

L^AT_EX for beginners

Stéphane Péchard

September 22, 2009

Outline

1 Introduction (or Why using \LaTeX ?)

2 First document

- Document structure
- Form-content separation
- Basic commands

3 Mathematics and code

4 Playing with content

5 Going further

6 Resources

Some terminology

$\text{T}_{\text{E}}\text{X} \dots$

- is a low-level markup and programming language
- was created by Donald Knuth between 1977 and 1989!
- is very stable and powerful but time-consuming and difficult to learn

$\text{\LaTeX} \dots$

- is a package of macros based on $\text{T}_{\text{E}}\text{X}$ to make a document preparation system
- created by Leslie Lamport and still actively maintained
- easier to code and extendable (classes, packages, styles)

but $\text{T}_{\text{E}}\text{X} \neq \text{\LaTeX}$

Pros and cons of a non-WYSIWYG approach

Pros

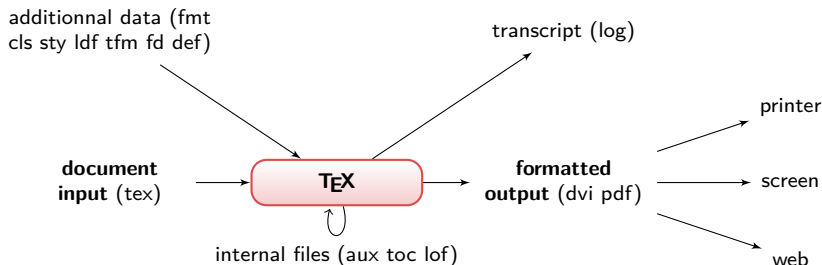
- style is consistent (layout, fonts, tables, maths, etc.)
- mathematics are easily typeset
- indexes, footnotes and references are easily generated
- the author is forced to correctly structure your documents

Cons

- the final result is not visible straight away
- the necessary \LaTeX commands have to be learned
- customization can sometimes be difficult

\LaTeX uses a WYSIWYM-approach: **What You See Is What You Mean**

Simplified overview of the T_EX/L^AT_EX system



Mittelbach Frank & Goossens Michel, *The L^AT_EX Companion, Second Edition*, 2004.

Outline

1 Introduction (or Why using \LaTeX ?)

2 First document

- Document structure
- Form-content separation
- Basic commands

3 Mathematics and code

4 Playing with content

5 Going further

6 Resources

Document structure

```
\documentclass{report} % or article , book , etc.  
  
% this part is the preamble  
% it contains commands that affect the entire document.  
  
\begin{document}  
  
% this part is the body of your document  
% this is where you write your text  
  
\end{document}
```

Comments start with `%` and ends at the end of the line.

Preamble – document class

```
\documentclass[11pt,a4paper,oneside]{report}  
      % global options      class
```

First line of code, it defines the type of document and some global options. Common classes are:

- `article` for articles, presentations, short reports, documentation. . .
- `report` for reports containing several chapters, small books, thesis. . .
- `book` for real books. . .
- `letter` for writing letters.

For more classes and options: <http://en.wikibooks.org/wiki/LaTeX/Basics>

Preamble – package inclusion, with options

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

A lot of packages exist, the most important are:

- `amsmath`, `amssymb` and `amsthm` for mathematical symbols
- `babel` for the internationalization of LaTeX, **mandatory**
- `fontenc` to choose the font encoding of the output text
- `geometry` for easy management of document margins
- `graphicx` to manage external pictures
- `inputenc` to choose the encoding of the input text, **mandatory**

For more packages: <http://en.wikibooks.org/wiki/LaTeX/Packages>

Once the form is fixed for the whole document...

A simple document: `exemple.tex`

```
\documentclass[a4paper,12pt]{article}

\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern} % font
\usepackage{amsmath,amssymb,amsfonts} % maths

\usepackage[english]{babel} % babel

% content begins here
\title{Some explanations about stuff} % the title
\author{Philip J. Fry} % your name
\date{\today} % quite explicit!
```

... the author may concentrate on the content!

```
\begin{document}
\maketitle % generates the title

\begin{abstract} % to sum things up
This document is destined to help you understand things.
\end{abstract}

\section{Introduction} % to create a new section
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Maecenas pretium urna ut nisl semper sed mattis erat interdum.
Vestibulum eget massa nisi.
Donec feugiat consequat leo, a vehicula est imperdiet at.

\section{Do you know I can do that?}
I like maths:  $\forall x \neq 0, \frac{x^2}{x} = x$ 

\end{document}
```

Compiling the source

```
pdflatex exemple.tex
```

and voilà!

Some explanations about stuff

Philip J. Fry

September 17, 2009

Abstract

This document is destined to help you understand things.

1 Introduction

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas pretium urna ut nisl semper sed mattis erat interdum. Vestibulum eget massa nisi. Donec feugiat consequat leo, a vehicula est imperdiet at.

2 Do you know I can do that?

I like maths: $\forall x \neq 0, \frac{x^2}{x} = x$

see it: <http://spechard.dgplug.org/LaTeXForBeginners/exemple.pdf>

Sectioning commands

Code is organized depending on the logical structure of the content:

Command	Level
<code>\part{this is a part}</code>	-1
<code>\chapter{this is a chapter}</code>	0
<code>\section{this is a section}</code>	1
<code>\subsection{this is a subsection}</code>	2
<code>\subsubsection{this is a subsubsection}</code>	3
<code>\paragraph{this is a paragraph}</code>	4
<code>\subparagraph{this is a subparagraph}</code>	5

Note: `\chapter{}` only exists for `book` and `report` classes.

Font size

Ten local environment are available by default to change font size:

Command	Size
<code>\begin{tiny}tiny\end{tiny}</code>	tiny
<code>\begin{scriptsize}scriptsize\end{scriptsize}</code>	scriptsize
<code>\begin{footnotesize}footnotesize\end{footnotesize}</code>	footnotesize
<code>\begin{small}small\end{small}</code>	small
<code>\begin{normalsize}normalsize\end{normalsize}</code>	normalsize
<code>\begin{large}large\end{large}</code>	large
<code>\begin{Large}Large\end{Large}</code>	Large
<code>\begin{LARGE}LARGE\end{LARGE}</code>	LARGE
<code>\begin{huge}huge\end{huge}</code>	huge
<code>\begin{Huge}Huge\end{Huge}</code>	Huge

You can also use it with like this: `\tiny{tiny test}`.

Font styles

As styles are not all available for this this presentation font, examples here are shown in classic Computer Modern.

Command	Style
<code>\textit{italic}</code>	<i>italic</i>
<code>\textsl{slanted}</code>	<i>slanted</i>
<code>\emph{emphasize}</code>	<i>emphasize</i>
<code>\textbf{boldface}</code>	boldface
<code>\texttt{typewriter}</code>	typewriter
<code>\textsc{small caps}</code>	SMALL CAPS

Positioning

```
\begin{center}  
This text is centered.  
\end{center}
```

This text is centered.

```
\begin{flushright}  
This text is flushright.  
\end{flushright}
```

This text is flushright.

```
\begin{flushleft}  
This text is flushleft.  
\end{flushleft}
```

This text is flushleft.

Outline

1 Introduction (or Why using \LaTeX ?)

2 First document

- Document structure
- Form-content separation
- Basic commands

3 **Mathematics and code**

4 Playing with content

5 Going further

6 Resources

Do the maths!

We need: `\usepackage{amsmath}`

Inline with `\begin{math}...\end{math}`, or `\(...\)` or `$...$` like:

```
$\forall x \neq 0, \frac{x^2}{x} = x$
```

$$\forall x \neq 0, \frac{x^2}{x} = x$$

or separate from the text with `\begin{displaymath}...\end{displaymath}` or `\begin{equation}...\end{equation}` or `\[...\]`, which gives:

This is an exemple of equation separate from the text:

$$f_{h,\varepsilon}(x, y) = \varepsilon \mathbf{E}_{x,y} \int_0^{t_\varepsilon} L_{x,y\varphi(\varepsilon u)} \varphi(x) du \quad (1)$$

Maths are easy to type

Powers and indices:

```
\[  
k_{n+1} = n^2 + k_n^2 - k_{n-1}  
\]
```

$$k_{n+1} = n^2 + k_n^2 - k_{n-1}$$

Fractions and binomials:

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

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

Roots:

```
\[  
\sqrt[n]{1+x+x^2+x^3+\ldots}  
\]
```

$$\sqrt[n]{1+x+x^2+x^3+\dots}$$

You can do almost anything you want!

Sums:

```
\[  
\sum_{i=1}^{10} t_i  
\]
```

$$\sum_{i=1}^{10} t_i$$

Integrals:

```
\[  
\int_0^{\infty} e^{-x} \, dx  
\]
```

$$\int_0^{\infty} e^{-x} dx$$

Automatic bracket sizing:

```
\[  
\left(\frac{x^2}{y^3}\right)  
\]
```

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

And sometimes more...

Matrices and arrays:

```
\[
A_{m,n} =
\begin{pmatrix}
a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\
a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{m,1} & a_{m,2} & \cdots & a_{m,n}
\end{pmatrix}
\]
```

$$A_{m,n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix}$$



<http://en.wikibooks.org/wiki/LaTeX/Mathematics>

Print the code!

We need: `\usepackage{listings}`

```
\begin{lstlisting}
  [language=[GNU]C++]
#include <iostream>
using namespace std;

int main () {
  int N = 42;
  for(int i=0; i < N; ++i)
    cout << i << endl;

  return 0;
}
\end{lstlisting }
```

Here is my code:

```
\begin{lstlisting}[language=C++]
#include <iostream>
using namespace std;

int main () {
  int N = 42;
  for(int i = 0; i < N; ++i)
    cout << i << endl;

  return 0;
}
\end{lstlisting }
```

Many languages: Ada, Assembler, bash, C, C++, Caml, Cobol, Delphi, Fortran, HTML, Java, Logo, Matlab, Perl, Pascal, PHP, Prolog, Python, R, Ruby, SQL, TeX, VHDL, XML, XSLT, etc.



<http://www.ctan.org/tex-archive/macros/latex/contrib/listings/>

Outline

1 Introduction (or Why using \LaTeX ?)

2 First document

- Document structure
- Form-content separation
- Basic commands

3 Mathematics and code

4 **Playing with content**

5 Going further

6 Resources

Lists and enumeration

Three environments:

```
\begin{itemize}
  \item first stuff
  \item second thing
\end{itemize}
```

- first stuff
- second thing

```
\begin{enumerate}
  \item first stuff
  \item second thing
\end{enumerate}
```

- 1 first stuff
- 2 second thing

```
\begin{description}
\item[itemize]
  list with bullet points
\item[enumerate]
  numbered list
\item[description]
  list with definitions
\end{description}
```

- itemize** list with bullet points
- enumerate** numbered list
- description** list with definitions

Labels and cross-referencing

Three commands:

- `\label{marker}` to make an object reference a marker, mostly like a name
- `\ref{marker}` to reference the object marked before, it prints the number assigned to the object
- `\pageref{marker}` will print the number of the page where the object is

Important!

This is a two-step process: first the compiler stores the labels, then it replaces the `\ref` with the right number. Then, several compilation are necessary to see the proper output.

If \LaTeX uses deprecated labels information, you get a warning:

LaTeX Warning: There were undefined references.

Tables: a tough one!

The tabular environment: `\begin{tabular}[pos]...{table spec}\end{tabular}`:

pos: vertical position

h	here
b	bottom
c	center
t	top

table spec: column alignment and vertical lines

l	left-justified column
c	centered column
r	right-justified column
p{width}	paragraph column with text vertically aligned at the top
	vertical line
	double vertical line

... : content management

&	column separator
\\	start new row
\hline	horizontal line
\cline	partial line

One of the most difficult part of \LaTeX !

Tables: exemples, please!

```
\begin{tabular}{l c r }
 1 & 2 & 3 \\ \hline
 4 & 5 & 6 \\ \hline
 7 & 8 & 9 \\
\end{tabular}
```

1	2	3
<hr/>		
4	5	6
<hr/>		
7	8	9

```
\begin{tabular}{|r|l|}
 \hline
 7C0 & hexadecimal \\
 3700 & octal \\
 11111000000 & binary \\
 \hline \hline
 1984 & decimal \\
 \hline
\end{tabular}
```

7C0	hexadecimal
3700	octal
11111000000	binary
<hr/>	
1984	decimal

 <http://en.wikibooks.org/wiki/LaTeX/Tables>

Floats

A float is an *unbreakable* box containing text, images, etc.

- to deal with objects that won't fit on the current page
- to help when you really don't want the object here just now
- not part of the normal stream of text, they're separate entities
- may disturb beginners because it breaks they're WYSIWYG mind!

The author specifies placement with:

h	here (in fact as soon as possible in the text flow)
t	at the top of the page (the current or the following)
b	at the bottom of the page (the current or the following)
p	on a special page for floats only (far away)
H	precisely here (requires the <code>float</code> package, to be used wisely)

Figures and tables may be floats

```
\begin{figure}[placement specifier]
... figure contents ...
\end{figure}
```

This way, you can specify a `\caption{}` for your table:

```
\begin{table}[h]
\caption{simple showcase.}
\begin{tabular}{|r|l|}
\hline
7C0 & hexadecimal \\
3700 & octal \\
11111000000 & binary \\
\hline \hline
1984 & decimal \\
\hline
\end{tabular}
\end{table}
```

Table: simple showcase.

7C0	hexadecimal
3700	octal
11111000000	binary
1984	decimal

Remember labels?

Floats are always numbered, so they can be referred to

- use `\label{}` to create a marker
- always put the label after the caption, inside the `float` environment!
- again, (at least) two compilations are needed!

```
\begin{table}[h]
\begin{tabular}{|r|l|}
\hline
7C0 & hexadecimal \\
3700 & octal \\
11111000000 & binary \\
\hline
1984 & decimal \\
\hline
\end{tabular}
\caption{simple showcase.}
\label{tab:1984representations}
\end{table}
```

2.3 Referencing a table

Here, I make a reference and the table 1 of the following section.

3 Tables

7C0	hexadecimal
3700	octal
11111000000	binary
1984	decimal

Table 1: simple showcase.

Graphics

We need: `\usepackage{graphicx}`

The magical command

- use `\includegraphics[options]{name}` to insert a extern graphic document
- there is a lot of options (height, width, trim, page, etc.): read the doc!
- you can insert `png`, `jpg` and `pdf` format images
- make it a float!

```
\begin{figure}[t]
\caption{Saint-Nazaire's bridge.}
\label{fig:SaintNazaireBridge}
\centering
\includegraphics[width=\textwidth,
trim=400px 100px 0px 400px, clip]
{Pont_de_Saint-Nazaire}
\end{figure}
```

Figure 1: Saint-Nazaire's bridge.



4 Graphics

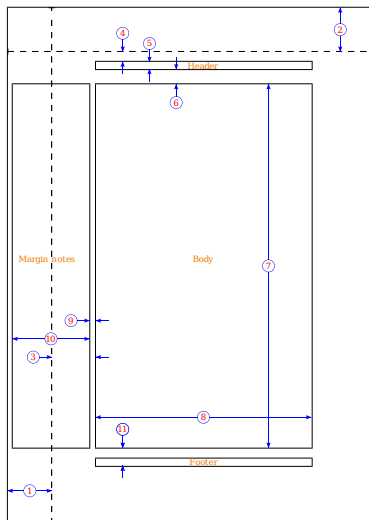
Page Layout

Everything is under control!

- default page size: *US letter* format
- A4 format (21x29.7cm):
`\documentclass[a4paper]{article}`
- everything is a parameter (header, footer, margins, body sizes)

Package `geometry` to define margins, orientation, etc.

- landscape (portrait is default):
`\usepackage[landscape]{geometry}`
- margins: `\usepackage[hmargin=3cm,vmargin=2cm]{geometry}`



http://en.wikibooks.org/wiki/LaTeX/Page_Layout

Headers and footers content

We need: `\usepackage{fancyhdr}`

```
% page headers/footers for a twoside document
\pagestyle{fancy} % mandatory
\fancyhf{} % clear defaults
\fancyhead[RO]{\nouppercase{\emph{\rightmark}}}% RO=right-odd
\fancyhead[LE]{\nouppercase{\emph{\leftmark}}}% LE=left-even

\fancypagestyle{plain}{% for chapter pages
\fancyhf{} % clear defaults
\renewcommand{\headrulewidth}{0pt}} % remove the line
```

using:

- `\thepage` the current page number
- `\leftmark` the current chapter name
- `\rightmark` the current section name
- `\thesection` current section number



http://en.wikibooks.org/wiki/LaTeX/Page_Layout

Useful things



http://en.wikibooks.org/wiki/LaTeX/Useful_Measurement_Macros

Measurement units and macros

- \LaTeX knows a lot of measurement units (pt, bp, mm, cm, in, ex, em)
- and a lot of macros: `\baselineskip`, `\baselinestretch`, `\columnsep`, `\columnwidth`, `\evensidemargin`, `\linewidth`, `\oddsidemargin`, `\paperwidth`, `\paperheight`, `\parindent`, `\parskip`, `\tabcolsep`, `\textheight`, `\textwidth`, `\topmargin`, `\unitlength`, etc.
- you can change their values: `\setlength{\textwidth}{0.5\paperwidth}`
- or make a slight change: `\addtolength{\itemsep}{0.5\baselineskip}`

Tables of interest:

```
\tableofcontents  
\listoffigures  
\listoftables
```

Contents

1	Mathematics	1
2	Labels and cross-reference	2
2.1	Example	2
2.2	Referencing	2
2.3	Referencing a table	2
3	Tables	2
4	Graphics	3

Outline

1 Introduction (or Why using \LaTeX ?)

2 First document

- Document structure
- Form-content separation
- Basic commands

3 Mathematics and code

4 Playing with content

5 Going further

6 Resources

Guidelines to make L^AT_EX coding easier

Split the document into multiple files

- to avoid writing in one giant document ;
- `\input{file}` inserts content as it in file ;
- `\include{file}` is more flexible thanks to `\includeonly{file1, file2}` ;

Create and recreating new commands

- using `\newcommand{name}[num]{definition}` with `name` the name of the new command, `num` number of arguments taken, up to 9 (default=0), `definition` the command itself

```
\newcommand{\note}[1]{\marginpar{% notes in margin
  \vskip-\baselineskip %raise the marginpar a bit
  \raggedright\footnotesize\color{red}
  {\itshape\hrule\smallskip#1}\par\smallskip\hrule}}
\newcommand{\nonotes}{\renewcommand{\note}[1]{} }
```

A bit of debugging: DON'T PANIC!

Everybody makes error like:

- mistype or mis-spell commands
- forget curly braces
- type a forward slash / instead of a backslash \
- use a special character by mistake

But happily:

- errors and warning are well defined in \LaTeX log

```
! Undefined control sequence.  
1.6 \tableofcontents
```

- an editor may help you locate or even avoid them
- an erroneous document is not what you want, so don't let them!

A bit of debugging: What can I do?

On the command line, when \LaTeX finds an error, you have to specify it what to do about it:

Key	Meaning
x	stop and exit \LaTeX
q	carry on as best as possible and stop showing errors
e	stop and position the editor at error line (if possible)
i	correct here (not in the editor) and carry on
r	non-stop mode, ignore errors (up to 100)
h	try to give me more help

Some editors run in non-stop mode and analyze log to show errors and warnings.



http://en.wikibooks.org/wiki/LaTeX/Errors_and_Warnings

Going local :-)

Want to print pretty bengali?

Use XeTeX:

- a variation around \LaTeX (component of TeXLive)
- based on unicode and modern fonts systems
- specific syntax for new features and own binary

```
xelatex sample.tex
```

পাগলা দাশু

সুকুমার রায়

আমাদের স্কুলের যত ছাত্র তাহার মধ্যে এমন কেহই ছিল না, যে পাগলা দাশুকে না চিনে। যে লোক আর কাহাকেও জানে না, সেও সকলের আগে পাগলা দাশুকে চিনিয়া লয়। সেবার একজন নূতন দারোয়ান আসিল, একেবারে আনকোরা পাড়াগোঁয়ে লোক, কিন্তু প্রথম যখন সে পাগলা দাশুর নাম শুনিল, তখনই সে আন্দাজে ঠিক ধরিয়া লইল যে, এই ব্যক্তিই পাগলা দাশু। কারণ তার মুখের চেহারা, কথাবার্তায়, চলনে চালনে বোঝা যাইত যে তাহার মাথায় একটু 'ছিট' আছে। তাহার চোখদুটি গোল-গোল, কানদুটা অনাবশ্যক রকমের বড়, মাথায় এক বস্তা ঝাঁকড়া চুল। চেহারাটা দেখিলেই মনে হয়---

বিঃদ্রঃ এই বাংলা পাতাটি লেখার জন্য **XeTeX** ব্যবহার করা হয়েছে।



<http://methopath.wordpress.com/2008/06/26/writing-unicode-bengali-in-latex/>

Many (many!) packages

- `bibtex` bibliography management ;
- `makeidx` indexes management ;
- `pstricks` low-level and historic graphic package ;
- `pgf` modern, nice and simple graphics (preferred) ;
- `beamer` presentation class, like this one ;
- `hyperref` PDF hyperlinks management (use with caution) ;
- `color` to make colored text (use wisely) ;
- `pdfpages` to insert PDF or PS pages in a PDF ;
- `verbatim` to insert \LaTeX code ;
- `minipage` to create a page in a page ;

Editors

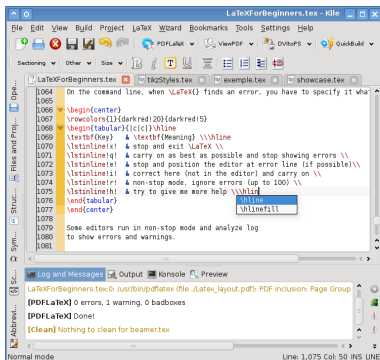
Both ViM and Emacs have a \LaTeX edition mode.

 <http://vim-latex.sourceforge.net/>

 <http://www.gnu.org/software/auctex/>

Kile – a \LaTeX Environment (for KDE)

 <http://kile.sourceforge.net/>



Many resources to learn even more

Books:

- Mittelbach Frank & Goossens Michel, *The L^AT_EX Companion, Second Edition*, 2004.
- Donald E. Knuth, *The TeX book*, 1986.
- [The Comprehensive LaTeX Symbol List \(pdf\)](#)
- [LaTeX Reference Card \(pdf\)](#)

Websites:

- <http://www.latex-project.org/> – official website
- <http://www.ctan.org/> – where you can find everything!
- <http://www.tug.org/> – The TeX Users Group
- <http://www.tex.ac.uk/cgi-bin/texfaq2html> – The UK TeX FAQ
- <http://en.wikibooks.org/wiki/LaTeX/> – a book being written online