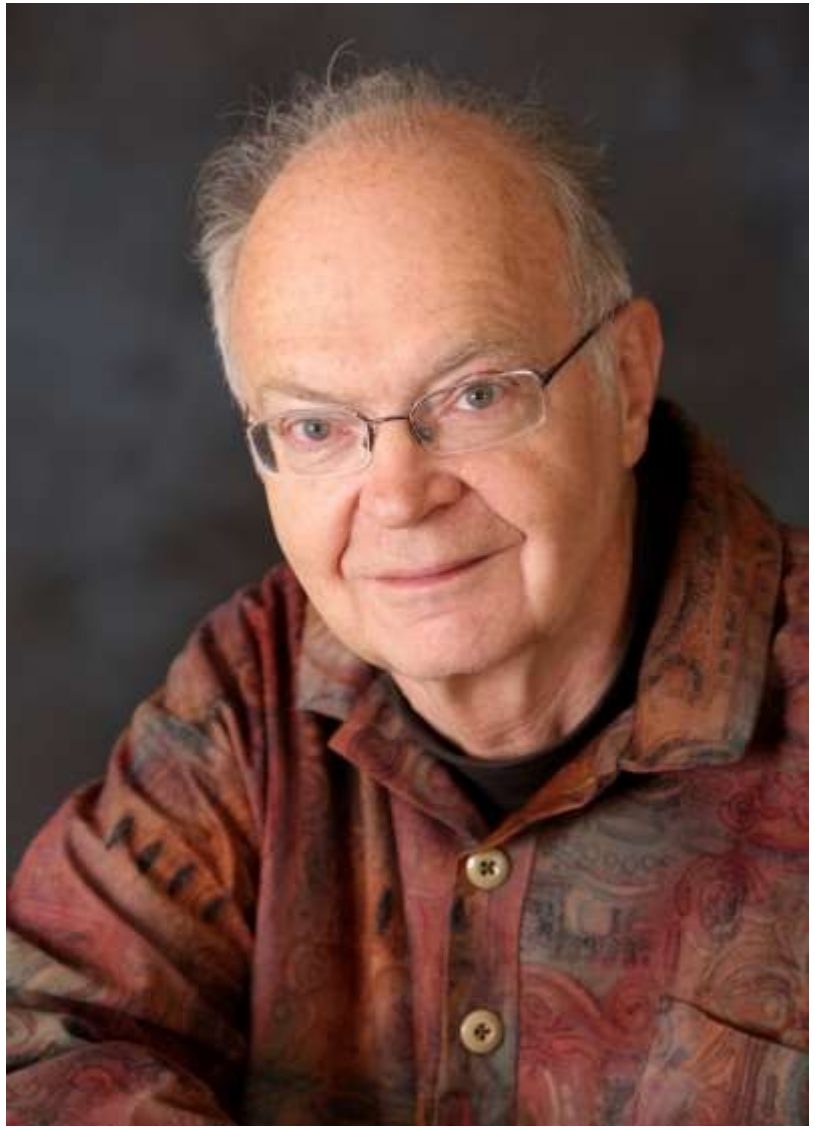
Introduction to LaTeX

„Creating documents with LaTeX is simple and fun. “

(Claudio Vellage)

Based on LaTeX tutorial by Claudio Vellage and the Latex for Beginners workbook (University of Edinburgh).



Donald Ervin Knuth (1938 -), PhD

„Father“ of software engineering.

Author of the ***bible*** for all computer scientists: **The Art of Computer Programming** (this is actually a series of 4 (7) books).

Author of **TeX** system for ducument preparation.

Author of **MF** system for alphabet design.

Authow of **WEB** and **CWEB** languages for structured documentation.

....

What is LaTeX?

- LaTeX is a **document preparation system** for producing professional-looking documents.
- It‘s suited for making long, structured documents.
- It is very good at typesetting **equations**.
- It is available as **free software** for most operating systems.
- LaTeX is based on TeX, a typesetting system designed by **Donald E. Knuth** in 1978 for high quality digital typesetting.

What is LaTeX?

- Microsoft Word is ‘What You See Is What You Get’ (WYSIWYG) type of editor. You see how the final document will look as you are typing.
- With LaTeX you do not see how the final document will look while you are typing it - this allows you to concentrate on the content rather than appearance.
- A LaTeX document is a plain text file with a **.tex** file extension. It can be typed in a simple text editor such as Notepad, but it is easier to use a dedicated LaTeX editor.
- As you type you **mark** the document structure (title, chapters, subheadings, lists etc.) **with tags**.
- When the document is finished you **compile** it — this means **converting** it into another format.
  - Several different output formats are available, but probably the most useful is PDF

Online LaTeX editor (for today)

https://www.overleaf.com/

My first LaTeX file – Hello World!

```
\documentclass{article}
\begin{document}
  Hello World!
\end{document}
```

A backslash \ tells LaTeX this is not an actual text you want to see printed in your document but instead is an instruction or command for the LaTeX compiler. All commands share the structure:

```
\commandname{option}
```

```
\begin{document}
  ...
\end{document}
```

Text, images, tables etc. That you want to actually see printed in the document, should be put between the `\begin{document}` and `\end{document}` statements. This defines an environment – an area of the document where certain typesetting rules apply.

Document class

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

Document types https://www.ctan.org/topic/class

book
report
cv
….

Other font sizes, 9pt, 11pt, 12pt, can be used, but if none is specified, the default size is 10pt. As for the paper size, other possible values are a4paper, letterpaper and legalpaper.

My first LaTeX file - environments

```
\begin{document}
  \begin{environment1}
    \begin{environment2}
    \end{environment2}
  \end{environment1}
\end{document}

%Invalid:
\begin{document}
  \begin{environment1}
    \begin{environment2}
  \end{environment1}
    \end{environment2}
\end{document}
```

```
\begin{env1}
  ...
\end{env2}
```

It is possible (and usually necessary) to have multiple environments in a document.
The `document` environment is always the topmost environment.

There are numerous different environments and we will need them for inserting mathematical formulas and figures to the document.

It is possible to define your own environments, but it is very likely that the environment you desire already exists. LaTeX itself comes with a few predefined environments.

My first LaTeX file – title page

```
\documentclass{article}

\title{My first document}
\date{17 October 2017}
\author{Vida Groznik}

\begin{document}
  \maketitle
  \newpage

  Hello World!
\end{document}
```

The area before our main document is called ***preamble***.
In this specific example we use it to set up the values for the `\maketitle` command for later use in our document.
This command will create a titlepage for us.
The `\newpage` command speaks for itself.

If you don‘t want a number at the bottom of the first page, You can do so by adding the `\pagenumbering{gobble}` command and then changing it back to `\pagenumbering{arabic}` on the next page numbers:

```
\begin{document}
  \pagenumbering{gobble}
  \maketitle
  \newpage
  \pagenumbering{arabic}
    Hello World!
\end{document}
```

My first LaTeX file – sections

```
\documentclass{article}

 \title{My first document}
 \date{17 October 2017}
 \author{Vida Groznik}

 \begin{document}
   \maketitle
   \pagenumbering{gobble}
   \newpage
   \pagenumbering{arabic}

   \section{Section}
   Hello World!
   \subsection{Subsection}
   Structuring a document is easy!

 \end{document}
```

LaTeX offers commands to structure the content of the documents into logic units.

```
\section{}
\subsection{}
\subsubsection{}

\paragraph{}
\subparagraph{}
```

Using the `\section{}` command it generates section headings and numbers them automatically.
The *section* commands are numbered and will appear in The table of contents of your document. *Paragraphs* aren't numbered and won't show in the table of contents.

My first LaTeX file – sections

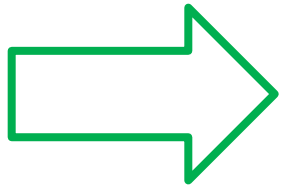
```
\documentclass{book}

 \begin{document}

   \chapter{First Chapter}

   \section{Section}
   Hello World!
   \subsection{Subsection}
   Structuring a document is easy!
   \section*{Unnumbered Section}
   Lorem ipsum dolor sit amet

 \end{document}
```

Documents created using the book class can be split into parts, chapters, sections, subsections and so forth but the letter class does not provide (support) any commands to do that.

```
\part{part} (only report and book class)
\chapter{chapter} (only report and book class)
\section{section}
\subsection{subsection}
\subsubsection{subsubsection}
\paragraph{paragraph}
\subparagraph{subparagraph}
```

My first LaTeX file – sections (continued)

```
\documentclass{article}

\begin{document}

  \section{Section}
  Hello World!

  \subsection{Subsection}
  Structuring a document is easy!

  \subsubsection{Subsubsection}
  More text.

  \paragraph{Paragraph}
  Some more text.

  \subparagraph{Subparagraph}
  Even more text.

  \section{Another section}

\end{document}
```



1 Section

Hello World!

1.1 Subsection

Structuring a document is easy!

1.1.1 Subsubsection

More text.

Paragraph Some more text.

Subparagraph Even more text.

2 Another section

Comments

```
\documentclass{article}

\begin{document}

  % this is the
  text that will
  not be displayed
  in the document

  \section{Section}
  Hello World!

\end{document}
```

A LaTeX comment is a section of text that will not be typeset or affect the document in any way—often used to add “to do” notes; include explanatory notes; provide in-line explanations of tricky macros or comment-out lines/sections of LaTeX code when debugging.

Comments are created using %. When LaTeX encounters a % character while processing a *.tex* file, it ignores the rest of the line (until the [Return] key has been pressed to start a new line — not to be confused with line wrapping in your editor). This can be used to write notes in the input file which will not show up in the printed version.

My first LaTeX file – packages

```
\documentclass{article}

\usepackage{PACKAGENAME}

\begin{document}
...
\end{document}
```

```
\documentclass{article}

\usepackage{amsmath}

\begin{document}

  \begin{equation*}
    f(x) = x^2
  \end{equation*}
```

LaTeX offers a lot of functions by default, but in some situations it can become in handy to use so called *packages*.
To import a package in LaTeX, add the `\usepackage` command to the *preamble* of your document.

In case of Ubuntu installing *texlive-full* from the package manager would provide all packages available.

To typeset math, LaTeX offers (among others) an ***environment*** called ***equation***. Everything inside this environment will be printed in *math mode*, a special typesetting environment for math.
LaTeX automatically numbers the equations in the text. But if you don‘t want to have the equations automatically numbered, you have to include a package that allows you to do that.

More options

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

\usepackage{graphicx}

\title{My first document}
\date{17 October 2017}
\author{Vida Groznik}
\begin{document}
  \maketitle
  \newpage
  Hello World!
\end{document}
```

The preamble line

```
\usepackage{graphicx}
```

is an example of loading an external package (here, graphicx) to extend LaTeX's capabilities, enabling it to import external graphics files.

My first LaTeX file – math features

```
...
This formula $E=mc^2$ is
an example.
...
```

```
\documentclass{article}

\usepackage{amsmath}

\begin{document}

  \begin{equation*}
    1 + 2 = 3
  \end{equation*}

  \begin{equation}
    1 = 3 - 2
  \end{equation}

  \begin{align*}
    1 + 2 &= 3\\
    1 &= 3 - 2
  \end{align*}

\end{document}
```

There are two major modes of typesetting math in LaTeX:

- embedding the math directly into your text by ***encapsulating*** a formula in ***dollar signs*** and
- using a predefined ***math environment***.

The most useful *math environments* are:

- ***equation*** *environment* for typesetting single equations and
- ***align*** environment for multiple equations and automatic alignment.

The ***align*** environment aligns the equations at the ***sign &***. Single equations have to be *separated* by a *linebreak* \\. There is no alignment when using the simple *equation* environment. It is not possible to enter two equations in that environment, it will result in a *compilation error*. The asterisk (e.g. equation*) indicates, that we don't want the equations to be numbered.

My first LaTeX file – figures

```
\documentclass{article}

\usepackage{graphicx}

\begin{document}

  \begin{figure}
    \includegraphics[width=\linewidth]{boat.jpg}
    \caption{A boat.}
    \label{fig:boat1}
  \end{figure}

  Figure \ref{fig:boat1} shows a boat.

\end{document}
```

LaTeX automatically indexes all pictures and tags them with successive numbers when using the ***figure environment*** and the ***graphicx package***.

The *figure* environment takes care of the numbering and positioning of the image within the document. To include a figure, use the `\includegraphics` command. It takes the image width as an option in brackets and the path to your image file.
`\linewidth` - the picture will be scaled to fit the width of the document.
`\caption` – sets the text shown below the image
`\label` - is invisible, but useful for refering to the figure in the document. To refer to the figure, use `\ref{label}` in the text. It will be replaced by the correct number.

You need to include the ***graphicx*** package in order to use this code.

My first LaTeX file – figures (continued)

```
\documentclass{article}

\usepackage{graphicx}

\begin{document}

  \begin{figure}[h!]
    \includegraphics[width=\linewidth]{boat.jpg}
    \caption{A boat.}
    \label{fig:boat1}
  \end{figure}

  Figure \ref{fig:boat1} shows a boat.

\end{document}
```

The figure doesn't necessarily show up in the place where you put your code in the .tex file. LaTeX will put the picture on the page where it finds sufficient space.
To prevent this behavior, it's necessary to set the ***float*** value for the figure environment.

```
%...
\begin{figure}[h!]
%...
```

Setting the float by adding *[h!]* behind the figure environment `\begin` tag will force the figure to be shown at the location in the document. Possible values are:

- h (here) - same location
- t (top) - top of page
- b (bottom) - bottom of page
- p (page) - on an extra page
- ! (override) - will force the specified location

Bold, italics and underlining

```
Some of the \textbf{greatest} discoveries in
\underline{science} were made by
\textbf{\textit{accident}}.

Some of the greatest \emph{discoveries} in
science were made by accident.

\textit{Some of the greatest \emph{discoveries}
in science were made by accident.}

\textbf{Some of the greatest \emph{discoveries}
in science were made by accident.}
```

- **Bold**: bold text in LaTeX is typeset using the \textbf{...} command.
- *Italics*: italicised text is produced usingthe \textit{...} command.
- Underline: to underline text use the \underline{...} command.

Another very useful command is \emph{*argument*}, whose effect on its *argument* depends on the context. Inside normal text, the emphasized text is italicized, but this behaviour is reversed if used inside an italicized text—see the next example:

Creating lists in LATEX – unordered

```
\documentclass{article}
\begin{document}

\begin{itemize}
   \item The individual entries are indicated
         with a black dot, a so-called bullet.
   \item The text in the entries may be of any
         length.
\end{itemize}

\end{document}
```

Unordered lists are produced by
the itemize environment. Each list entry must be
preceded by the \item command, as shown below:

Creating lists in LATEX – ordered

```
\documentclass{article}
\begin{document}

\begin{enumerate}
    \item The individual entries are indicated
          with a black dot, a so-called bullet.
    \item The text in the entries may be of any
          length.
\end{enumerate}

\end{document}
```

As with unordered lists, each entry must be preceded by the `\item` command which, here, automatically generates the numeric ordered-list label value, starting at 1.

Abstract

```
\documentclass{article}
\begin{document}

\begin{abstract}
This is a simple paragraph at the beginning of
the document. A brief introduction about the
main subject.
\end{abstract}

After our abstract we can begin the first
paragraph.

This line will start a second paragraph.

I will start the third paragraph and then add \\
a manual line break which causes this text to
start on a new line but remains part of the same
paragraph.

\end{document}
```

Scientific articles usually provide an *abstract* which is a brief overview/summary of their core topics, or arguments.

A new paragraph is created by pressing the "enter" key twice, ending the current line and inserting a subsequent blank line.

A new line without starting a new paragraph by inserting a manual line break using the \\ command, which is a double backslash; alternatively, use the \newline command.

My first LaTeX file – tables

```
\documentclass{article}

\begin{document}

  \begin{table}[h!]
    \centering
    \caption{Caption for the table.}
    \label{tab:table1}
    \begin{tabular}{l|c||r}
      1 & 2 & 3\\
      \hline
      a & b & c\\
    \end{tabular}
  \end{table}

\end{document}
```

LaTeX offers an environment for table creation. For this purpose we use the ***table***, ***tabular*** and the ***center*** *environment*.

The ***table*** *environment* holds our other environments and allows to add a caption to our table.
The data is contained in the ***tabular*** *environment.*
Use the ***center*** *environment* to center the table on the page.

The *ampersands* & are ***column separators*** and *newline symbols* `\\` are ***row separators***.
Vertical lines, passed as an argument to the tabular environment (e.g. `\begin{tabular}{l|c||r}`) and the letters tell whether we want to align the content to the **left (l)**, to the **center (c)** or to the **right (r)** for each column.

Row separators can be added with the `\hline` command.
`\caption` and `\label` commands can be used in the same way as for pictures.

My first LaTeX file – tables

```
\documentclass{article}

\begin{document}

  Table \ref{table:data} shows how to
  use a caption and reference it.

  \begin{table}[h!]
     \centering
     \begin{tabular}{||c c c c||}
     ...
     \end{tabular}
     \caption{Description}
     \label{table:data}
  \end{table}

\end{document}
```

We can caption and reference tables in much the same way as images. The only difference is that instead of the **figure** environment, you use the **table** environment.

My first LaTeX file – table of contents

```
\documentclass{article}

\begin{document}

  \tableofcontents
  \newpage

  \section{Section}

  Dummy text

  \subsection{Subsection}

  Dummy text

\end{document}
```

Generating a ***table of contents*** can be done with a few commands. LaTeX uses the section headings to create the *table of contents.*

```
\tableofcontents
```

You can also create a ***list of figures*** and a ***list of tables***.

```
\listoffigures
\listoftables
```

Example:
https://www.overleaf.com/11640013pcccqbjgqpnq