

# A Brief Introduction to $\text{\LaTeX}$ \*

Theresa A Scott, MS

Department of Biostatistics  
theresa.scott@vanderbilt.edu  
D-2217 MCN; 3-1713

<http://biostat.mc.vanderbilt.edu/TheresaScott>

\* Pronounced <<Lah-tech>> or <<Lay-tech>>; rhymes with <<blech>>.

## Outline

- 1 Some background information
- 2 How to use  $\text{\LaTeX}$  to generate a document
- 3  $\text{\LaTeX}$  input files
- 4 Some additional information

## Some background information

## Background [1]

### What is $\text{\LaTeX}$ ?

- A system used to typeset documents.
- “Typesetting”: closely related to the printing process; entails arranging all the text (and perhaps tables, and/or graphics/images) in a document in order to achieve the best possible results in terms of stylistics, clarity, and readability.
- “Documents”: include journal articles, technical reports, books, and slide presentations.

### Who uses $\text{\LaTeX}$ ?

- Scientific and academic communities for writing journal articles and theses, which may or may not contain mathematical formulas or other complex notation.
- The technical publishing industry.

## Background [2]

### Where did $\text{\LaTeX}$ come from?

- 1977: Donald Knuth began writing  $\text{\TeX}$ , a typesetting engine.
  - Output was equal to professional printers.
  - Especially good for complex mathematical text.
  - Needed to learn lots of intricate commands.
- Early 1980s: Leslie Lamport created  $\text{\LaTeX}$ .
  - A macro package that allowed authors to use  $\text{\TeX}$  easily.
  - Added many useful tools for automating tedious tasks.
- 1985: First widely available version of  $\text{\LaTeX}$  (2.09) appeared.
- 1994:  $\text{\LaTeX}2\text{e}$  was released.
  - Standardized many enhancements.
  - Contained many new features.

**NOTE:** Until a newer version appears, “LaTeX” means  $\text{\LaTeX}2\text{e}$ .

## Background [3]

### How is $\text{\LaTeX}$ different from a word processor like MS Word?

- With a word processor,
  - Specify the document layout and logical structure interactively while typing text into the document.
  - Use drop-down menus to format typed text.
  - Often generate aesthetically pleasing documents with very little or inconsistent structure.
- With  $\text{\LaTeX}$ ,
  - Takes input from a source file.
    - Simple text file that contains the document’s text, as well as the *commands* that describe the document’s *logical structure* and how to *format* the text.
  - Processes the content and decides how best to typeset the document.
    - Most suitable layout chosen based on specified logical structure.
  - Outputs a device-independent (DVI) file.

## Background [4]

### What does $\text{\LaTeX}$ do?

- Analogy: publishing something involves a *book designer* and a *typesetter*.
  - Book designer:
    - Decides the layout of the document.
    - Also decides on the “logical structure” of the document.
    - Writes their instructions into the document and then gives it to a typesetter.
  - Typesetter:
    - Typesets the book according to the instructions left by the book designer.
- In a  $\text{\LaTeX}$  environment,
  - Book designer =  $\text{\LaTeX}$ .
    - Additional information must be specified in the document because LaTeX is “only” a program and not human.
  - Typesetter =  $\text{\TeX}$ .
    - The  $\text{\LaTeX}$  “instructions” are translated into lower-level  $\text{\TeX}$  typesetting commands.

## Background [5]

### What are the advantages of using $\text{\LaTeX}$ instead of a word processor?

- Only need to learn commands that specify the logical structure of the document; almost never need to tinker with the actual layout of the document.
- Allows you to focus on the content rather than the appearance of your document.
- Easy to typeset complex mathematical formulas.
- Tedious elements like table of contents; lists of figures and tables; numbering of sections, tables, and figures; bulleted/enumerated lists; footnotes; cross-references; and bibliographies are automatically generated AND updated when the document changes.
- Facilitates consistent presentation of your documents.
- Can easily change document styles.

## Background [6]

### What are the advantages of using $\text{\LaTeX}$ instead of a word processor?, cont'd

- Easy to scale your document to create a very large one.
- Portability of the input file (simple text file), of the system itself (runs on almost any hardware platform available), and of the output file produced (can be easily converted to PS or PDF).
- Flexibility – hundreds of add-on packages exist for many typographical tasks not directly supported by basic LaTeX.
  - Examples: including Postscript or PDF graphics in your document; typesetting documents and bibliographies to conform to exact standards of a specific journal; generating slide presentations (like this one).
- Professional quality of output.
- Stability – never crashes or corrupts your files.
- Cost – free!

## Background [7]

### What are the disadvantages of using $\text{\LaTeX}$ ?

- Learning curve (like anything else).
- Not interactive like a word processor.
  - When using  $\text{\LaTeX}$ , not normally possible to see the final output while typing the text, but the final output can be previewed on the screen after processing the file with LaTeX (a matter of a single keystroke or mouse click; typesetting is very fast).
- Can't share  $\text{\LaTeX}$  files with people who use MS Word.
  - Tools available that convert documents between MS Word and  $\text{\LaTeX}$  – check out Word2Tex and Tex2Word from Chikrii Softlab ([www.chikrii.com](http://www.chikrii.com)).
- Writing the  $\text{\LaTeX}$  commands to create more complex data, such as tables, may take more time than the visual approach employed by a word processor.

# How to use $\text{\LaTeX}$ to generate a document

## How do I generate my document?

- Write your input file in a text editor and save it as an ASCII (plain) text file in a specific folder/directory with a `.tex` extension.
  - Less troublesome if the file name contains no spaces.
- From a command prompt, in the folder/directory where your `.tex` file is stored, type the command: `latex filename`
  - NOTE: The `.tex` file extension is not required, although you can include it if you wish.
  - Various bits of info about  $\text{\LaTeX}$  and its progress will be displayed.
  - If all went well, the last two lines displayed in the console will be:  
Output written on filename.dvi (...).  
Transcript writtern on filename.log.
  - DVI file (`.dvi`) created (as well as additional supporting files) in the same folder/directory as your `.tex` file.

## Now what?

- You can view the generated DVI file using any of several viewers.
- You can convert the generated DVI file to a Postscript file (.ps; to then view and/or print) by typing the following command (from a command prompt):

```
dvips filename.dvi -o filename.ps
```

- You can convert the generated DVI file to a PDF file (.pdf; to then view and/or print) in one of several ways:
  - You can convert the generated DVI file to a PDF file by typing:

```
dvipdf filename.dvi filename.pdf
```
  - You can convert the generated Postscript file to a PDF file by typing:

```
ps2pdf filename.ps filename.pdf
```
  - You can create a PDF directly from  $\text{\LaTeX}$  by typing:

```
pdflatex filename
```

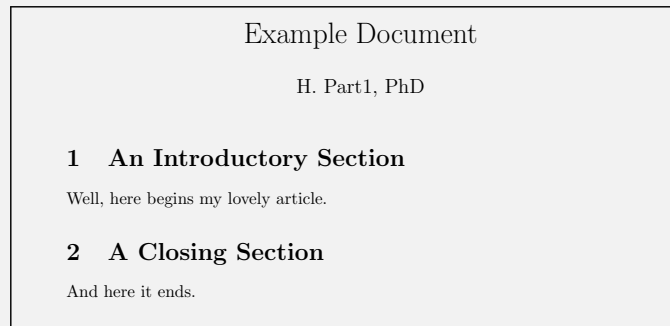
## Is there anything else I should know?

- Must have  $\text{\LaTeX}$  installed for all of these steps to work.
  - On a Windows machine, actually going to install MikTeX (<http://miktex.org>).
- May be necessary to run  $\text{\LaTeX}$  several times to get the table of contents and all internal references right.
  - Not necessary with MikTeX – compiles as many times as necessary.
- When your input file has an error/bug,  $\text{\LaTeX}$  will tell you about it and stop processing your input file.
  - Type Ctrl-D to get back to the command prompt.
  - Will need to find and fix the problem and then re-compile your .tex input file.
- Many text editors to choose from – some recommendations:
  - On a Windows machine, WinEdt ([www.winedt.com](http://www.winedt.com); free; by default already configured for MikTeX).
  - On a Linux/Unix machine, Kate or Emacs.

# How about an example?

The following input file produces the following (cropped) document:

```
\documentclass[10pt]{article}
\title{Example Document}
\author{H. Part1, PhD}
\date{}
\begin{document}
\maketitle
\section{An Introductory Section}
Well, here begins my lovely article.
\section{A Closing Section}
And here it ends.
\end{document}
```



## Section III:

$\LaTeX$  input files



# L<sup>A</sup>T<sub>E</sub>X input files: General structure

Every LaTeX input file has the same general form:

```
prologue
\documentclass[options-list]{document-class}
preamble
\begin{document}
  document-body
\end{document}
```

- NOTE: The special typewriter font indicates what you type in your input file.
- The input file must end with a single blank line after the `\end{document}` command – will receive warnings if you don't.
- Comments can be added to your input file using the % sign – the % and all characters following it (on that line) are ignored.

## L<sup>A</sup>T<sub>E</sub>X input files: General structure, cont'd

### What does all of this mean?

- *prologue*: usually empty; only a very small number of commands can occur in it – will not be discussed any further.
- `\documentclass` command: specifies what sort of document you intend to write.
- *preamble*: includes commands that define information about the document, commands that influence the style of the whole document, and/or `\usepackage` commands that load add-on packages that will be used in the document.
- `\begin{document}` and `\end{document}` commands: signifies the start and finish of the body of the document.
- *document-body*: contains the text of your document and other commands needed to specify the document's logical structure, formatting of the text, and generation of desired elements.

# L<sup>A</sup>T<sub>E</sub>X input files: Commands & environments

## Commands:

- Primary way of telling L<sup>A</sup>T<sub>E</sub>X what you want to do.
- Take one of the following formats:
  - Consist of a backslash `\` followed by a single non-alphanumeric character.
    - Used to specify special characters and line breaks (eg, `\%`).
  - Consist of a backslash `\` followed by a name consisting of only letters (eg, `\noindent`).
  - `\commandname[optional-args]{mandatory-args}`
    - Must provide mandatory arguments (within the `{}`); optional arguments (specified within the `[]`) not always defined.
    - Example: `\includegraphics[0.5]{file.pdf}`

## Environments:

- Defined with a `\begin{env}` and `\end{env}` command.
- Can be thought of as “containers” – “insulate” the typesetting that occur within them (examples to come).

# L<sup>A</sup>T<sub>E</sub>X input files: Document class

- Specified using the `\documentclass` command, which takes the form `\documentclass[options-list]{document-class}`
- The mandatory *document-class* argument specifies the type of document to generate, which includes `article`, `report`, and `book`.
- The optional *options-list* argument customizes the behavior of the specified document class.
  - Is a sequence of one or more arguments separated by commas and *no* spaces (eg, `[12pt,twocolumn]`).
  - Default values for the `article` document class are `10pt`, `letterpaper`, and `notitlepage`.
  - Other options include `titlepage`, and `landscape`; other only applicable to specific document classes (eg, `openright`).
  - If no optional arguments are chosen, then you also leave out the square brackets (eg, `\documentclass{article}`).

# L<sup>A</sup>T<sub>E</sub>X input files: Preamble

Often includes the following commands:

- `\title{document-title}`
- `\author{names}`
- `\date` (can suppress the printing of any date, print today's date, or print a specified date depending on its format).
- `\pagestyle{style}`, which specifies the content and format of the header and footer on *all* pages of the document.
  - The mandatory *style* argument can be specified as `plain` (default), `empty`, or `headings`.
- `\usepackage[options]{package-name}` commands, which allow you to enhance the capabilities of basic L<sup>A</sup>T<sub>E</sub>X.
  - Provide additional commands and/or environments.
  - Some packages come with the L<sup>A</sup>T<sub>E</sub>X base distribution, while others are provided separately.

# L<sup>A</sup>T<sub>E</sub>X input files: Body of the document

Often includes (not necessarily in this order):

- The `\maketitle` command (no arguments), which produces the title of the document.
  - Must be preceded by both the `\title` and `\author` commands in the preamble; also prints the date by default.
  - Usually placed directly below `\begin{document}` command.
- The `\tableofcontents` command (no arguments), which creates a Table of Contents.
  - Usually placed directly below the `\maketitle` command.
  - Takes the section headings and page numbers from the last compile of the document.
  - A new document has to be compiled twice to get a correct Table of Contents.

## L<sup>A</sup>T<sub>E</sub>X input files: Body of the document, cont'd

In addition, often includes (not necessarily in this order):

- The `\section[entry]{heading}` command and its kin (`\subsection` and `\subsubsection`), which are used to start and label (with *heading* and number) new “sections” of the document.
  - The optional *entry* argument can be used to specify a shorter heading to be used in the Table of Contents and in page headers/footers.
  - Additional sectioning commands exist for specific document classes (eg, `\chapter` for the book document class).
- Paragraphs, which are ended by one or more completely blank lines (equivalent to the `\par` command – no arguments).
  - By default, a new paragraph is indented, but you can suppress this indentation by using the `\noindent` command (no arguments).

## L<sup>A</sup>T<sub>E</sub>X input files: Body of the document, cont'd

In addition, often includes (not necessarily in this order):

- The `\footnote[i]{text}` command, which produces footnotes.
  - Can use the optional numerical argument *i* to specify the footnote number (instead of allowing the internal footnote counter to determine its number).
  - The footnote number is printed as a superscript in the body of the page where the `\footnote` command occurs and also at the bottom of the body of the page where *text* is printed.
- The `itemize` and `enumerate` *environments*, which generate bulleted lists and numbered lists, respectively.
  - The `\item` command is used to indicate the start of a new entry in the list.
  - Can be nested within each other (to create multiple indented lists; only 4 levels permitted) as long as the correct nesting order is maintained.

## L<sup>A</sup>T<sub>E</sub>X input files: Body of the document, cont'd

In addition, often includes (not necessarily in this order):

- The `tabular` *environment*, which generates a table.
  - If nested within a `table` *environment*, can add a caption using the `\caption` command (similar to the `\section` command).
    - If captions defined, can use the `\listoftables` command to produce a List of Tables – similar to the `\tableofcontents` command and usually placed directly below it.
    - Causes the table to “float” (placed on a later page) if it doesn’t fit on the current page (current page filled with body text) – this behavior can be modified in the declaration of the `table` *environment*.
- The `\includegraphics[options]{file}` command, which inserts an Encapsulated Postscript (EPS) or PDF image.
  - Need to have `\usepackage{graphicx}` in preamble.
  - If nested within a `figure` *environment*, can add a caption and causes similar “floating” behavior (see above).
  - Similar `\listoffigures` command.

## L<sup>A</sup>T<sub>E</sub>X input files: Body of the document, cont'd

In addition, often includes (not necessarily in this order):

- Cross-references to tables, figures, sections, and more.
  - The `\label{marker}` command is placed next to the command/environment that generates the table, figure, section, etc you wish to reference.
  - The `\ref{marker}` and/or `\pageref{marker}` commands then produce the associated reference value and page number, respectively.
  - The reference/page numbers from the previous compiling are used (similar to section titles; may have to compile more than once).
- The `flushleft`, `flushright`, and `center` *environments*, which left-, right-, or center-align the body of the document, respectively.
  - Can be wrapped around text, inserted images, or other environments, such as `tabular/table` or `figure`.

## L<sup>A</sup>T<sub>E</sub>X input files: Body of the document, cont'd

In addition, often includes (not necessarily in this order):

- Spaces, special characters, and quotation marks:
  - 1 space is as good as 100 to mark the end of words/sentences.
  - The # \$ % & \_ { } ~ symbols are special characters and can be used by adding a prefix backslash \ (eg, \\_).
  - Also includes the backslash itself – must use `\backslash`.
  - Use two ' (back-tick) for opening quotation marks and two ' (apostrophe) for closing quotation marks (eg, ‘ ‘quotes’ ’’); use one of each for single quotes.
- Line and page breaking commands – none have arguments.
  - The `\linebreak` command forces a line break; similarly the `\\` command starts a new line.
  - The `\pagebreak` command forces a page break, the `\newpage` command starts a new page, and the `\clearpage` command prints all figures and tables and then starts a new page.

## L<sup>A</sup>T<sub>E</sub>X input files: Body of the document, cont'd

In addition, often includes (not necessarily in this order):

- Font changing commands, including ones that modify the:
  - Font, such as the `\textit{text}` command that *italicizes* text, the `\textbf{text}` command that **bolds** text, and the `\textsc{text}` command that produces SMALL CAPS.
  - Font size:
    - The actual size of each font is a design issue and depends on the document class and its options.
    - The font size commands modify the size of the text *relative* to its “normal” size defined in the document class statement.
    - Example: This is “normal”-sized font, whereas the `\tiny` command produces tiny font and the `\Huge` command produces the **largest** font.
- Mathematical notation, formulas, and graphics – see References.
- Bibliographies – see References.

### Some additional information

## Where can I learn more?

- General introduction: *LaTeX: A Document Preparation System* by Leslie Lamport.
- More detailed description: *The LaTeX Companion* by Michel Goossens, Frank Mittelbach, and Alexander Samarin.
  - Includes an overview of more than 150 add-on packages.
  - Contains a CD that has a complete LaTeX installation.
- Documentation section of the LaTeX project site ([www.latex-project.org](http://www.latex-project.org); L<sup>A</sup>T<sub>E</sub>X downloads available).
- Andrew Roberts' "Getting to grips with LaTeX" ([www.andy-roberts.net/misc/latex](http://www.andy-roberts.net/misc/latex)).
- Tobias Oetiker's "The (Not So) Short Introduction to LaTeX2e" ([www.ctan.org/tex-archive/info/lshort/english](http://www.ctan.org/tex-archive/info/lshort/english)).
- CTAN ([www.ctan.org](http://www.ctan.org)) and TUG ([www.tug.org](http://www.tug.org)).