMS provides more environments for multiline equations. See page 18 of this tutorial.

Math Examples

1. Display Math (for unnumbered equations)

The quadratic equation $ax^2+bx+c=0$ has 2 roots: `\[x=\frac{-b \pm \sqrt{b^2-4ac}}{2a} \]`

produces:

The quadratic equation $ax^2 + bx + c = 0$ has 2 roots:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

2. Equation Environment (for numbered equations)

```
\begin{equation}
\frac{a^2 - b^2}{a + b} = a - b
\end{equation}
```

produces:

$$\frac{a^2 - b^2}{a + b} = a - b \tag{1}$$

3. Eqnarray Environment (for multiline equations)

This environment builds a 3-column array of equations. It numbers each line by default, but the command `\nonumber` suppresses the number. (The alternative environment `eqnarray*` does not number any lines.)

```
\begin{eqnarray}
(a+b)(a+b) & = & a^2+ab+ba+b^2 \nonumber \\
& = & a^2+2ab+b^2
\end{eqnarray}
```

produces:
$$\begin{aligned} (a+b)(a+b) &= a^2 + ab + ba + b^2 \\ &= a^2 + 2ab + b^2 \end{aligned} \tag{1}$$

4. Array Environment (for matrices, etc.)

This environment uses the same syntax as `tabular`. Note that it must be inside a math environment.

```
\[
\begin{array}{ccc}
x-\lambda & 1 & 0 \\
0 & x-\lambda & 1 \\
0 & 0 & x-\lambda
\end{array}
\]
```

produces:

$$\begin{array}{ccc} x - \lambda & 1 & 0 \\ 0 & x - \lambda & 1 \\ 0 & 0 & x - \lambda \end{array}$$

Exercise 3

1. Close xdvi or Acrobat Reader.
2. In your editor, close `exart.tex` and open `ex3.tex`.
3. Study the L^AT_EX commands used.
4. Run `latex` or `pdflatex` and view the result.
5. Return to your editor
6. At the end of the `ex3.tex` file, add the L^AT_EX commands to format the following two mathematical expressions:

$$\binom{n}{k-1} + \binom{n}{k} = \binom{n+1}{k}$$

$$\lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{k^2} = \frac{\pi^2}{6}$$

$$I_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

(Hint: To make the large parentheses, see [Text Formatting with L^AT_EX](#) on using the `\left` and `\right` commands.)

For the solution to this exercise, see [ex3-sol.tex](#).

If You're Looking for More...

Including Graphics in your Document

For an explanation and examples of including graphics files in your L^AT_EX document, read the file `graphics.pdf` and then look at the file that produced it, `graphics.tex`. You can also run `latex` or `pdflatex` on `graphics.tex` yourself and view the result. In addition, the pair of files `exrotating.pdf` and `exrotating.tex` provide more information, especially on including landscape figures and tables. (All these files are among the example files you copied from the [L^AT_EX training page](#).)

Preparing a Thesis

For information on doing a thesis with L^AT_EX, see the [LaTeX thesis web page](#). The Help Desk page offers easy access: under Quick Links, choose “prepare a thesis” and then choose “Preparing a thesis with LaTeX”. From the this page, you can also download the RPI thesis class and the template files.

Math Extensions from the AMS

Some mathematicians like to use AMSL^AT_EX, a collection of several packages: **amssymb** provides additional mathematical symbols; **amsmath** provides additional environments for building mathematical expressions.

For information on using AMSL^AT_EX, see *The Short Math Guide for L^AT_EX*, at: <ftp://ftp.ams.org/pub/tex/doc/amsmath/short-math-guide.pdf>