

# Introduction to the Typesetting System $\text{\LaTeX}$

## day one

Moritz Brinkmann  
mail@latexkurs.de

March 2024

# Learning Objectives

After the two workshop days, you will be able to:

- create simple documents in  $\text{\LaTeX}$
- find assistance in class and package documentation
- create multilingual documents
- incorporate images and create tables
- generate reference lists
- typeset mathematical formulas
- structure larger projects

## Dates

- Two sessions per date:
    - Saturday, February 17
    - Sunday, February 18
  - Each session from 10 am to 3 pm
  - Approximately 45 minutes break
- or*
- Saturday, March 2
  - Sunday, March 3

## Materials

- All materials are available on the workshop website or in ILIAS for download:

<https://ma.latexkurs.de/>



# Organizational Matters

## Exercises

- Theory and practical phases alternate.
- You are allowed (and encouraged) to try examples *at any time* on your computer.
- Feel free to experiment with something new *immediately!*
- Don't hesitate to ask if you're having trouble with anything.
- If you are using Overleaf, you can share your source code with `overleaf@latexkurs.de` when you have questions.

## $\text{\LaTeX}$ Flavor

The content of this course is based on the (relatively modern) variant Lua $\text{\LaTeX}$ .

# Content I

- ① What is it all about?
- ② Basic Operation
  - Classes and Packages
  - Basic Commands
- ③ Typographic Basics
  - Macrotypography
  - Microtypography
- ④ Documentation & Error Messages
  - Documentation
  - Error Messages
- ⑤ Languages
- ⑥ Floating Objects
  - Float Environments
  - Graphics

## 7 Bibliographies

biblatex

Managing References

## 8 Math Typesetting

## 9 Tables

Beautiful Tables

Automatic Column Width

## 10 Larger Projects

## 11 Charts

Part I

# The Name of the Game

# The Name of the Game

- Program  $\text{\TeX}$  (Since 1977)  
Written by Donald E. Knuth for his book "The Art of Computer Programming".  
"TeX" from Greek τέχνη
- Macro package  $\text{\plainTeX}$   
Makes  $\text{\TeX}$  usable for regular users.
- Extended macro package  $\text{\LaTeX}$  (Early 1980s)  
By Leslie Lamport: "Lamport's  $\text{\TeX}$ ".  
Many simplifications for the average user.
- Current stable version:  $\text{\LaTeX 2}_{\epsilon}$  (1994)  
"in an  $\epsilon$ -environment of 2"
- Future development:  $\text{\LaTeX 3}$   
not yet independently available, but as a package `expl3` in  $\text{\LaTeX 2}_{\epsilon}$



# What is T<sub>E</sub>X – and what is it not?

## L<sup>A</sup>T<sub>E</sub>X is well-suited for...

- All documents with a logical structure
  - Scientific papers (excellent mathematical typesetting)
  - Humanities papers (excellent multilingual support, bibliography creation, apparatus creation, etc.)
  - Articles, bachelor's theses, dissertations, etc.
  - Book series, letters
  - Presentations
- Much "abuse" by creative package authors

## L<sup>A</sup>T<sub>E</sub>X is less suitable for...

- Documents without a logical structure
  - Presentations (colorful, rotating, blinking, "chaotic")
  - Flyers
  - Posters

# How does T<sub>E</sub>X work?

- WYSIWYM
- Plain text files
- No hidden settings
- Text formatting using special commands:
  - "I want to write an article!"
  - "Create a heading!"
  - "Make the following bold!"
  - "Create a table that..."

# How does T<sub>E</sub>X work?

## Advantages

- Stability and portability
- Small file sizes
- Editable with any text editor
- Text files are always readable
- Consistent output everywhere

## Disadvantages

- Result not immediately visible
- Non-intuitive interface
- Steep learning curve
- Changes require recompilation
- Complex layout desires are hard to achieve

# A simple T<sub>E</sub>X document

How can text be distinguished from commands?

Approach in *classical* programming languages:

```
print ( "Hello, World!" );
```

⇒ unsuitable for a typesetting program

# A simple T<sub>E</sub>X document

- T<sub>E</sub>X is a markup language
- Individual characters have special meanings
- Backslash (\) serves as an escape character and marks the beginning of a command:  
`\chapter \section \author`

Simplest T<sub>E</sub>X document:

```
Hello, World! \bye
```

```
$ tex document.tex
```

creates a .dvi document and a .log file

# A simple $\text{\LaTeX}$ document

```
\documentclass{minimal}  
\begin{document}  
Hello, World!  
\end{document}
```

Hello, World!

## Assignment

Create your first  $\text{\LaTeX}$  document by typing this minimal example in your editor!

# Command Characters

<code>\</code>	<i>escape character</i> , marks the beginning of commands
<code>{}</code>	<i>grouping character</i> , groups related characters together e.g., arguments <code>\textbf{bold}</code>
<code>\$</code>	<i>math character</i> , starts and ends math mode
<code>&amp;</code>	<i>tabbing character</i> , separates columns in tables
<code>%</code>	<i>comment character</i> , comments out the rest of the line
<code>^_~#</code>	other characters with special meaning

Part II

# Basic Operation



# Document Classes

Document classes define fundamental properties of the document:

- Layout
- Default fonts
- Page layout
- Sectioning commands
- Appearance of lists, tables, enumerations, etc.

Properties can be customized by changing options or loading packages.

# Document Classes

## Standard Classes

article	(Short) articles
report	Reports, conference reports
book	Books
letter	Letters
minimal	For minimal examples

## KOMA-Script

scrartcl	Extension of article
scrreprt	Extension of report
scrbook	Extension of book
srlttr2	Very powerful letter class

## Special Classes

beamer	For presentations
tikzposter	Scientific posters

# Packages

- Packages provide additional functionality
- Work facilitators
- Error corrections
- Include in the preamble using `\usepackage[option(s)]{package name}`:

```
\documentclass{article}
\usepackage{
  amsmath,
  hyperref,
}
\usepackage[left=2cm]{geometry}
```

# Sectioning Commands

- Sectioning structures organize documents
- Enable automatic numbering, entry in tables of contents, column titles, etc.
- Defined by the document class
- Basic structure defined in the kernel

⇒ certain elements always available

```
\teil{Part I}  
\chapter{Chapter}  
\section{Section}  
\subsection{Subsection}  
\subsubsection{Subsubsection}  
\paragraph{Paragraph}  
\subparagraph{Subparagraph}
```

*try this in Overleaf*



<http://qn3.de/tex01>

# Basic Commands

## General

<code>\textrm{Serif}</code>	Serif <i>Abcdxyz</i>
<code>\textit{italic}</code>	<i>italic</i> <i>Abcdxyz</i>
<code>\textsl{slanted}</code>	<i>slanted</i> <i>Abcdxyz</i>
<code>\textsf{sans-serif}</code>	sans-serif <i>Abcdxyz</i>
<code>\textbf{bold}</code>	<b>bold</b> <b>Abcdxyz</b>
<code>\texttt{typewriter}</code>	typewriter <i>Abcdxyz</i>
<code>\textsc{Small Caps}</code>	SMALL CAPS    ABCDXYZ
<code>\emph{emphasis}</code>	<i>emphasis</i> <i>Abcdxyz</i>
<code>\\</code>	Line break
<code>\par</code> or blank line	Paragraph break
<code>\$E = \frac{p^2}{2m}\$</code>	Inline math mode: $E = \frac{p^2}{2m}$
<code>\[E = \frac{p^2}{2m}\]</code>	Display math mode: $E = \frac{p^2}{2m}$
<code>\tableofcontents</code>	Produces table of contents
<code>\today</code>	Current date



# Basic Commands

## Font Sizes

<code>\tiny</code>	tiny
<code>\small</code>	small
<code>\normalsize</code>	normal
<code>\large</code>	large
<code>\Large</code>	larger
<code>\LARGE</code>	even larger
<code>\huge</code>	huge
<code>\Huge</code>	even huge



# Auxiliary Files

## **Input**

.tex      T<sub>E</sub>X file with document text

## **Output**

.pdf      pdfT<sub>E</sub>X output or conversion from (x)dvi

## **Auxiliary files (write only)**

.log      Log file with information, warnings, errors

## **Auxiliary files (read and write)**

.aux      Auxiliary file with temporary information

.toc      table of contents

.lof      list of figures

.synctex.gz      required for the SyncT<sub>E</sub>X feature

⋮      ⋮

Part III

# Basics of Typography



# Macrotypography

- Text Area
- Header and Footer
- Font Selection
- Formatting Spacing
- Appearance of Index, Footnotes, etc.

## Assignment

Download the file `exercise_layout.tex` from the workshop website. Apply all typographic settings discussed to this file step by step.

Ideally, choose values that meet the requirements for your upcoming thesis.



## “Style

Format one-sided DIN A4

Font Size 12 pt

Line Spread 1.5 pt

Alignment justified (“Blocksatz”)

Left and right margin 3 cm

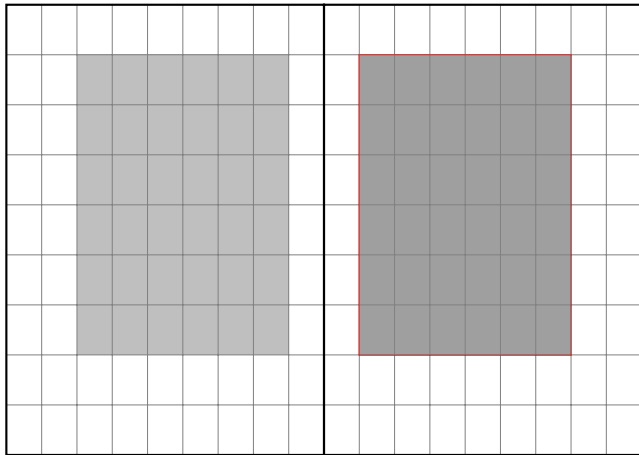
*Guidelines for Bachelor theses*

”

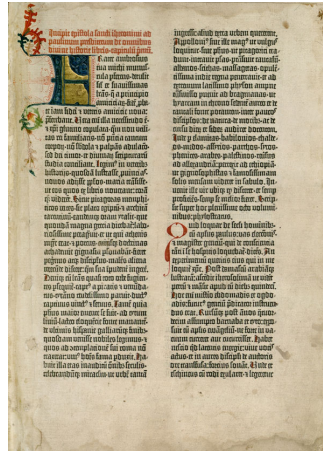
The text area refers to the portion of the page covered by the text (as opposed to the margins).

- Single or double-sided layout?
- Font size, line width,
- Header and footer
- Text columns

# Modern Text Area Construction



# Text Area in Gutenberg's Printing



# Text Area with KOMA-Script

- KOMA-Script provides optimal text area construction through its own package `typearea`.
- Adjustment is usually necessary only for exceptionally wide or narrow fonts:  
Option `DIV=⟨Factor⟩`  
Automatic calculation based on page size: `DIV=calc`  
Calculation based on medieval book page canon: `DIV=classic`
- Binding correction using option `BCOR=⟨Length⟩`

```
\documentclass[DIV=9, BCOR=12mm]{scrbook}
```

For non-KOMA classes, `typearea` must be loaded directly:

```
\usepackage[DIV=13, BCOR=2cm]{typearea}
```

# Text Area with geometry

Package geometry allows manual adjustment of the text area:

```
\usepackage[top=2cm, bottom=5cm]{geometry}
```

or:

```
\usepackage{geometry}  
\geometry{top=2cm, bottom=5cm}
```

# Text Area with geometry

## Possible Options

paper

left, right, inner, outer, hmargin

top, bottom, vmargin

margin

bindingoffset, textwidth, textheight

twocolumn, columnsep, marginparsep, footnotesep

headsep, footsep, nofoot, nohead

hoffset, voffset, offset

includehead, includefoot



# Line Spacing

Package `setspace` allows adjustment of line spacing:

```
\usepackage{setspace}
\singlespacing
\onehalfspacing
\doublespacing
```

Spacing in footnotes, etc., remains the same.

Fine-tuning: `\setstretch{Factor}`

# Headers and Footers

- Headers and footers contain important information about the document
  - Live column titles
  - Page numbers
- Adjustment using various packages
- Selection via `\pagestyle{<page style>}` or `\thispagestyle{<page style>}`
- Default settings: empty, plain, headings

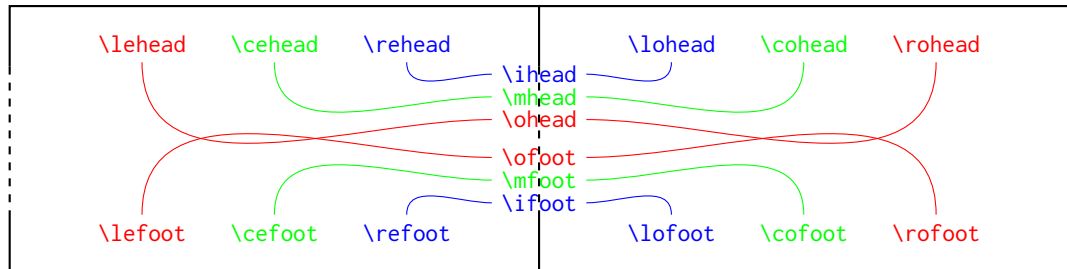


# Headers and Footers with scrlayer-scrpage

The package defines two page styles: `scrheadings` and `screadings.plain`

Adjustment is done through, e. g.

```
\lehead[<content plain.scrheadings>][<content scrheadings>]
```



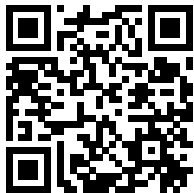
```
\documentclass{scrartcl}
\usepackage{scrlayer-scrpage}
\lohead*{Jane Smith}
\rohead*{Pagestyles with KOMA-Script}
\pagestyle{scrheadings}
```

# Font Selection

- Many fonts are available as packages and can be loaded with  
`\usepackage{package name}`

```
\usepackage{nimbusserif}
```

- Fonts available in T<sub>E</sub>Xlive can be found in the “L<sup>A</sup>T<sub>E</sub>X Font Catalogue”  
<http://www.tug.dk/FontCatalogue/>



# Font

- Package fontspec allows access to system fonts (OTF, AAT, TTF).
- Fonts are loaded via special commands

`\setmainfont[<Options>]{<Font Name>}`

```
\usepackage{fontspec}
\setromanfont{Linux Libertine O}
\setsansfont{Linux Biolinum O}
\setmonofont[Scale=.95]{DejaVu Sans Mono}
```

- Loading specific fonts or features in the document with

`\fontspec{<Font Name>}[<Features>]`



# Font Size

The size of the main font can be changed by class option:

```
\documentclass[12pt]{scrartcl}
```

Size of `\large`, `\small`, etc. adjusts automatically.

Standard classes support 10pt, 11pt, and 12pt.

For those who *know exactly* what they want:

```
\fontsize{⟨Size⟩}{⟨Baseline Skip⟩}\selectfont
```

```
\fontsize{10}{12}\selectfont
```

## Assignment

Adapt your document according to the specifications for bachelor theses!

Format one-sided DIN A4

Font Size 12 pt

Line Spread 1.5

Alignment justified

Left and right margin 3 cm

# Environments

- $\text{\LaTeX}$  documents are often structured by environments:

```
\begin{\langle Environment \rangle}[\langle Optional Arguments \rangle]{\langle Arguments \rangle}
```

...

```
\end{\langle Environment \rangle}
```

- Commands are executed at the beginning and end to achieve specific behavior within the environment.
- Each environment is a group (like `{}`)  
⇒ All settings within an environment are local.



# Environments

## Important Environments

Itemization	<code>itemize</code>
Enumeration	<code>enumerate</code>
Description list	<code>description</code>
Verbatim	<code>verbatim</code>
Two-column layout	<code>twocolumn</code>
Quotation	<code>quotation</code>
Short quote	<code>quote</code>
Centered	<code>center</code>
Table	<code>tabular</code> , <code>tabularx</code> , <code>tabulary</code> , <code>supertabular</code> etc.
Figure	<code>figure</code>
Floating table	<code>table</code>
Equation	<code>align (Math)</code>
Matrix	<code>matrix (Math)</code>

# Environments

## Simple Lists

```
\begin{itemize}
  \item First item
  \item Second item
  \item[3] Third item
\end{itemize}
```

- First item
- Second item
- 3 Third item

```
\begin{enumerate}
  \item First item
  \item Second item
  \item[3] Third item
\end{enumerate}
```

- ① First item
- ② Second item
- 3 Third item

Appearance of itemize and enumerate is determined by document class.

## Assignment

Add one or more quotes to your document. Observe the difference between quote and quotation.

Also, test the appearance of other environments like `itemize` and `description`.

# Microtypography

Microtypography refers to the design of fine details at the letter level:

**protrusion** Optical margin alignment

Lorem ipsum dolor sit amet consectetur, adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Microtypography

Microtypography refers to the design of fine details at the letter level:

**protrusion** Optical margin alignment

**expansion** Adjustment of glyph width ( $\leq 2\%$ )

Text

Text

# Microtypography

Microtypography refers to the design of fine details at the letter level:

**protrusion** Optical margin alignment

**expansion** Adjustment of glyph width ( $\leq 2\%$ )

**tracking** Adjustment of glyph spacing within words  
( $\leq 3\%$ )

VA Fo  
VA Fo

# Microtypography

Microtypography refers to the design of fine details at the letter level:

<b>protrusion</b>	Optical margin alignment	fi fi
<b>expansion</b>	Adjustment of glyph width ( $\leq 2\%$ )	fl fl
<b>tracking</b>	Adjustment of glyph spacing within words ( $\leq 3\%$ )	ff ff
		ffl ffl
<b>ligatures</b>	Connection of multiple letters to a single glyph	Qu Qu

# Microtypography

The package `microtype` takes care of these typographic subtleties. Usually, the default settings are sufficient:

```
\usepackage{microtype}
```

- Automatically activates protrusion (in pdf $\text{\TeX}$ , X $\text{\TeX}$  and Lua $\text{\TeX}$ ) and expansion (in pdf $\text{\TeX}$  and Lua $\text{\TeX}$ )
- For further options: Documentation

## Assignment

Activate optical margin alignment in your document.



# Whitespace and Dashes

Good typography distinguishes between various width spaces and horizontal dashes

- normal space
- thin space: `\,` e.g.  $a\,b$
- hair space: `\enskip` e.g.  $a\enskip b$
- em space (white square): `\quad` e.g.  $a\quad b$
- negative space: `\!` e.g.  $ab$
- explicit kerning: `a\kern-.1em b` e.g.  $ab$
- en dash: `-` e.g.  $a-b$
- em dash, German hyphen: `--` e.g.  $a--b$
- horizontal bar, English hyphen: `---` e.g.  $a---b$
- minus sign: `-$-$` e.g.  $a-b$   
e.g.  $a+b$

Part IV

# Documentation & Error Messages

- $\text{\LaTeX}$  is excellently documented.
- Each class and package usually comes with its own manual.
- Documentation can be accessed using the `texdoc` command.

On the command line:

- `$ texdoc` searches the  $\LaTeX$  folders for documentation.
- `$ texdoc amsmath` opens `amsmath.pdf`.
- `$ texdoc -l amsmath` lists all results.
- `$ texdoc -s amsmath` provides results from an extended search.
- `$ texdoc --help` displays help.

Graphical interface: `texdoctk` / `texdoc-gui`

Web service: <http://texdoc.org>

## Assignment

Open the English documentation of the KOMA-Script classes using the `texdoc` mechanism.

# Handling Errors

## What to do when $\text{\LaTeX}$ stops?

- Stay calm! (tex files cannot be damaged)
- Start troubleshooting with the latest changes.
- Correct any typos if necessary.
- Read the log file!
- Many editors assist in error detection by jumping to the line where the error occurred. (It may not be the faulty line.)

# Error Messages

Typical error message:

```
! Undefined control sequence.  
1.3 Ein \Latex-Dokument  
.  
?  
! Emergency stop.  
1.3 Ein \Latex-Dokument.  
.  
No pages of output.  
Transcript written on document.log.
```

⇒ Command misspelled in line 3

# Error Messages

Typical error message:

```
Runaway argument?  
{itemize \item Erstes Item  
! Paragraph ended before \begin was complete.  
<to be read again>  
                                \par  
1.60  
  
?
```

⇒ Forgot a } or an \end{} somewhere after itemize.

# Complete Minimal Working Example (MWE)

When seeking help in web forums/Usenet, a *complete Minimal Working Example* (MWE) is usually requested.

- ① Delete code from the document until the error occurs.
- ② Remove all unnecessary packages.
- ③ Use `minimal` if the document class does not matter.
- ④ If the error occurs only with a lot of text, use `blindtext`.

Often, you can find the error while creating the MWE alone.

## Assignment

Download the document `exercise_errors.tex` from the workshop homepage, create an MWE, and try to fix all errors if possible.





Part V

# Languages

The document must be localized based on the input language.

- Hyphenation rules
- Names of directories, chapters, etc.
- Typographic peculiarities

```
\usepackage{polyglossia}  
\setmainlanguage{german}  
\setotherlanguage{english}
```



# Loading Languages

```
\setmainlanguage[⟨Options⟩]{⟨Language⟩}  
\setotherlanguage[⟨Options⟩]{⟨Language⟩}  
\setotherlanguages{⟨Languages⟩}
```

## Available Languages:

---

albanian	danish	icelandic	nko	slovenian
amharic	divehi	interlingua	norsk	spanish
arabic	dutch	irish	nynorsk	swedish
armenian	english	italian	occitan	syriac
asturian	esperanto	kannada	piedmontese	tamil
bahasai	estonian	khmer	polish	telugu
bahasam	farsi	korean	portuges	thai
basque	finnish	lao	romanian	tibetan
bengali	french	latin	romansh	turkish
brazil[ian]	friulan	latvian	russian	turkmen
breton	galician	lithuanian	samin	ukrainian
bulgarian	german	lsorbian	sanskrit	urdu
catalan	greek	magyar	scottish	usorbian
coptic	hebrew	malayalam	serbian	vietnamese
croatian	hindi	marathi	slovak	welsh
czech				

---

# Switching Languages

Command `\text{⟨Language⟩{⟨Text⟩}}` for individual words

Environment `\begin{⟨Language⟩}` for longer passages

% in the preamble:

```
\setmainlanguage{english}  
\setotherlanguages{french, greek}
```

% in the document:

The document body is in English, but single words can be in `\textgreek{ελληνικά}` or `\textfrench{français}`.

```
\begin{french}  
  Il est également possible d'écrire des phrases entières en français.  
\end{french}
```

# Localized Objects

Labels of elements in the text adapt to the language:

```
heute ist der \today \\
\textenglish{today is \today}\\
\texttrussian{ сегодня, является \today }
```

```
heute ist der 2024-02-16
today is 2024-02-16
сегодня, является 16 февраля 2024 г.
```

## Assignment

Ensure correct hyphenation in at least two languages in your document.

Part VI

# Floating Objects

# What are Floating Objects?

- Objects that can freely "float" within the document
- Floating helps avoid large blank spaces
- $\text{\TeX}$  tries to achieve optimal positioning
- Considerations:
  - Objects should not appear before references
  - Objects should not swap order
  - Page breaks heavily depend on floating objects
  - *Optimal page breaks are not possible with  $\text{\TeX}$ !*

# Float Environments

A float environment consists of various parts:

- Content (image, table, text, ...)
- Automatic labeling: "Table 1:" (`\caption`)
- Caption: "Measurement results" (argument of `\caption{}`)
- Label for references: `\label{fig:comparison-data}`
- The label can be referenced in the text using `\ref{fig:comparison-data}`
- `\listoffigures` and `\listoftables` automatically create lists of figures and tables



# Float Environments

- $\text{\LaTeX}$  provides various float environments:
- `table` for tables
- `figure` for images
- `float` package allows defining custom environments
- For two-column layout: `table*`, `figure*` spanning both columns



## Positioning parameters for float environments:

```
\begin{table}[\langle parameters \rangle]
```

- ! Overrides internal parameters
- h Set object exactly at this point
- t Set object at the top of the page
- b Set object at the bottom of the page
- p Set object on a dedicated float page or column
- H "exactly here and nowhere else" – float package

# table

```
\begin{table}
  \centering
  \begin{tabular}{ccc}
    a & b & c
  \end{tabular}
  \caption{A meaningless table}
  \label{tab:meaningless}
\end{table}
```

In the text, you can refer to Table  
`\ref{tab:meaningless}`.

a	b	c
---	---	---

**Table 1:** A meaningless table

In the text, you can refer to Table 1.

# Non-floating Float Environments

Present non-floating environments as floating environments:

caption package

A small image in a text that is not really an image:

```
\begin{minipage}[b]{3cm}
  \fbox{ I am not an image }
  \captionof{figure}{test}
\end{minipage}
```

In the `\verb/minipage/`, any content can be placed `\dots`

A small image in a text that is not really an image:

I am not an image

Figure 1: test

In the minipage, any content can be placed ...

# Including External Graphics

```
\usepackage{graphicx}
```

- Basic command: `\includegraphics[<options>]{<file>}`
- key=value interface:  
[scale = 0.5, angle=50]
- File extension is not required
- Avoid using absolute path names (portability)

# Including Graphics

```
\includegraphics[width=2cm]{raptor.pdf}  
\includegraphics[width=.3\textwidth,angle=25]{raptor}
```



# Options for includegraphics

`\includegraphics` has many options, e.g.,

<code>scale</code>	<code>0.8</code>
<code>width</code>	<code>.2\textwidth, 15pt, ...</code>
<code>height</code>	<code>2em, 40mm, ...</code>
<code>keepaspectratio</code>	<code>true or false</code>
<code>angle</code>	<code>50</code>
<code>bb</code>	<code>0 0 10 20</code>
<code>clip</code>	<code>true or false</code>

⇒ see documentation for `graphicx`

# Multiple Images in One Figure

```
\usepackage{subcaption}

\begin{figure}
  \begin{subfigure}{.5\textwidth}
    \includegraphics{image1}
    \caption{First subfigure}
  \end{subfigure}
  \begin{subfigure}{.5\textwidth}
    \includegraphics{image2}
    \caption{Second subfigure}
  \end{subfigure}
  \caption{Caption for both images}
\end{figure}
```

The subcaption package provides the subfigure environment within the figure environment.





# Further Reading I



Frank Mittelbach and Ulrike Fischer.

“The  $\LaTeX$  companion.”

Addison-Wesley, 2023.



Tobias Oetiker et al.

“The (Not So) Short Introduction to  $\LaTeX$  2 <sub>$\epsilon$</sub> .”

`texdoc lshort`



Robert Bringhurst.

“The Elements of Typographic Style.”

Vancouver: Hartley & Marks, 1992.



Markus Kohm.

“KOMA-Script.”

`texdoc koma-script`

2023.

## Further Reading II



Arthur Reutenauer, François Charette, and Elie Roux.

“polyglossia: An Alternative to Babel for XeLaTeX and LuaLaTeX.”

[texdoc polyglossia](#)



Michael Goossens, Sebastian Rahtz, and Frank Mittelbach.

“The  $\text{\LaTeX}$  Graphics Companion.”

Addison-Wesley, 2008.

Happy T<sub>E</sub>Xing