LaTeX
- an introduction -

Marvin Reich

Summerretreat 2014
Section 5.4 (Hydrology)

GFZ
Helmholtz-Zentrum
Potsdam

1 Overview

2 The setup

3 How does LaTeX work?

4 Getting your hands on..

5 Creating your own LaTeX document

6 Recommendations

What LaTeX is..

..a document preparation system and document markup language

..based on Donald E. Knuth's typesetting system TeX

..used for publication of scientific documents in many fields

..NOT a word-processor (WYSIWYG)

..outputs in DVI, PostScript and PDF

Pro's and con's

- output looks awesome
- platform independent
- formatting options for the whole document in one command
- modified tables or graphics will update without work on compiling
- templates for quick content filling
- using same content for different projects

- needs some time to be learned
- usability not convincing for all projects (presentation? poster?)
- sharing projects only possible with other LaTeXusers

Pro's and con's

- output looks awesome
- platform independent
- formatting options for the whole document in one command
- modified tables or graphics will update without work on compiling
- templates for quick content filling
- using same content for different projects

- needs some time to be learned
- usability not convincing for all projects (presentation? poster?)
- sharing projects only possible with other LaTeXusers

LaTeX distributions

MiKTeX:

- especially build for Windows
- selectable extend of package installation
- http://miktex.org/

TeX Live:

- oldest distributions
- installs with big functionality and extra packages
- Windows and Unix
- http://www.tug.org/texlive/

MacTeX:

- same as TeX Live, only compiled for MacOS
- http://www.tug.org/mactex/

LaTeX distributions

MiKTeX:

- especially build for Windows
- selectable extend of package installation
- http://miktex.org/

TeX Live:

- oldest distributions
- installs with big functionality and extra packages
- Windows and Unix
- http://www.tug.org/texlive/

MacTeX:

- same as TeX Live, only compiled for MacOS
- http://www.tug.org/mactex/

LaTeX distributions

MiKTeX:

- especially build for Windows
- selectable extend of package installation
- http://miktex.org/

TeX Live:

- oldest distributions
- installs with big functionality and extra packages
- Windows and Unix
- http://www.tug.org/texlive/

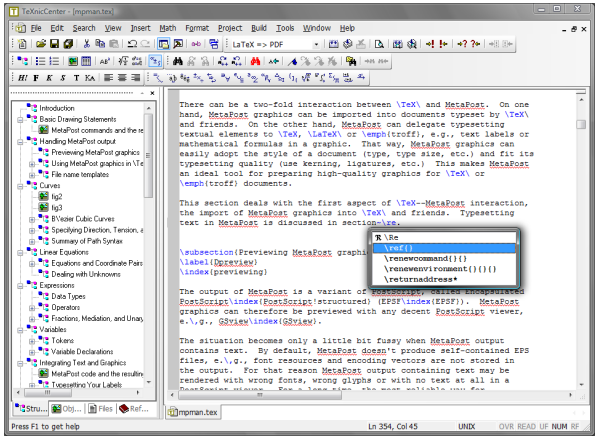
MacTeX:

- same as TeX Live, only compiled for MacOS
- http://www.tug.org/mactex/

Choosing an editor

There are 3 options:

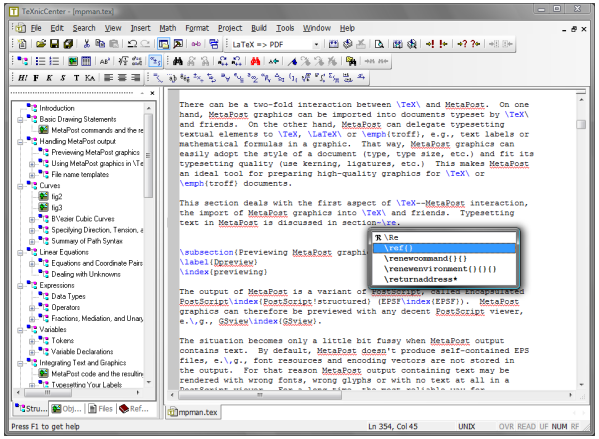
- **"All-in-one solution"** (TEXnicCenter)



Choosing an editor

There are 3 options:

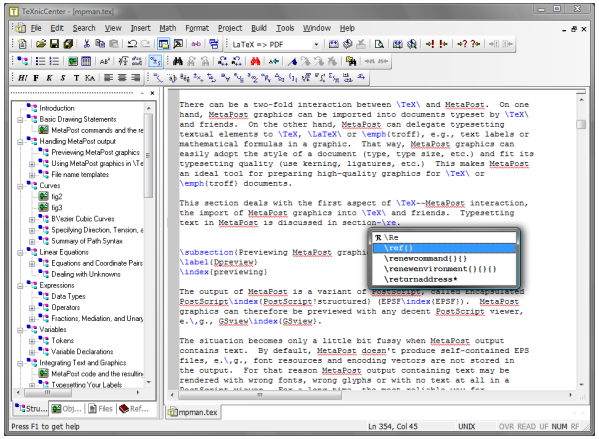
- **"All-in-one solution"** (TEXnicCenter)
- **"Middle solution"** (Kile, TEXmaker, ..)

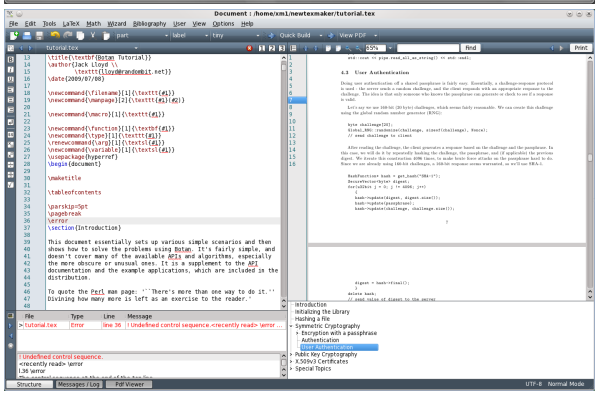


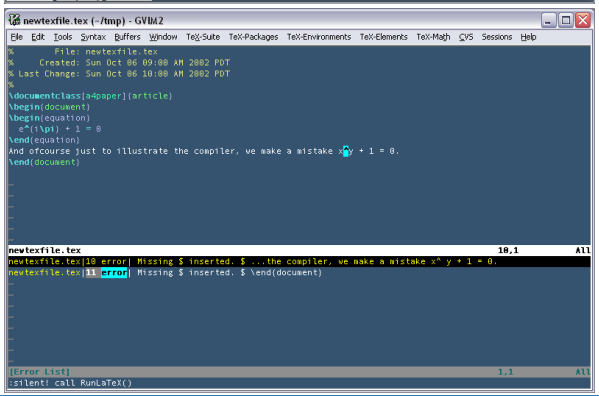
Choosing an editor

There are 3 options:

- **"All-in-one solution"** (TEXnicCenter)
- **"Middle solution"** (Kile, TEXmaker, ..)
- **"Raw solution"** (Notepad++, gedit, Emacs, VIM)







Installation

1. choose a distribution and install it
2. install ghostscript (to work with PostScript)
3. choose an editor and install it (or tell existing editor where to find the LaTeX distribution)
4. optional: install additional packages

Installation

1. choose a distribution and install it
2. install ghostscript (to work with PostScript)
3. choose an editor and install it (or tell existing editor where to find the LaTeX distribution)
4. optional: install additional packages

Installation

1. choose a distribution and install it
2. install ghostscript (to work with PostScript)
3. choose an editor and install it (or tell existing editor where to find the LaTeX distribution)
4. optional: install additional packages

Installation

1. choose a distribution and install it
2. install ghostscript (to work with PostScript)
3. choose an editor and install it (or tell existing editor where to find the LaTeX distribution)
4. optional: install additional packages

Document structure

documentclass[define and set options]{
  packages[load and set options]{
    document[actual content]{
      titelpage
      index and tables of figures, equations, etc..
      chapters
      bibliography[load database and set style options]
    }
  }
}

Compiling

- **In order to see the "coded content" from the editor in a "real output", the tex-file has to be compiled.**
- Options to choose from as output format: DVI, PS & PDF.
- Mostly used by scientists is probably direct output in PDF.
- This is done with the standard package pdflatex (or pdftex).

Generating output via commandline, type pdflatex filename.tex
Otherwise just press the "pdflatex"button in your workspace

- To include bibliography, you have to do 2x pdflatex filename, 1x bibtex filename, 1x pdflatex filename
- In most workspaces this is done automatically pressing "pdflatex"

Compiling

- **In order to see the "coded content" from the editor in a "real output", the tex-file has to be compiled.**
- Options to choose from as output format: DVI, PS & PDF.
- Mostly used by scientists is probably direct output in PDF.
- This is done with the standard package pdflatex (or pdftex).

Generating output via commandline, type pdflatex filename.tex
Otherwise just press the "pdflatex"button in your workspace

- To include bibliography, you have to do 2x pdflatex filename, 1x bibtex filename, 1x pdflatex filename
- In most workspaces this is done automatically pressing "pdflatex"

Compiling

- **In order to see the "coded content" from the editor in a "real output", the tex-file has to be compiled.**
- Options to choose from as output format: DVI, PS & PDF.
- Mostly used by scientists is probably direct output in PDF.
- This is done with the standard package pdflatex (or pdftex).

Generating output via commandline, type pdflatex filename.tex
Otherwise just press the "pdflatex"button in your workspace

- To include bibliography, you have to do 2x pdflatex filename, 1x bibtex filename, 1x pdflatex filename
- In most workspaces this is done automatically pressing "pdflatex"

LaTeX files

*.tex files with content and LaTeX commands
*.bib file containing database of bibliography entries
*.aux file with internal information about formatting and bibliography
*.bbl bibliography environment used for output file
*.bst information on style of the bibliography
*.sty style file with definitions of macros
*.pdf, *.ps, *.dvi final output

Syntax

1. **\command[how?]{what?}**

example

```
\documentclass[12pt]{article}
\includegraphics[width=10cm]{graphicfile.png}
```

2. **\begin{environment}**

   ...

   \end{environment}

example

```
\beginn{itemize}
\item argument1
\end{itemize}
```

Syntax

1. **\command[how?]{what?}**

example

```
\documentclass[12pt]{article}
\includegraphics[width=10cm]{graphicfile.png}
```

2. **\begin{environment}**

   ...

   \end{environment}

example

```
\beginn{itemize}
\item argument1
\end{itemize}
```

Syntax

1. \command[how?]{what?}

example

```
\documentclass[12pt]{article}
\includegraphics[width=10cm]{graphicfile.png}
```

2. \begin{environment}
   ...
   \end{environment}

example

```
\beginn{itemize}
\item argument1
\end{itemize}
```

Syntax

1. **command**[**how?**]{**what?**}

example

```
\documentclass[12pt]{article}
\includegraphics[width=10cm]{graphicfile.png}
```

2. **begin**{**environment**}
   ...
   end{**environment**}

example

```
\beginn{itemize}
\item argument1
\end{itemize}
```

Packages

- packages extend or add functionality to almost all thinkable worksteps
- many are included in the distributions and just have to be activated using \usepackage[options]{packagename}
- when they are not included, they can be manually installed or inside the workspace used

these packages should almost always be included in preample:
\usepackage[utf8]{inputenc}: improves and simplifies character input
\usepackage[T1]{fontenc}: improves output of font and hyphenation
\usepackage{lmodern}: better font visualization for pdf-putput
\usepackage[ngerman, english]{babel}: language settings
\usepackage{amsmath,amsfonts,amssymb}: math.equations & expressions

Packages

- packages extend or add functionality to almost all thinkable worksteps
- many are included in the distributions and just have to be activated using \usepackage[options]{packagename}
- when they are not included, they can be manually installed or inside the workspace used

these packages should almost always be included in preample:
\usepackage[utf8]{inputenc}: improves and simplifies character input
\usepackage[T1]{fontenc}: improves output of font and hyphenation
\usepackage{lmodern}: better font visualization for pdf-putput
\usepackage[ngerman, english]{babel}: language settings
\usepackage{amsmath,amsfonts,amssymb}: math.equations & expressions

Basic example

```
\documentclass[
12pt,
a4paper
]{scrreprt}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
\usepackage[ngerman]{babel}
\usepackage{amsmath,amsfonts,amssymb}
\begin{document}
This is the actual content.
\end{document}
```

Dividing a document

Sectioning of a document in LaTeX is quite intuitive
The chapter definition only exists in document classes books and reports

- \chapter[table of content]{text}
- \section{}
- \subsection{}
- \subsubsection{}
- \paragraph{}
- \subparagraph{}

Should the part not go into the table of content, mark it with an asterisk:
\section*{}

..text formatting

- \centering
- \begin{center}..\end{center}
- \textbf{exampletext}
- \textit{exampletext}
- {\LARGE exampletext}
- {\tiny exampletext}
- \newline or \\
- \clearpage
- \vspace{1cm}exampletext
- \hspace{2.5cm}exampletext

example

exampletext
exampletext

exampletext

exampletext

exampletext

exampletext

creates new line

starts a new page

exampletext

exampletext

..lists

- \begin{itemize}
  \item fact 1
  \item fact 2
  \end{itemize}
- \begin{enumerate}
  \item introduction
  \item conclusion
  \end{enumerate}
- \begin{description}
  \item[office] write paper
  \item[field] measurements
  \end{description}

example

- fact 1
- fact 2

1. introduction
2. conclusion

office write paper

field measurements

..titlepage

Two options to create a titlepage:

1. passing content to predefined structures, creating titlepage using **maketitle**
2. using the **titlepage environment** to create it completly on your own

   this way is much more individual but sometimes a try and error process with margins

example

```
\title{docuent title}
\author{Marvin Reich}
\date{01.01.2014}
\maketitle

\begin{titlepage}
design here your own page
\end{titlepage}
```

..titlepage

Two options to create a titlepage:

1. passing content to predefined structures, creating titlepage using **maketitle**
2. using the **titlepage environment** to create it completly on your own

   this way is much more individual but sometimes a try and error process with margins

example

```
\title{docuent title}
\author{Marvin Reich}
\date{01.01.2014}
\maketitle

\begin{titlepage}
design here your own page
\end{titlepage}
```

..list of contents, header/footer

to create the table of content, list of figures and list of tables, use

- **tableofcontents**
- **listoffigures**
- **listoftables**

to display header and footer, use **pagestyle**{**option**}

- \thispagestlye{option} only changes the current page
- use the **fancyhdr-package** for much more control and endless options

example

\pagestyle{headings}
header with section and page number info
\pagestyle{empty}
no footer or header
\thispagestyle{empty}
only the current page is left without header and footer
plain: no header but pagenumber in footer
myheadings: can be used to personalize the header

..list of contents, header/footer

to create the table of content, list of figures and list of tables, use

- **tableofcontents**
- **listoffigures**
- **listoftables**

to display header and footer, use **pagestyle**{**option**}

- \thispagestlye{option} only changes the current page
- use the **fancyhdr-package** for much more control and endless options

example

\pagestyle{headings}
header with section and page number info
\pagestyle{empty}
no footer or header
\thispagestyle{empty}
only the current page is left without header and footer
plain: no header but pagenumber in footer
myheadings: can be used to personalize the header

..graphics

- load the package **graphicx** in the preamble
- graphics are included with the command **\includegraphics[atr]{file}**
- some attributes: width, height, scale, angle..
- to label or for reference put the graphic in an figure environment
- when using pdflatex, only .png, .jpg and .pdf formats are supported

example

```
\usepackage{graphicx}
\includegraphics[.1\textwidth]{testP}


\begin{figure}
\includegraphics[scale=.2,
angle=-5]{testP}
\caption{this is a test screen.}
\end{figure}
```

..graphics

- load the package **graphicx** in the preamble
- graphics are included with the command **includegraphics[atr]**{**file**}
- some attributes: width, height, scale, angle..
- to label or for reference put the graphic in an figure environment
- when using pdflatex, only .png, .jpg and .pdf formats are supported

example

```
\usepackage{graphicx}
\includegraphics[.1\textwidth]{testP}
```

```
\begin{figure}
\includegraphics[scale=.2,
angle=-5]{testP}
\caption{this is a test screen.}
\end{figure}
```

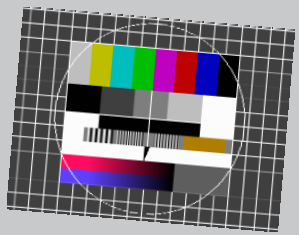


Figure: this is a test screen.

..equations

- inside textblocks, activate **math-mode** using $
- to list an equation, use the **equation environment**
- only equations listed in environments are counted and will be included in a table of equations

example

..as
$x= \frac{ \sqrt{time}}{faktor_{init}}$
equals..

..as $x = \frac{\sqrt{time}}{faktor_{init}}$ equals..

```
\begin{equation}
x= \frac{ \sqrt{time}}{faktor_{init}}
\label{equation1}
\end{equation}
```

$$x = \frac{\sqrt{time}}{faktor_{init}} \quad (1)$$

..equations

- inside textblocks, activate **math-mode** using $
- to list an equation, use the **equation environment**
- only equations listed in environments are counted and will be included in a table of equations

example

..as

```
$x= \frac{ \sqrt{time}}{faktor_{init}}$
```

equals..

..as $x = \frac{\sqrt{time}}{faktor_{init}}$ equals..

```
\begin{equation}
x= \frac{ \sqrt{time}}{faktor_{init}}
\label{equation1}
\end{equation}
```

$$x = \frac{\sqrt{time}}{faktor_{init}} \quad (1)$$

..tables

- tables are created with the **tabular environment**:
  \begin{tabular}[pos]{specs}
  ..
  \end{tabular}
- pos: b,c or t
- specs: alignment of each colum & vertical lines
- separate cells using &
- rows are ended with \\
- special table-packages can be used **inside** the tabular environment
- recommended: **booktabs, pgfplotstable**

- for labeling and referening "tabular" has to be put inside **\begin{table}..\end{table}**

example

```
\begin{tabular}[c]{c | l | r || r |}
cell1 & c2 & c3 & last cell \\\hline
r2 & left & right & 2 lines \\\hline
last &row 2 & 3 & 4 \\
\multicolumn{2}{c}{2cells}&3 & 4 \\
\end{tabular}
```

cell1	c2	c3	last cell
r2	left	right	2 lines
last row	2	3	4
2cells		3	4

..tables

- tables are created with the **tabular environment**:
  \begin{tabular}[pos]{specs}
  ..
  \end{tabular}
- pos: b,c or t
- specs: alignment of each colum & vertical lines
- separate cells using &
- rows are ended with \\
- special table-packages can be used **inside** the tabular environment
- recommended: **booktabs, pgfplotstable**

- for labeling and referening "tabular" has to be put inside **begin**{**table**}..**end**{**table**}

example

```
\begin{tabular}[c]{c | l | r || r |}
cell1 & c2 & c3 & last cell \\\hline
r2 & left & right & 2 lines \\\hline
last &row 2 & 3 & 4 \\
\multicolumn{2}{c}{2cells}&3 & 4 \\
\end{tabular}
```

cell1	c2	c3	last cell
r2	left	right	2 lines
last row	2	3	4
2cells		3	4

..labeling and referencing

- **label**{**labelname**}
  should be used inside of an environment (table, figure, etc.)
- **caption**{**captiontext**}
  should be used inside of an environment (table, figure, etc.)
- standard LaTeX
  ref{**labelname**}
- better: **cleveref package**
  automatically detects type of reference
  cref{**labelname**}
- using the **hyperref package**
  references and index can be generated as links

example

```
\begin{figure}
\includegraphics{picturefile}
\caption{This is picture 445.}
\label{pic445}
\end{figure}

\ref{pic445}
2.1
\cref{pic445}
fig. 2.1
```

..labeling and referencing

- **label**{**labelname**}
  should be used inside of an environment (table, figure, etc.)
- **caption**{**captiontext**}
  should be used inside of an environment (table, figure, etc.)
- standard LaTeX
  ref{**labelname**}
- better: **cleveref package**
  automatically detects type of reference
  cref{**labelname**}
- using the **hyperref package**
  references and index can be generated as links

example

```
\begin{figure}
\includegraphics{picturefile}
\caption{This is picture 445.}
\label{pic445}
\end{figure}

\ref{pic445}
2.1
\cref{pic445}
fig. 2.1
```

..minipage

- A minipage, like the names says, creates a page inside the page
- it is often used when liking two objects with a distinct idea of visualization (sort of forcing)
- very handy for presentations and posters!

example

```
\begin{minipage}[c][2cm][b]{2cm}
text in minipage one (left)
\end{minipage}
\begin{minipage}[c]{0.65\textwidth}
\includegraphics[height=2cm]{testbild}
\end{minipage}
```

text in mini-page one (left)

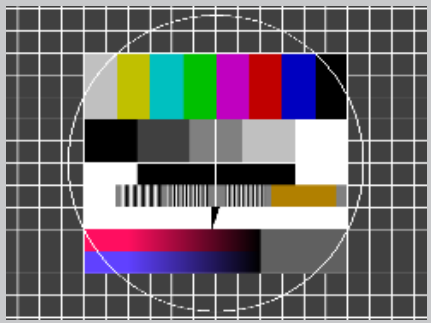
..minipage

- A minipage, like the names says, creates a page inside the page
- it is often used when liking two objects with a distinct idea of visualization (sort of forcing)
- very handy for presentations and posters!

example

```
\begin{minipage}[c][2cm][b]{2cm}
text in minipage one (left)
\end{minipage}
\begin{minipage}[c]{0.65\textwidth}
\includegraphics[height=2cm]{testbild}
\end{minipage}
```

text in minipage one (left)



..bibliography & citing

- including literature database most easy in **.bibtex format**
- sources cited in the document are automatically shown in the bibliography section

- standard LaTeX citing: **cite**{**source**}
- using the **natbib package** there are endless **bibliography-section styles** and numerous different **citing commands**!!

example

```
\bibliographystyle{bibstyle.sty}
\bibliography{databasefile}
```

```
\usepackge[square]{natbib}
```

```
..stated by \citet{mueller2013}, the..
..stated by Mueller et. al. [2013], the..
```

```
..used before \citep{mueller2013}.
..used before [Mueller et. al., 2013].
```

..bibliography & citing

- including literature database most easy in **.bibtex format**
- sources cited in the document are automatically shown in the bibliography section

- standard LaTeX citing: **cite**{**source**}
- using the **natbib package** there are endless **bibliography-section styles** and numerous different **citing commands**!!

example

```
\bibliographystyle{bibstyle.sty}
\bibliography{databasefile}
```

```
\usepackge[square]{natbib}
```

..stated by \citet{mueller2013}, the..
..stated by Mueller et. al. [2013], the..

..used before \citep{mueller2013}.
..used before [Mueller et. al., 2013].

Now let's get down to some practice..!!

→ create an latex document containing:

- titlepage
- table of content
- various chapters / sections
- a list
- an equation
- a figure
- a table
- a citation
- bibliography

tips & tricks

- alt + 6 compliles into pdf
- alt + 7 shows you the pdf
- F11 shows line numbers (find errors quicker)
- tab autocompletes commands

Now let's get down to some practice..!!

→ create an latex document containing:

- titlepage
- table of content
- various chapters / sections
- a list
- an equation
- a figure
- a table
- a citation
- bibliography

tips & tricks

- alt + 6 compliles into pdf
- alt + 7 shows you the pdf
- F11 shows line numbers (find errors quicker)
- tab autocompletes commands

General

- don't think too much about formatting but let LaTeX do the job for you!
- use the manuals of the packages
- try out some tables easily using http://www.tablesgenerator.com/
- comment your own code!!
- if you are stuck, minimize your own example
- if you are still stuck: search the net, you are probably not the first with that issue!

Useful packages

wrapfig: wrapping text around your figures; forcing figures to stick
subfigure: merging various graphics into one, with subcaption for each
ctable: different table environment based on booktabs
fancyhdr: greatly extends visual options for footer and header
pdflscape: enables turning around of single pages
float: alternative ways to wrapfig of letting graphics float in the text
appendix: good management of appendix
tcolorbox: create beautiful and endless creative boxes
tikz: drawn almost anything (a pretty much standard)
tables: R package for outputting tables in LaTeX format

Useful commands

scalebox: scales content to a certain percentage
include: includes a tex-file at the position of the command
renewcommand: modifies existing commands
newcommand: creates own commands with endless possibilities
newenvironment: creates own environments with endless possibilities
definecolor: defines own colors
multicols: creates an environment, divididing its content into n colums
graphicspath: sets standard folder where to look for graphic files

Where to look for help..

http://latex-project.org
http://tex.stackexchange.com/
http://www.ctan.org/
http://www.golatex.de/
http://www.sharelatex.com/
http://en.wikibooks.org/wiki/LaTeX/
http://startpage.com (instead of google!)

Thank you for your attention

..and enjoy creating your own individual documents!!