

# L<sup>A</sup>T<sub>E</sub>X Tutorial

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

## About L<sup>A</sup>T<sub>E</sub>X

L<sup>A</sup>T<sub>E</sub>X (pronounced either “Lay-tech” or “Lah-tech”) is a portable document formatting system based on T<sub>E</sub>X (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<sub>E</sub>X and L<sup>A</sup>T<sub>E</sub>X 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.

(Note: If you are using L<sup>A</sup>T<sub>E</sub>X in a public lab (e.g., VCC North), first log on using your RCS User ID and password, and when finished remember to log off.)

Create a new folder, then copy into it the following example files from the [L<sup>A</sup>T<sub>E</sub>X 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

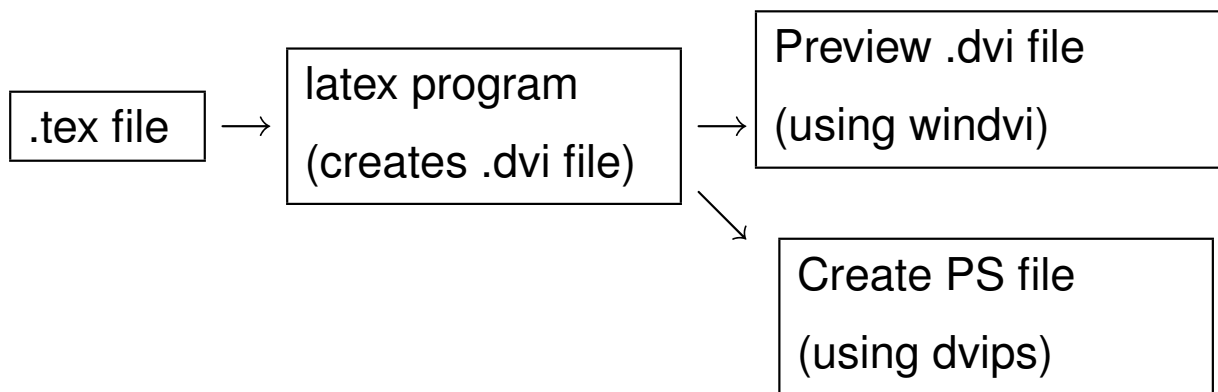
smokeblk.eps

smokeblk.pdf

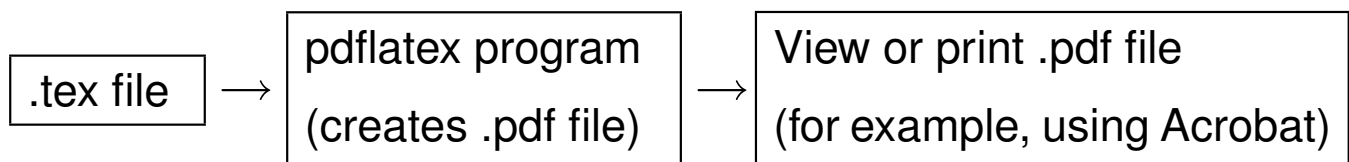
## How L<sup>A</sup>T<sub>E</sub>X Works

To use L<sup>A</sup>T<sub>E</sub>X, you first create a file using a plain text editor (such as WinShell or WinEdt on Windows) 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 GSView.



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<sup>A</sup>T<sub>E</sub>X language, you can get a feel for what it's like to process and view a L<sup>A</sup>T<sub>E</sub>X file by trying out both methods on the simplest of the files you copied, `basic.tex`.

Double-click on `basic.tex`. Your editor/shell should open the file. We'll assume that you are using WinShell and that your previewer is Windvi.

### Try out method 1

1. Run L<sup>A</sup>T<sub>E</sub>X by clicking on the L<sup>A</sup>T<sub>E</sub>X icon on the toolbar.
2. Preview the resulting DVI (device independent) file by clicking on the DVI icon on the toolbar. The Windvi previewer will open. You can zoom in or out to choose a viewing magnification.

Leave the DVI window open while you return to your editor to make changes; the window will refresh when you rerun L<sup>A</sup>T<sub>E</sub>X.

3. If/when you want to print (probably not now), do so by using GSView: From WinShell, click the DVIPS icon. After `dvips` runs, click the GSView icon to open the `.ps` file. From the File menu, select **Print....**

### Try out method 2

1. Run pdfL<sup>A</sup>T<sub>E</sub>X by clicking on the pdfL<sup>A</sup>T<sub>E</sub>X icon on the toolbar.
2. Click the PDFView icon on the toolbar to view the PDF output in Acrobat Reader. If you want to rerun L<sup>A</sup>T<sub>E</sub>X, you will have to close the file first (just close the file, not Acrobat Reader) because AR will not allow you to write on an open file. After rerunning L<sup>A</sup>T<sub>E</sub>X, open the file again to view your changes.
3. To print, select **Print...** from Acrobat's File menu. Before printing, be sure Page Scaling is set to "None" in the Print window.

## Structure of a L<sup>A</sup>T<sub>E</sub>X File

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

*Preamble* (for L<sup>A</sup>T<sub>E</sub>X commands only)

```
\begin{document}
```

*Document text* (text with embedded L<sup>A</sup>T<sub>E</sub>X 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<sup>A</sup>T<sub>E</sub>X](#)).

**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<sup>A</sup>T<sub>E</sub>X **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<sup>A</sup>T<sub>E</sub>X Basics

The backslash “\” is used to begin all L<sup>A</sup>T<sub>E</sub>X 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<sup>A</sup>T<sub>E</sub>X. The complete list is in [Text Formatting with L<sup>A</sup>T<sub>E</sub>X](#); the most common are listed below.

<u>Char</u>	<u>Input</u>	<u>Special T<sub>E</sub>X 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<sup>A</sup>T<sub>E</sub>X 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 Windvi and GSView or Acrobat if you haven't already.
2. In WinShell, close `basic.tex`, 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 `latex` (click the L<sup>A</sup>T<sub>E</sub>X icon) or `pdflatex` (click the PDF L<sup>A</sup>T<sub>E</sub>X icon).
4. Preview the result by clicking on the DVI icon (for `latex`) or the PDF icon (for `pdflatex`).
5. Leave Windvi open, or in Acrobat<sup>†</sup> click the “x” to close the file, and return to WinShell. 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. Run `latex` again, and bring Windvi to the front. Or run `pdflatex` again, and in Acrobat<sup>†</sup>, go to the File menu and under History, click on `exart.pdf` to reopen the file.

Note the differences in the formatted result.

---

<sup>†</sup> For a shortcut to closing and reopening a file in Acrobat, you can close the display with `Ctrl-W` and reload the file with `Alt-←` (left arrow). Alternatively, you can configure your editor to view PDF files with GSView instead of Acrobat. GSView can refresh the display.



## Responding to Error Messages

When L<sup>A</sup>T<sub>E</sub>X finds an error, the Command window will remain open, and you will see a message such as:

```
! Undefined control sequence.  
1.9 \secton  
      {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<sup>A</sup>T<sub>E</sub>X 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 "x". Another handy "Emergency stop sequence" is Ctrl-z.

**IMPORTANT:** *Do not simply close the Command window without responding!*

The window may disappear but L<sup>A</sup>T<sub>E</sub>X is still running, which will cause confusion when you run it again.

## Exercise 2

1. Quit Windvi or Acrobat.
2. In WinShell, close `exart.tex` and open `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 clicking on the appropriate icon.
5. Preview the result by clicking on the DVI or PDF icon.
6. Return to WinShell 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<sup>A</sup>T<sub>E</sub>X has several modes for formatting equations. When in a math mode, L<sup>A</sup>T<sub>E</sub>X 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<sub>E</sub>X notation, convenient but frowned on in L<sup>A</sup>T<sub>E</sub>X*)

`\[ ... \]` (*This and the following are approved L<sup>A</sup>T<sub>E</sub>X 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)<sup>†</sup>

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.*

---

<sup>†</sup> 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 Windvi or Acrobat.
2. In WinShell, close `exart.tex` and open `ex3.tex`.
3. Study the L<sup>A</sup>T<sub>E</sub>X commands used.
4. Run `latex` or `pdflatex` and view the result.
5. Return to Winshell
6. At the end of the `ex3.tex` file, add the L<sup>A</sup>T<sub>E</sub>X 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<sup>A</sup>T<sub>E</sub>X](#) 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<sup>A</sup>T<sub>E</sub>X 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<sup>A</sup>T<sub>E</sub>X training page](#).)

### Preparing a Thesis

For information on doing a thesis with L<sup>A</sup>T<sub>E</sub>X, 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<sup>A</sup>T<sub>E</sub>X, a collection of several packages: **amssymb** provides additional mathematical symbols; **amsmath** provides additional environments for building mathematical expressions.

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

## Installing L<sup>A</sup>T<sub>E</sub>X on your PC

The TeXLive distribution of Windows is available for installation at Rensselaer in RCS space.

You do not need to download the files to your PC; you can install directly from RCS. But you first need to connect to RCS space via sambasrv so that a drive is mapped to the appropriate RCS directory. (Finding the files via the web will not work; your PC must have a drive mapped to `\\sambasrv\swinstall`.)

For instructions on mapping the network drive, see:

[www.rpi.edu/dept/arc/software/latex-doc/Readme-RPI.html](http://www.rpi.edu/dept/arc/software/latex-doc/Readme-RPI.html)

When you are ready to install, print the detailed instructions

[www.rpi.edu/dept/arc/software/latex-doc/install-windows.pdf](http://www.rpi.edu/dept/arc/software/latex-doc/install-windows.pdf)

and follow them carefully to install the TeXLive system and its support programs to your PC.