SKILL PILLS

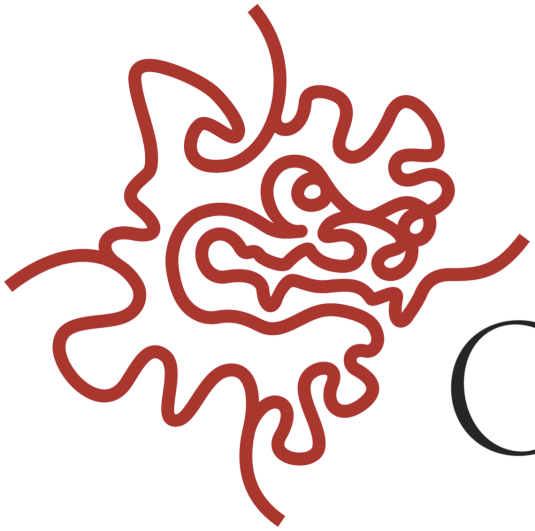
Skill Pill: LaTeX Course

Lecture 2: Mathematical Typesetting, Tables, Figures

Alex Martyushev

Okinawa Institute of Science and Technology

November 07, 2018



OIST

Equations

One of the great advantages of LaTeX is the typesetting of mathematical expressions.
Equations can be part of the text (*inline mode*), such as $\mathrm{e}^{\mathrm{i}\pi} + 1 = 0$, or they can be put on different lines which are not the part of text (*display math mode*) like this :

$$\mathrm{e}^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} \, .$$

Equations

One of the great advantages of LaTeX is the typesetting of mathematical expressions.
Equations can be part of the text (*inline mode*), such as $\mathrm{e}^{\mathrm{i}\pi} + 1 = 0$, or they can be put on different lines which are not the part of text (*display math mode*) like this :

$$\mathrm{e}^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}\,.$$

Equations in the text are created with `$ ... $` or `\( ... \)`, while separate equations with
`\[ ... \]` or `$$ ... $$` or `\begin{equation} ... \end{equation}`.

Equations

One of the great advantages of LaTeX is the typesetting of mathematical expressions.
Equations can be part of the text (*inline mode*), such as $\mathrm{e}^{\mathrm{i}\pi} + 1 = 0$, or they can be put on different lines which are not the part of text (*display math mode*) like this :

$$\mathrm{e}^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}\,.$$

Equations in the text are created with `$ ... $` or `\( ... \)`, while separate equations with
`\[ ... \]` or `$$ ... $$` or `\begin{equation} ... \end{equation}`.

It is recommended to use the $\mathcal{AMS}$-packages, since those extend the whole equation typesetting enormously. Those packages are included via

```
\usepackage{amsmath, amssymb}
```

.

Equations

- Letters, numbers and symbols such as `+ - * / = < > , . ( ) [ ]` may be directly used in math mode (*except the special ones*).
  For many other symbols one needs to use special commands (see cheat sheet), e. g. for multiplication:

  `2 \cdot 3=6` creates $2 \cdot 3 = 6$

Equations

- Letters, numbers and symbols such as `+ - * / = < > , . ( ) [ ]` may be directly used in math mode (*except the special ones*).
  For many other symbols one needs to use special commands (see cheat sheet), e. g. for multiplication:

  `2 \cdot 3=6` creates $2 \cdot 3 = 6$

- Fractions are created with `\frac{}{}`:

  `\frac{x+y}{2}` creates $\frac{x+y}{2}$

Equations

- Letters, numbers and symbols such as `+ - * / = < > , . ( ) [ ]` may be directly used in math mode (*except the special ones*).
  For many other symbols one needs to use special commands (see cheat sheet), e. g. for multiplication:

  `2 \cdot 3=6` creates $2 \cdot 3 = 6$

- Fractions are created with `\frac{}{}`:

  `\frac{x+y}{2}` creates $\frac{x+y}{2}$

- Square roots are created using `\sqrt{}` and n^{th} roots using `\sqrt[n]{}`:

  `\sqrt{abc}` creates $\sqrt{abc}$

  `\sqrt[3]{abc}` creates $\sqrt[3]{abc}$

Superscripts and Subscripts

- For superscript use `^`: `$x^2$`, which creates x^2.
- For subscript use `_`: `$(x_n)_n$`, which creates $(x_n)_n$.

Superscripts and Subscripts

- For superscript use `^`: `$x^2$`, which creates x^2.
- For subscript use `_`: `$(x_n)_n$`, which creates $(x_n)_n$.
- If more than one character needs to be set super- or subscript, one needs to use braces:

`X_12`	creates	X_12
`X_{12}`	creates	X_{12}

Superscripts and Subscripts

- For superscript use `^`: `$x^2$`, which creates x^2.
- For subscript use `_`: `$(x_n)_n$`, which creates $(x_n)_n$.
- If more than one character needs to be set super- or subscript, one needs to use braces:

`X_12`	creates	X_12
`X_{12}`	creates	X_{12}

- Super- and subscript can be combined and nested:

`X^3_{n_i^2}` creates $X^3_{n_i^2}$

Brackets and delimiters

- Brackets (or delimiters) which are created with `\left` and `\right` are adopted to the heights of the content:

  `( \frac{x+y}{2} )` creates $(\frac{x+y}{2})$

  `\left( \frac{x+y}{2} \right)` creates $\left(\frac{x+y}{2} \right)$

  Brackets can be combined in any way:

  `\left[ ... \right)`

Brackets and delimiters

- Brackets (or delimiters) which are created with `\left` and `\right` are adopted to the heights of the content:

  `( \frac{x+y}{2} )` creates $(\frac{x+y}{2})$

  `\left( \frac{x+y}{2} \right)` creates $\left(\frac{x+y}{2} \right)$

  Brackets can be combined in any way:

  `\left[ ... \right)`

- The dot represents (if used with `\left` or `\right`) an invisible bracket:

  `\left( ... \right.`

- Manually we can resize the brackets using size commands:

  `\big( \Big( \bigg( \Bigg(` creates $\big(\Big(\bigg(\Bigg($

Operators

- Variables are set cursive, while functions are set upright, e. g. `$\sin(x)$` creates $\sin(x)$

- Limits of summation/integration (and similar structures) are created with `\int_{lower}^{upper}`:

  `\int_{i=1}^n` creates $\int_{i=1}^{n}$

  `\int \limits_{i=1}^n` creates $\int\limits_{i=1}^{n}$

Multiple Equations

- The amsmath-packages provides additional environments in order to enumerate and/or vertically align equations:

```
\begin{align}
    \partial_\mu F^{\mu\nu} &= j^\nu \\
    \partial_{[\mu}F_{\nu\lambda]} &= 0
\end{align}
```

$$\partial_\mu F^{\mu\nu} = j^\nu \tag{1}$$
$$\partial_\mu F_{\nu\lambda} = 0 \tag{2}$$

The & symbols indicate where to align the equations.

- There are many options with slightly different behaviour such as: `\begin{eqnarray}...\end{eqnarray}` and `\begin{IEEEeqnarray}...\end{IEEEeqnarray}`

Long Equations

- For equations longer than a line use the multline environment. Insert a double backslash to set a point for the equation to be broken.

```
\begin{multline*}
p(x) = 3x^6 + 14x^5y + 590x^4y^2 + 19x^3y^3\\
- 12x^2y^4 - 12xy^5 + 2y^6 - a^3b^3
\end{multline*}
```

$$p(x) = 3x^6 + 14x^5y + 590x^4y^2 + 19x^3y^3 \\ - 12x^2y^4 - 12xy^5 + 2y^6 - a^3b^3$$

Exercise

Time for an exercise...!!!!

$$\frac{\partial^2 f}{\partial x^2} = \int_0^\infty \vec{\nabla} \cdot \vec{A}$$

Exercise

Time for an exercise...!!!!

$$\frac{\partial^2 f}{\partial x^2} = \int_0^\infty \vec{\nabla} \cdot \vec{A}$$

ANSWER

```
\frac{\partial^2 f}{\partial x^2} = \int_{0}^{\infty}
\vec{\nabla} \cdot \vec{A}
```

The Comprehensive LaTeX Symbol List

A detailed list about symbols and the corresponding packages may be found at the following link:

`http://ftp.jaist.ac.jp/pub/CTAN/info/symbols/comprehensive/symbols-a4.pdf`

This online tool can help us find a symbol by just drawing it:
`http://detexify.kirelabs.org/classify.html`

Arrays in Math Mode

To typeset arrays use the `\array` environment in math mode.

```
\left(
\begin{array}{ccc}
x_1 & x_2 &  x_3\\
x_4 & x_5 &  x_6 \\
1 & 2 & 3
\end{array}
\right)
```

$$\left(\begin{array}{ccc} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \\ 1 & 2 & 3 \end{array} \right)$$

Here $\&$ jumps to the next column and `\\` starts a new line .
Aligning options are `l`, `c`, `r`.

Matrices in Math Mode

Moreover the amsmath package offers various tools for the formatting of vectors and matrices.

```
A = \begin{pmatrix}
    1 & 2\\
    3 & 4
\end{pmatrix}
```

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

```
B=\begin{bmatrix}
    1 & 2\\
    3 & 4
\end{bmatrix}
```

$$B = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

```
C=\begin{Bmatrix}
    1 & 2\\
    3 & 4
\end{Bmatrix}
```

$$C = \begin{Bmatrix} 1 & 2 \\ 3 & 4 \end{Bmatrix}$$

Matrices in Math Mode

- In case you do not need brackets or you want to set them yourself with `\left ... \right` you can use the `matrix` environment:

```
\begin{matrix}
    1 & 2\\
    3 & 4
\end{matrix}
```

$$\begin{matrix} 1 & 2 \\ 3 & 4 \end{matrix}$$

Matrices in Math Mode

- In case you do not need brackets or you want to set them yourself with `\left ... \right` you can use the `matrix` environment:

```
\begin{matrix}
    1 & 2\\
    3 & 4
\end{matrix}
```

$$\begin{matrix} 1 & 2 \\ 3 & 4 \end{matrix}$$

- For smaller matrices in-line you can use the `smallmatrix` environment.

```
\begin{smallmatrix}
    1 & 2\\
    3 & 4
\end{smallmatrix}
```

$\begin{smallmatrix} 1 & 2 \\ 3 & 4 \end{smallmatrix}$

Cases in math mode

- For case definitions you can use the cases environment:

```
\Theta(x) =
\begin{cases}
    0, & \text{if} x<0, \\
    1, & \text{else} x\geq 0.
\end{cases}
```

$$\Theta(x) = \begin{cases} 0, & \text{if } x < 0, \\ 1, & \text{else } x \geq 0. \end{cases}$$

The tabular-Environment

The tabular-environment is used to create tables.

The tabular-Environment

The tabular-environment is used to create tables.

```
\begin{tabular}{l|cr}
    flush left & centered & flush right\\
    \hline
    text & more text & even more text
\end{tabular}
```

flush left	centered	flush right
text	more text	even more text

The tabular-Environment

```
\begin{tabular}{l|cr}
    flush left & centered & flush right\\
    \hline
    text & more text & even more text
\end{tabular}
```

The tabular-Environment

```
\begin{tabular}{l|cr}
    flush left & centered & flush right\\
    \hline
    text & more text & even more text
\end{tabular}
```

- `l` left aligned column (as wide as the widest entry)
- `c` centered column
- `r` right aligned column
- `p{1cm}` justified (with the value of the column width)
- `|` division of the columns with a vertical line (more than one possible)

The tabular-Environment

```
\begin{tabular}{l|cr}
    flush left & centered & flush right\\
    \hline
    text & more text & even more text
\end{tabular}
```

`&` separates the columns

`\\` separates the rows

`\hline` horizontal line over the whole width of the table (more than one possible)

`\cline{1-2}` horizontal line from column 1 to column 2

The \multicolumn command

The \multicolumn command connects the entries of several columns of one row.

```
\multicolumn{Number}{Alignment}{Content}
```

The \multicolumn command

The \multicolumn command connects the entries of several columns of one row.

```
\multicolumn{Number}{Alignment}{Content}
```

Number	Number of columns to be connected.
Alignment	Alignment of text in the created cell (`l`, `c`, `r` as needed, with succeeding `|`)
Content	Content of the created cell.

The \multicolumn command – Example

```
\begin{tabular}{|l|l|l|}
    \hline
    Atoms & \multicolumn{2}{c|}{Ions} \\
    & Cations & Anions \\
    \hline
    H, Ne & Na$^+$ & Cl$^-$ \\
    \hline
\end{tabular}
```

Atoms	Ions	
	Cations	Anions
H, Ne	Na^+	Cl^-

The \multirow Command

Similar to the \multicolumn command, the \multirow command is used to connect the entries of several rows of one column. In order to use it you need to add the multirow package:

```
\usepackage{multirow}
```

The syntax is simply:

```
\multirow{Number}{Width}{Content}
```

The \multirow Command

Similar to the \multicolumn command, the \multirow command is used to connect the entries of several rows of one column. In order to use it you need to add the multirow package:

```
\usepackage{multirow}
```

The syntax is simply:

```
\multirow{Number}{Width}{Content}
```

- **Number** Number of columns to be connected.
- **Width** Within of the box in which the text will be set. (* automatically adjusts the width to the column-width)
- **Content** Content of the created cell.

The \multirow Command – Example

```
\begin{tabular}{|l|l|l|}
    \hline
    \multirow{2}{*}{Atoms} & \multicolumn{2}{c|}{Ions} \\
    \cline{2-3} & Cations & Anions \\
    \hline
    H, Ne & Na$^+$ & Cl$^-$ \\
    \hline
\end{tabular}
```

Atoms	Ions	
	Cations	Anions
H, Ne	Na^+	Cl^-

The tabular-Environment – Exercise

Nature

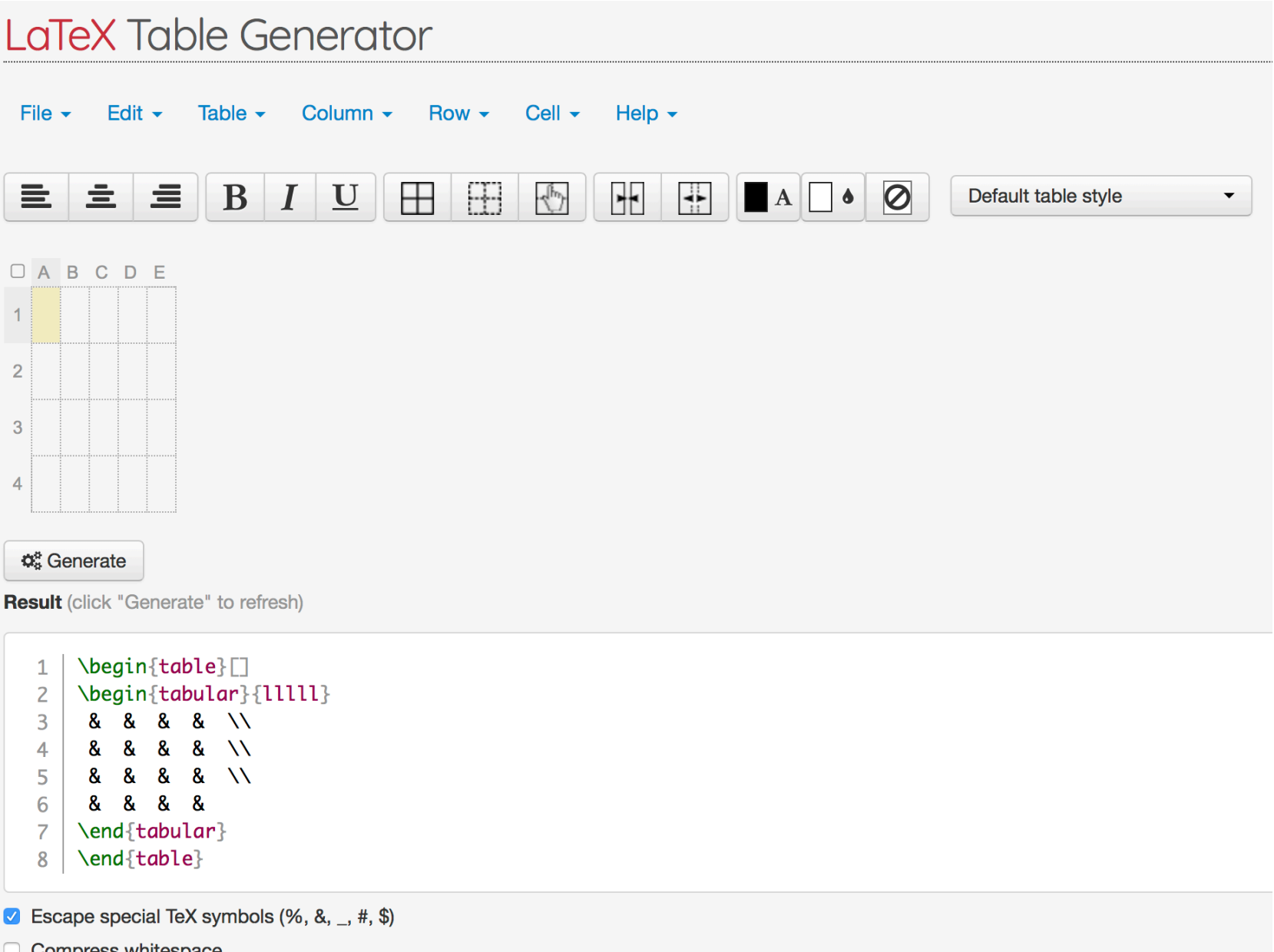
Nature amazes us with its wonders!

Mountains	Rivers	Lakes	Beaches and Deserts
rough	fast	standing	sandy

The table was centered with the `center` environment. (\begin{center} ... \end{center})

The table generator

https:\\www.tablesgenerator.com



The graphicx Package

To include pictures you can use the graphicx packages. Supported formats are .jpg, .png, .pdf, and .eps (using epstopdf).

The graphicx Package

To include pictures you can use the graphicx packages. Supported formats are .jpg, .png, .pdf, and .eps (using epstopdf).

The syntax is:

```
\includegraphics[arguments]{filename.jpg}
```

Optional Arguments for `\includegraphics`

Size related arguments:

`width`	Width, e. g. `width=10cm`
`height`	Height, e. g. `height=5cm`
`scale`	Scaling, e. g. `scale=0.5`

Optional Arguments for `\includegraphics`

Size related arguments:

`width`	Width, e. g. `width=10cm`
`height`	Height, e. g. `height=5cm`
`scale`	Scaling, e. g. `scale=0.5`

If only `width` **or** `height` is set, the size ratio is kept. Alternatively you can use the `scale` argument.

Optional Arguments for `\includegraphics`

Size related arguments:

`width`	Width, e. g. `width=10cm`
`height`	Height, e. g. `height=5cm`
`scale`	Scaling, e. g. `scale=0.5`

If only `width` **or** `height` is set, the size ratio is kept. Alternatively you can use the `scale` argument.

Example:

```
\includegraphics[width=8cm]{figure.jpg}
```

Optional Arguments for `\includegraphics`

Cropping arguments:

`clip` Trim figure by lengths defined by `trim`

`trim=l u r o` Lengths by which the figure is trimmed in the left (`l`), down (`u`), right (`r`) and up (`o`) directions.

Optional Arguments for `\includegraphics`

Cropping arguments:

`clip`	Trim figure by lengths defined by `trim`
`trim=l u r o`	Lengths by which the figure is trimmed in the left (`l`), down (`u`), right (`r`) and up (`o`) directions.

Example:

```
\includegraphics[clip, trim=1cm 2.5cm 1cm 0cm]{figure.pdf}
```

Float Environments

It is useful to place tables and figures in a float environment for the following advantages:

- 'optimized' placing
- referencing
- simple labeling
- automatic listing of figures and tables

Float Environments

It is useful to place tables and figures in a float environment for the following advantages:

- 'optimized' placing
- referencing
- simple labeling
- automatic listing of figures and tables

For figures you can use the `figure` environment and for tables you can use the `table` environment.

```
\begin{figure}[placing] ... \end{figure}

\begin{table}[placing] ... \end{table}
```

Content of Float Environments

In general you can place any object within a float environment. A typical application is given below:

```
\begin{figure}
    \centering
    \includegraphics[width=10cm]{sombrero.jpg}
    \caption{The Sombrero Galaxy (M104).}
\end{figure}
```

Content of Float Environments

In general you can place any object within a float environment. A typical application is given below:

```
\begin{figure}
    \centering
    \includegraphics[width=10cm]{sombrero.jpg}
    \caption{The Sombrero Galaxy (M104).}
\end{figure}
```

`\centering` Centers the content of the environment.

`\caption{}` Includes a caption with successive numbering. It can be placed either above or below the object.

Outcome

Figure 1: The Sombrero Galaxy (M104).

Image credits: NASA and The Hubble Heritage Team (STScI/AURA)

Positioning

Positioning Arguments:

```
\begin{figure}[!ht]
    \centering
    \includegraphics[width=10cm]{sombrero.jpg}
    \caption{The Sombrero Galaxy (M104).}
\end{figure}
```

Positioning

Positioning Arguments:

```
\begin{figure}[!ht]
    \centering
    \includegraphics[width=10cm]{sombrero.jpg}
    \caption{The Sombrero Galaxy (M104).}
\end{figure}
```

- `h` „‘here’ – The floating object is placed at the same location as in the source code.
- `t` „‘top’ – The floating object is placed at the top of the page.
- `b` The floating object is placed at the bottom of the page.
- `p` „‘page’ – The floating object is placed on its own page.
- `!` The floating object is prioritized.

Lists of Figures and Tables

The lists of figures and tables can be created using `\listoffigures` and `\listoftables` respectively. The entries are the captions.

Lists of Figures and Tables

The lists of figures and tables can be created using `\listoffigures` and `\listoftables` respectively. The entries are the captions.

If you want to use a customized caption (e. g. if the original caption is to long) you can do this with an optional argument of the `\caption` command.

```
\caption[short caption]{long caption}
```