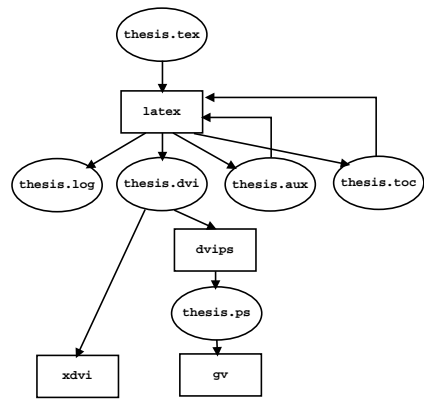




1



2

The most fundamental \LaTeX component is the

Environment

Inside an environment the text gets a special layout and/or special commands are defined.

This is a paragraph with some surrounding text.

```
\begin{itemize}
  \item This is the first point.
  \item And here comes number two.
\begin{enumerate}
  \item Multiple levels are possible
  \item They get automatically indented and enumerated.
\end{enumerate}
  \item The last point
\end{itemize}
```

We also have some text after the different items.

3

This is a paragraph with some surrounding text.

- This is the first point.
- And here comes number two.
 1. Multiple levels are possible
 2. They get automatically indented and enumerated.
- The last point

We also have some text after the different items.

4

A Short Document

```
\documentclass{rapport}
\usepackage{schoolbook}
\usepackage[swedish]{babel}
\begin{document}
  Här kommer texten till mitt banbrytande dokument.
\end{document}
```

The part between `\documentclass` and `\begin{document}` is called the *preamble*, and may contain definitions special to this document. In particular it may call on *packages* with the `\usepackage` command.

There are also *style options*

```
\documentclass[chapter,final]{rapport}
```

5

Documentclasses

Standard \LaTeX :

article report book letter

Do not use these unless you have to.

Use the local classes

rapport artikel lic avhandling pm brev onslide

More classes will be created as needed.

Options for standard classes

10pt 11pt 12pt final

Options for local classes

The above plus

9pt brevhuvud letterhead landscape

6

Special Characters

To get	Write	Used for
\$	<code>\\$</code>	Start and end of math
%	<code>\%</code>	Comment to end of line
&	<code>\&</code>	Column separator
_	<code>_</code>	Math subscript
^	<code>\^{}</code>	Math superscript
#	<code>\#</code>	Parameter placeholder
{	<code>\{</code>	Start group or parameter
}	<code>\}</code>	End group or parameter
\	<code>\textbackslash</code>	Command character
~	<code>\texttildelow</code>	Non-breaking space

The last character requires `\usepackage{textcomp}`. Often `\usepackage{url}` is better.

7

Miscellaneous Commands

Sectioning

```
\chapter{...} \section{...} \subsection{...}
\subsubsection{...}
```

Type Size

```
\tiny \scriptsize \footnotesize \small \normalsize
\large \Large \LARGE \huge
```

Footnotes

```
\footnote{Some explanation}
```

gives an automatically numbered footnote

Cross Reference

```
\label{key} Define the reference. One place.
\ref{key} Use the reference. Many places.
key is any user-defined string
```

8

New Paragraph and Vertical Space

The command `\par` or an *empty line* ends a paragraph. Any text starts a new paragraph.

To make a stretchable vertical space, use the commands `\smallskip`, `\medskip`, or `\bigskip` respectively. To make a fixed, possibly larger vertical space, use `\vspace{55mm}`. These commands should be used *between* paragraphs.

The paragraph indent and the spacing between paragraphs are decided by the document class. Do not make local changes for each paragraph!

Note! Some \LaTeX tutorials claim that the command `\\` ends a paragraph. This is **WRONG!**

9

Environments

`\begin{center}...\end{center}`

Centered lines, use `\\` to separate

`\begin{quotation}...\end{quotation}`

Narrower than surrounding text

`\begin{itemize}...\end{itemize}`

"Ticked" items, bullet default

`\begin{enumerate}...\end{enumerate}`

Numbered items. Automatic numbering

`\begin{description}...\end{description}`

Labeled items.

In the last three cases the item is started with an `\item` command. The description needs an argument: `\item[keylabel]`. `itemize` and `enumerate` take an optional argument.

10

Multiple command forms

Many commands have multiple forms

▲ Optional argument:

`\item[\spadesuit]`

▲ Star-form:

`\section*{An Unnumbered Section}`

They may sometimes be combined.

The meaning of star-forms and optional arguments vary from command to command, but in practice this is not a problem.

11

Grouping

A pair of curly braces `{...}` in the text delimit a \LaTeX group. Any change made to a property (size, font, width, etc.) is only valid inside the group.

Some people like `{\small small text}` and others `{\Large tend to shout}`. Back to normal size.

Some people like small text and others **tend to shout**. Back to normal size.

A \LaTeX environment is an implicit group, so after `\begin{center}\LARGE ... \end{center}` the text size would be back to normal.

12

Type Style 1

\LaTeX type style is specified by three components: shape, series, family.

Italic shape `\textit{Italic shape}`

Slanted shape `\textsl{Slanted shape}`

SMALL CAPS SHAPE `\textsc{Small Caps shape}`

Boldface series `\textbf{Boldface series}`

Roman family `\textrm{Roman family}`

Sans Serif family `\textsf{Sans Serif family}`

Typewriter family `\texttt{Typewriter family}`

Bold italic text `\textbf{\textit{Bold italic text}}`

Use `\emph{...}` to get *emphasized text* inside other text.

`\emph{... \emph{...} ...}` will work properly.

These commands work only in text mode. In math mode, use `\mathrm`, `\mathbf`, `\mathit` etc.

13

Type Style 2

Each of the commands in the previous slide have a corresponding *declaration*.

`{\itshape Italic shape}`

`{\slshape Slanted shape}`

`{\scshape Small Caps shape}`

`{\bfseries Boldface series}`

`{\rmfamily Roman family}`

`{\sffamily Sans Serif family}`

`{\ttfamily Typewriter family}`

The `{\em ...}` declaration corresponds to the `\emph{...}` command.

The old commands `\it` `\bf` `\ss` `\tt` etc. also work, but please do not use them in new documents.

14

Verbatim

Short verbatim strings: `\verb? any $ % & # characters?`

Result: any \$ % & # characters. The special marker may be any nonalphanumeric character.

Longer verbatim text is created with the `{verbatim}` environment.

```
\begin{verbatim}
Text with    & % # any characters $ \ } {
  except the special string
\end{verbatim}
```

The special string is `\end{verbatim}`

Result:

```
Text with    & % # any characters $ \ } {
  except the special string
```

15

Floating Figures and Tables

`\begin{figure}`

`\begin{center}`

`\insert the graphics here`

`\end{center}`

`\caption{...}\label{faa}`

`\end{figure}`

`\begin{table}`

`\caption{...}\label{taa}`

`\begin{center}`

`%tabular material here`

`\end{center}`

`\end{table}`

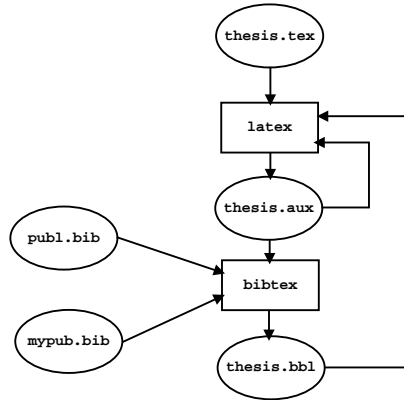
Makes a floating insert. Note different placement of `\caption`. This is a tradition, not a technical requirement. Note also that the `\label` must come after the `\caption`.

Both environments can take an optional argument specifying desired position. Do not use this until the really final version of the document. In particular, do not use the `[h]` variant at all.

Do not confuse the `{table}` environment with the `{tabular}` environment described later.

16

The L^AT_EX—BibT_EX Cycle



25

Defining and Redefining Commands

`\newcommand\block{\vrule height 3mm width 3mm\relax}` defines a command `\block` that gives a black square.

The new command *must not* exist previously. L^AT_EX will refuse if it does.

Use `\renewcommand{...}` to redefine a command, if you really know what you are doing. The command *must* exist previously. L^AT_EX will refuse if it does not.

`\newcommand\itbold[1]{\textit{\textbf{#1}}}` defines a command with one parameter, to create `\itbold{bold italic}` ***bold italic*** text.

26

Defining and Redefining Environments

`\newenvironment{largebold}{\large\bfseries}{\par}`

```

\begin{largebold}
  The quick brown fox jumps
  over the lazy dog's back.
\end{largebold}
  
```

The quick brown fox jumps over the lazy dog's back.

There is also `\renewenvironment{...}`. Same rules as for `\newcommand` and `\renewcommand`

`\newenvironment{ruledtext}[1]{\hrule\vspace{#1}\small}{\par\bigskip\hrule}`

```

\begin{ruledtext}{1cm}
  Ett stycke text
\end{ruledtext}
  
```

Ett stycke text

27

Theorems and such

`\newtheorem{theo}{Theorem}` `\newtheorem{lem}{Lemma}` in the preamble create two new environments:

```

\begin{lem}
  Lemmas are boring.
\end{lem}
\begin{theo}
  Some theorems need lemmas.
\end{theo}
  
```

LEMMA 1
Lemmas are boring. □

THEOREM 1
Some theorems need lemmas. □

Other theoremstyles are possible.

28