

LaTeX Grundlagen

für Naturwissenschaftler und Informatiker

Kapitel 0: Was ist LaTeX und warum will ich es benutzen?

Was ist TeX?

- Computerprogramm zum Setzen von Dokumenten = Textsatzsystem
- Eingabedokument wird in reinem Text in einem Text-Editor verfasst
- Formatierungssprache: Formatierung wird mittels Befehlen markiert
- eingebaute Makrosprache; Definition eigener Makros
- Umwandlung der Textdateien in Binärdatei einer Seitenbeschreibungssprache (DVI, PDF, PS)
- kein Textverarbeitungsprogramm; kann nicht mit Textverarbeitungsprogramm verglichen werden, da sie unterschiedlichen Zwecken dienen
- Open Source und betriebssystemunabhängig
- kein WYSIWYG („What You See Is What You Get“)
- sehr gut dokumentiert

Was macht TeX so besonders?

- eine besondere Stärke ist der mathematische Formelsatz sowie das Schriftbild
- Layout gilt als sehr sauber, Formelsatz als sehr ausgereift
- Regelsatz für das Setzen von Abständen in mathematischen Formeln
- Textsatz erfolgt absatzweise (Total-Fit statt First-Fit):
 - Worttrennungsalgorithmus
 - Zeilenumbruchoptimierung: alle Zeilenumbrüche (inklusive Worttrennung) im Absatz gleichzeitig bestimmt
 - Seitenumbruchoptimierung

Was ist LaTeX?

- Softwarepaket, das die Benutzung des Textsatzsystems TeX mit Hilfe von Makros vereinfacht
- Sammlung von TeX-Makros, die die Benutzung für den durchschnittlichen Anwender gegenüber TeX vereinfachen und erweitern
- logisches Markup für die Gliederung
- am einfachsten zu erlernende und auch für unterschiedlichste Kombinationen von Anforderungen am besten gepflegte Zugang zu TeX
- ermöglicht einfache

- automatische Nummerierung von Kapiteln, Abschnitten, Sätzen, Gleichungen,...
- Querverweise

TeX Engines

Es gibt mehrere Engines, die TeX-Eingaben verarbeiten können, wobei nicht alle in gleichem Umfang genutzt werden:

TeX

- Ursprüngliche Engine von Knuth
- erzeugt DVI Dokument
- kann EPS und PS importieren

pdfTeX

- vollständig rückwärtskompatibel zu Knuths TeX und bleibt damit ein 8-Bit-System
- Erzeugt PDF Dokument

XeTeX

- Unicode-basiert (UTF-8), d.h. Unicode als Eingabe
- verwendet systemabhängige Bibliotheken (werden in das Binärprogramm eingebaut), um auf Systemschriften zuzugreifen
- erzeugt keine direkte PDF-Ausgabe, sondern arbeitet über XDV (eXtended DVI) als Zwischenformat ⇒ als Teil des xetex-Laufs mittels xdvpdfmx in PDF konvertiert

LuaTeX

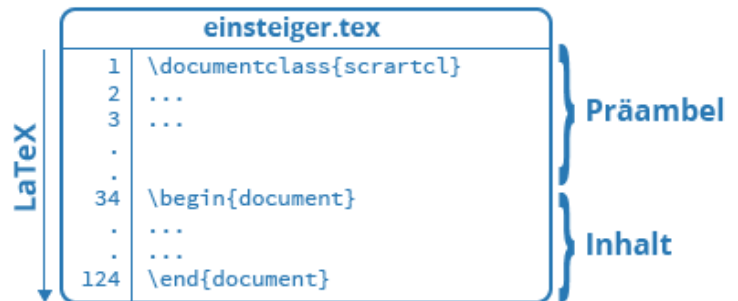
- ebenfalls Unicode-basiert (UTF-8), d.h. Unicode als Eingabe
- "eingebetteter" Lua-Interpreter hat Zugriff auf viele der Datenstrukturen, die für Textsatz verwendet werden
- Lua-Code kann in (La)TeX-Makros eingefügt werden
- Laden der Systemschriften über Lua-Code
- ConTeXt ist ein Format für TeX, wie LaTeX, aber neuer und viel größer und eine reine LuaTeX-Implementierung

Kapitel 1: Grundlagen

Syntax

Minimaldokument

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



Reservierte Zeichen & Leerzeichen

Folgende Zeichen sind Teil der LaTeX Syntax und werden nicht ins Dokument umgesetzt:

Sonderzeichen	Bedeutung	Beispiele	Druck
\	Befehl	\emph \rightarrow \alpha	\backslash, \textbackslash{}
{}	Gruppierung von Argumenten	\emph{kursiver Text} {hochgestellter Text}	\{ \}
#	Argumente in selbstdefinierten Befehlen	\newcommand{\mycom}[2] {what to do with my arguments #1 and #2}	\#
\$\$	mathematische Formeln	$2^{n \log n}$	\\$
^	Hochstellen (im Mathemodus)	$2^{n \log n}$	\textasciicircum \^{}
_	Tiefstellen (im Mathemodus)	x_i	_
&	Zellentrennung in Tabellen	Zelle 1 & Zelle 2 \\	\&
~	Leerzeichen ohne Zeilenumbruch	Dr.~Who?	\textasciitilde \~{}
%	Kommentare	% das ist ein Kommentar	\%

Typische Fehlermeldungen:

- ! Misplaced alignment tab character &. Ein & wurde unmaskiert außerhalb einer Tabellen oder Array Umgebung verwendet.

Leerzeichen

- Ein Leerzeichen ist das gleiche wie eine Million Leerzeichen.
- Tabs werden wie Leerzeichen behandelt.
- Leerzeichen am Ende einer Zeile werden ignoriert.
- Einzelner Zeilenumbruch wird wie Leerzeichen behandelt.
- Mehr als ein Zeilenumbruch markiert den Beginn eines neuen Absatzes.
- Leerzeichen nach Befehlen werden ignoriert ⇒ mit Backslash Leerzeichen erzwingen (z.B. `gro\ss\oder \LaTeX\`).

Befehle, Gruppierungen & Umgebungen

Befehle

`\commandname [<option1>,<option2>,...] {<argument1>} {<argument2>} ...`

eigene Befehle definieren:

`\newcommand{\mycommand} [<maxNumParam>] [<default>] {what to do with my arguments #1 and #2}`

`maxNumParam` Anzahl an Argumenten (max 9)

`default` default für den einen möglichen optionalen Parameter (immer #1)

Gruppierungen

lokale Bereiche innerhalb eines LaTeX-Dokuments, die innerhalb von geschweiften Klammern gesetzt sind

`{ <inhalt> }`

Befehle innerhalb einer Gruppierungen wirken sich nicht auf das gesamte Dokument aus

Gruppierungen können auch mittels `\begingroup <inhalt> \endgroup` erzeugt werden

Umgebungen

`\begin{<environment>} ... \end{<environment>}`

eigene Umgebungen definieren:

`\newenvironment{myenvironment} [<maxNumParam>] {<begin>} {<end>}`

Typische Fehlermeldungen:

- ! Undefined control sequence ...
→ Unbekannter Befehl verwendet: Tippfehler; benötigte Klasse bzw. Paket nicht eingebunden (selten wenn Klasse / Paket geupdated wurde und der Befehl nicht mehr existiert); falscher Compiler
- ! LaTeX Error: Environment ... undefined.
→ Unbekannte Umgebung
- ! Argument of \@fileswithoptions has an extra }.
Nicht zusammenpassende Klammern

Erzeugen des Dokuments

Je nach TeX-Engine (jeweils für LaTeX):

<code>latex <file></code>	tex Compiler mit LaTeX Initialisierungsdateien; erzeugt DVI Dokument kann EPS und PS importieren
<code>pdflatex <file></code>	pdfTeX Compiler mit LaTeX Initialisierungsdateien; erzeugt PDF kann JPG, PNG und PDF importieren
<code>xelatex <file></code>	XeTeX Compiler mit LaTeX Initialisierungsdateien; erzeugt PDF Unicode als Eingabe; Systemschriften
<code>lualatex <file></code>	LuaTeX Compiler mit LaTeX Initialisierungsdateien; erzeugt PDF Unicode als Eingabe; Systemschriften

⇒ zweimal compilieren nötig

Hilfsdateien

Werden von (La)TeX erzeugt und speichern Informationen

<code>.aux</code>	speichert Informationen von einem Kompilervorgang zum nächsten (z.B. Querverweise)
<code>.log</code>	Protokoll des letzten Kompilervorgangs
<code>.toc</code>	Inhaltsverzeichnis
<code>.lof</code>	Abbildungsverzeichnis
<code>.lot</code>	Tabellenverzeichnis
<code>.idx</code>	Index-Register

Dokumentaufbau und Präambel

```
\documentclass[<options>]{<class>}  
<Präambel>  
\begin{document}  
\end{document}
```

Dokumentklassen

Dokumentklasse	Verwendungszweck	Default Optionen
article	Arbeiten mit geringem Seitenumfang (wissenschaftliche Artikel,...) keine automatische Titelseite keine Kapitel (<code>\chapter</code>)	letterpaper, 10pt, oneside, onecolumn, final
report	kleinere Abschlussarbeiten (Projektarbeit, Bachelorarbeit,...) Titelseite	letterpaper, 10pt, oneside, onecolumn, final, openany
book	größere Dokumente (Dissertationen,...)	letterpaper, 10pt, twoside, onecolumn, final, openright
letter	amerikanisches Briefformat	letterpaper, 10pt, oneside, onecolumn, final

Die Klasse `report` eignet sich um mit `book` erstellte Dokumente schnell in eine bildschirmlesbare Variante zu überführen (z.B. keine leeren Seiten)

LaTeX-Standardklassen richten sich nach US-amerikanischen typografischen Konventionen und Papierformaten.

KOMA-Script

- beruht auf europäischen typografischen Konventionen und DIN-Papierformaten
- komplette Benutzerschnittstelle der Standardklassen erweitert um zusätzliche Funktionen

`scrartcl`, `scrbook`, `scrlettr2`

Optionen:

Schriftgröße: 10pt, 11pt, 12pt
 Seitenformat: onecolumn, twocolumn, oneside, twoside
 Papierformat: a4paper, letterpaper, a5paper, b5paper, executivepaper, legalpaper
 Titelseite: titlepage, notitlepage
 Kapitel: openany, openright
 Grafiken: final, draft
 Seitenausrichtung: portrait, landscape
 Abstand zwischen Absätzen: noparskip, halfparskip
 Formeln: fleqn, leqno

Typische Fehlermeldungen:

- ! LaTeX Error: File `articles.cls' not found.
→ Der Klassenname ist falsch oder die Klasse ist nicht installiert.
- ! LaTeX Error: Missing \begin{document}.
→ Vor dem Befehl \begin{document} steht normaler Text. Oder der Befehl \begin{document} fehlt.
- *
→ \end{document} fehlt.

Zusatzpakete

```
\usepackage[<options>]{<package>}  
\usepackage{<package1>,<package2>,<package3>}
```

Typische Fehlermeldungen:

- ! LaTeX Error: Can be used only in preamble.
→ \usepackage Befehl außerhalb der Präambel
- ! LaTeX Error: File `lmonder.sty' not found.
→ Das Paket ist nicht vorhanden bzw. der Name ist falsch geschrieben.
- ! LaTeX Error: Unknown option `...' for package `...'.
→ Unbekannte Option für ein Paket.
- ! LaTeX Error: Option clash for package `...'.
→ Verschiedene (widersprüchliche) Optionen werden für ein und dasselbe Paket gesetzt. Das Paket wurde zweimal geladen, möglicherweise durch ein weiteres Paket.

Ein erstes Paket: blindtext

```
\blindtext[<x>]          erstellt einen Absatz Mustertext mit <x> Wiederholungen  
\Blindtext[<y>][<x>]    erstellt <y> Absätze an Mustertext mit <x> Wiederholungen
```

Seitenlayout

Seitenlayout anzeigen lassen

Pakete zur Visualisierung des Seitenlayouts:

```
\usepackage{showframe}    blendet Seitenränder ein  
  
\usepackage{layout}      Ausgabe des aktuellen Seitenlayouts, inklusive der  
\layout                  Parameter und Werte  
  
\usepackage{layouts}     weitere mögliche Layout Ausgaben
```

Seitenlayout anpassen

Ändern der Layout-Variablen:

```
\setlength{\<längenbefehl>}{\<Abstand>}
```

Längeneinheiten:

```
pt    1/72 inch  
mm  
cm  
in    inch  
ex    Schriftart abhängig; etwa die Höhe eines "x"  
em    Schriftart abhängig; etwa Breite eines "M"s
```

Auch Längenbefehle können als Längeneinheiten verwendet werden:

```
\parindent    Länge des Einzugs von Absätzen  
\baselineskip vertikaler Abstand zwischen Zeilen  
\parskip      zusätzlicher Abstand zwischen Absätzen  
\textwidth    Zeilenbreite in der derzeitigen Umgebung  
\textheight   Höhe des Textes auf der derzeitigen Seite
```

Das geometry Paket

- ändert relativ viele Parameter innerhalb des Dokumentes
- sollte möglichst als letztes Paket eingebunden werden

```
\usepackage[<options>]{geometry}
```

oder

```
\usepackage{geometry}
```

```
\geometry{<options>}
```

Optionen:

- Optionen der Dokumentklassen werden automatisch übernommen oder können überschrieben werden
- ergänzt nicht gesetzte Werte entsprechend
- Die Werte der Längen-Variablen `\topmargin`, `\oddsidemargin` und `\evensidemargin` werden den mittels des `geometry`-Pakets eingestellten Randwerten hinzuaddiert

Seitenränder: `top=xx`, `bottom=xx`, `left=xx`, `right=xx`

Papierformat: `width=xx`, `height=xx`

Textblock: `textwidth=xx`, `textheight=xx`

Fußnoten: `footsep=xx`, `footskip=xx`

Kopf- und Fußzeilen

`\pagestyle{<option>}` gesamtes Dokument
`\thispagestyle{<option>}` einzelne Seite

Optionen:

`plain` Seitenzahl Fußzeile (default für `article`)
`empty` keine Seitenzahlen
`headings` laufende Kopfzeilen auf jeder Seite; Dokumentklassen-abhängig was in den Kopfzeilen steht; default für `book`
`book, report:`
 `einseitig:` — Kapitel
 `beidseitig:` Kapitel Section
`article`
 `einseitig:` — Section
 `veidseitig:` Section Subsection
`myheadings` eigene Kopfzeilen Gestaltung mittels
 `\markboth{<linke Kopfzeile>}{<rechte Kopfzeile>}` oder
 `\markright{<rechte Kopfzeile>}`

Kolumnentitel

Chapter, Section oder Subsection automatisch in Kopf- oder Fußzeile anzeigen

`\automark[<rechte Seite>]{<linke Seite>}`

⇒ Festlegen, was als Kolumnentitel ausgegeben werden soll

Optionales Argument nur für doppelseitige Dokumente nötig

Das `fancyhdr` Paket:

Anpassen von Kopf-/Fußzeilen

`\usepackage{fancyhdr}`

`\pagestyle{fancy}`

Löschen der defaults bzw. vorheriger Einstellungen:

`\fancyhead{}`

`\fancyfoot{}`

`\lhead[<lh-even>]{<lh-odd>}` `\lfoot[<lf-even>]{<lf-odd>}`

`\chead[<ch-even>]{<ch-odd>}` `\cfoot[<cf-even>]{<cf-odd>}`

`\rhead[<rh-even>]{<rh-odd>}` `\rfoot[<rf-even>]{<rf-odd>}`

Zugriff auf Kolumnentitel via `\leftmark` und `\rightmark`

Das `scrlayer-scrpage` Paket

KOMA-Erweiterung ⇒ nur für KOMA-Script Klassen

```
\usepackage{scrlayer-scrpage}
```

Zwei neue Seitenstile:

<code>scrplain</code>	Konfiguration für Seiten die z.B. Teil von Inhalts- und Tabellenverzeichnis sind
	Einseitige Dokumente: Seitennummer unten (zentriert)
	Zweiseitige Dokumente: Seitennummer unten (außen)
<code>scrheadings</code>	reguläre Konfiguration für den Hauptteil des Dokuments
	Einseitige Dokumente: Kolummentitel oben (zentriert), Seitennummer unten (zentriert)
	Zweiseitige Dokumente: Kolummentitel oben (außen, links Hierarchie 1, rechts Hierarchie 2) Seitennummer unten (außen)

- Stile werden automatisch auf den entsprechenden Seiten aktiviert
- Löschen der defaults bzw. vorheriger Einstellungen: `\clearpairofpagestyles`

```
\ihead[<definescrplain>]{<definescrheadings>}    innen  
\chead[<definescrplain>]{<definescrheadings>}    mitte  
\ohead[<definescrplain>]{<definescrheadings>}    außen
```

Gleiches für `\ifoot`, `\cfoot` `\ofoot`

`\headmark` Ausgabe des aktuellen Kolummentitels entsprechend der jeweiligen Seite

Anpassung des Kolummentitels:

```
\automark[chapter]{section}                    Voreinstellung  
\automark{<linke Seite>}{<rechte Seite>}      Anpassung
```

`automark` kann auch als Paketoption übergeben werden

Trennlinien zwischen Kopfzeile und Text erstellen

Paketoptionen:

<code>headtopline</code>	aktiviert die Linie über der Kopfzeile
<code>headsepline</code>	aktiviert die Linie unter der Kopfzeile
<code>footsepline</code>	aktiviert die Linie über der Fußzeile
<code>footbotline</code>	aktiviert die Linie unter der Fußzeile

Änderungen am Schriftsatz

```
\setkomafont{pageheadfoot}{<define font>}  
\setkomafont{pagenumber}{<define font>}
```

Seitenzahlen

```
\pagenumbering{gobble}
```

gobble	keine Seitenzahlen
arabic	arabische Zahlen
roman	kleine römische Zahlen
Roman	große römische Zahlen
alph	Kleinbuchstaben
Alph	Großbuchstaben

- Aktuelle Seitenzahl ausgeben: `\thepage` oder `\pagemark`
- Startseitenzahl festlegen: `\setcounter{page}{<number>}`

Mehrspaltige Dokumente

```
\documentclass[twocolumn]{article}
```

Das `multicol` Paket

```
\usepackage{multicol}  
\begin{multicols}{2} ... \end{multicols}
```

- bis zu zehn Spalten
- ermöglicht die Anzahl der Spalten in einem Dokument zu wechseln
- kann innerhalb anderer Umgebungen (z.B. `figure`) genutzt werden
- ausgeglichene Spaltenlänge auf der letzten Seite
- vertikale Linien zwischen den Spalten können angepasst werden
- Schwierigkeiten mit Floats (lieber * Umgebungen verwenden, z.B. `\begin{figure*}`), überspannt die ganze Seitenbreite)

```
\setlength{\columnseprule}{<Breite>}   vertikale Linie zwischen den Spalten  
\setlength{\columnsep}{<Breite>}       horizontaler Abstand zwischen den Spalten
```

Dokumentstruktur

Titel(seite)

```
\title{ }, \author{ }, \date{ }  
heutiges Datum: \date{\today}
```

```
\begin{titlepage} ... \end{titlepage}   generiert Titelseite  
\maketitle                               generiert Titel
```

Abstract

```
\begin{abstract} ... \end{abstract}  
\renewcommand{\abstractname}{<new name>}
```

Kapitel/Sections

```
-1 \part{<text>}  
0 \chapter{<text>}  
1 \section{<text>}  
2 \subsection{<text>}  
3 \subsubsection{<text>}  
4 \paragraph{<text>}  
5 \subparagraph{<text>}
```

```
\minisec{<text>}      nur in den KOMA-Script Klassen; keine Nummerierung
```

```
\section[<kurzer Titel>]{<langer Titel>}  
kurzer Titel erscheint im Inhaltsverzeichnis, langer Titel im Dokument
```

```
\section*{<Section ohne Nummerierung>}
```

* unterbindet die Nummerierung

```
\setcounter{secnumdepth}{1} Nummerierungstiefe  
\setcounter{tocdepth}{3}    Tiefe des Inhaltsverzeichnis  
\setcounter{section}{4}    Startwert für die Nummerierung der nachfolgenden  
                            Sections
```

Inhaltsverzeichnis

```
\tableofcontents  
\setcounter{tocdepth}{<depth>}    Tiefe des Inhaltsverzeichnis (3 ist default)  
\listoffigures    Abbildungsverzeichnis  
\listoftables    Tabellenverzeichnis
```

* Kommandos (Sections, Figures, Tables,...) werden nicht im Inhaltsverzeichnis aufgeführt

Einfügen einer unnummerierten Section ins Inhaltsverzeichnis:

```
\section*{Introduction}  
\addcontentsline{toc}{section}{Introduction}
```

bzw generell:

```
\addcontentsline{<type>}{<level>}{<entry>}
```

type	toc, lof, lot
level	Art des Eintrags chapter, section, subsection, ... für Inhaltsverzeichnis (toc) figure, table für Abbildungs (lof)- oder Tabellenverzeichnis (lot)
entry	Text; Befehle müssen geschützt werden mit <code>\protect</code>

Einrücken des Eintrags im Verzeichnis:

```
\addcontentsline{lof}{figure}{\protect\numberline{\thefigure}<Bildunterschrift>}
```

Eintragen von Zusatzinformationen oder z.B. extra Abstand:

```
\addtocontents{<type>}{<entry>}
```

Anpassen von Überschriften

Das `titlesec` Paket

```
\titleformat{\<level>}[<shape>]{<format>}{<label>}{<sep>}{<beforecode>}[<aftercode>]
```

level	chapter, section, ...
shape	Form der Überschrift: hang, block, display, runin, leftmargin, rightmargin, drop, wrap, frame
format	Formatierung der Überschrift (Text)
label	Beschriftung/Nummerierung
sep	horizontaler Abstand zwischen Nummerierung und Titel
beforecode	Code vor dem Titel (darf auch leer sein)
aftercode	Code nach dem Titel (optional)

```
\titlespacing{\<level>}{<left>}{<before-sep>}{<after-sep>}
```

left	linker Rand
before-sep	vertikaler Abstand vor dem Titel
after-sep	Abstand zwischen Titel und Text

Für KOMA-Skript Klassen:

Keine Verwendung des `titlesec` Pakets

```
\addtokomafont{<level>}{<fontformat>}  
\renewcommand*{\<level>format}{<labelformat>} benutzt Zähler \the<level>  
\RedeclareSectionCommand[<options>]{<level>}
```

Optionen: `before-skip=xx`, `after-skip=xx`

Querverweise

```
\label{<key>}  
\ref{<key>}
```

- `key` kann Groß- und Kleinbuchstaben, Zahlen und Satzzeichen (aber keine Leerzeichen) enthalten
- `\ref{key}` kann bereits vor `\label{key}` verwendet werden

Verwendung:

- Sections, Kapitel, etc.
- Einträge in nummerierten Listen
- Floats (Abbildungen, Tabellen)
- Theoreme etc
- Matheumgebungen
- einzelne Zeilen in bestimmten Umgebungen (`align`, ...)

Seiten referenzieren

```
\pageref{<key>}
```

Das `varioref` Paket

```
\usepackage{varioref}  
\vref{<key>}
```

je nach Seite des gelabelten Objekts und des Verweises:

- Objekt und Verweis auf der selben Seite: `\vref` erzeugt Standardbefehl `\ref` ohne `\pageref`
- Objekt und Verweis auf verschiedenen Seiten:
 - Differenz > 1: `\vref` erzeugt `\ref` mit `\pageref`
 - Differenz = 1: `\vref` erzeugt `\ref` und “on the preceding page” oder “on the following page” oder ähnliche Phrasen

Kapitel 2: Text Formatierung

Zeichenkodierung

pdfTex-Engine

Eingabekodierung

```
\usepackage[<option>]{inputenc}
```

- Unterstützung erweiterter Eingabe-Zeichensätze mit ihren unterschiedlichen Kodierungen
- Umwandlung der Eingabe-Zeichenkodierung in die interne LaTeX-Standardsprache
- Zeichenkodierung des Editors verwenden
- bei wechselseitiger Bearbeitung, kann es bei unterschiedlicher Eingabe-Zeichenkodierung zu Problemen kommen
- die empfohlene Eingabekodierung ist `utf8`

Ausgabekodierung

```
\usepackage[<option>]{fontenc}
```

- Standardschriftart von LaTeX ist die Computer-Modern-Schriftfamilie (CM)
- bietet standardmäßig nicht alle 256 Zeichen des europäischen Zeichenvorrates in T1-Kodierung
- Problem: spezielle Zeichen (z.B. Umlaute) werden aus den begrenzt vorhandenen Zeichen neu zusammengesetzt
- `fontenc`-Paket erzwingt Ausgabefonts in entsprechender Kodierung (Option `T1` für Westeuropa)
- Text-Zeichen für die entsprechenden Sprachen als einzelne Zeichen
- für LaTeX stehen mehr als 100 freie Schriftarten zur Verfügung (LaTeX Font Catalogue)

Sprache

```
\usepackage[<language1>,<language2>,...]{babel}
```

- Anpassung für LaTeX an viele Sprachen
- Enthält Silbentrennung, Sonderzeichen, sprachspezifische typografische Regeln
- Übersetzung der sprachspezifischen Befehle im Dokument
- Mehrere Sprachen können geladen werden; letzte Sprache aktiv

```
\selectlanguage{<languageA>}
```

```
\foreignlanguage{<languageB>}{<Text in another language>}
```

```
\begin{otherlanguage}{<languageB>} <Text in another language>
```

```
\end{otherlanguage}
```

`deutsch: ngerman`

Umlaute / Akzente

- Ist die richtige Eingabe- und Ausgabekodierung eingestellt, können Umlaute und Akzente ohne Komplikationen verwendet werden.
- Ohne die Verwendung von Zusatzpakete müssen Umlaute, Akzente und Sonderzeichen mit den entsprechenden LaTeX Befehlen generiert werden

XeTeX- oder LuaTeX-Engine

Eingabekodierung über die Engine (Unicode)

⇒ Editor sollte auf entsprechende Zeichenkodierung eingestellt sein

Ausgabekodierung

```
\usepackage{fontspec}
```

- Systemschriften können geladen werden

```
\setmainfont{<systemfont>}
```

```
\setsansfont{<systemfont>}
```

```
\setmonofont{<systemfont>}
```

Sprache

```
\usepackage{polyglossia}
```

```
\setdefaultlanguage[<optionen>]{<language1>}
```

```
\setotherlanguage[<optionen>]{<language2>}
```

Optionen:

```
variant
```

```
spelling
```

```
script
```

```
...
```

Befehle:

```
\selectlanguage[<options>]{<language>}
```

```
\textlang[<options>]{<language>}{<text>}
```

```
\begin{lang}[<options>]{<language>} ... \end{lang}
```


Silbentrennung

eigene Regeln zur Silbentrennung definieren:

mögliche Trennstellen für das gesamte Dokument:

```
\hyphenation{zwei-fels-oh-ne son-nen-ba-den}
```

mögliche Trennstellen für ein einzelnes Vorkommen eines Wortes:

```
zwei\ -fels\ -oh\ -ne
```

Silbentrennung verhindern:

```
\mbox{<text>}
```

Befehl	Bedeutung/Funktion	Beispiel	Mögliche Trennstellen
-	Bindestrich, der andere Trennungen unterdrückt	Mess-Ergebnis	Mess-Ergebnis
"=	Bindestrich, der andere Trennungen erlaubt	Mess"=Ergebnis	Mess-Er-geb-nis
"~	Bindestrich, an dem nicht getrennt werden darf	Mess"~, Schätz"~ und andere Ergebnisse	Mess-, Schätz-, ...*
\-	Trennmöglichkeit, die andere Trennungen ausschließt	Ur\ -instinkt	Ur- instinkt
"-	Trennmöglichkeit, die andere Trennungen nicht ausschließt, wenn Tex mal was nicht findet	Am"-nestie	Am-nes-tie
""	Trennmöglichkeit, bei der kein Trennstrich benötigt wird	und/" "oder	und/-oder

* nur Bindestrich, keine Trennstelle

Um diese Befehle auch mit dem `polyglossia` Paket zu verwenden:

```
\usepackage[babelshorthands=true]{polyglossia}
```

Globale Einstellungen

```
\hyphenpenalty=<xx>
```

Strafe für Silbentrennung

```
\tolerance=<xx>
```

Toleranz für ungleichmäßigen Abstand zwischen Wörtern

Textsatz

Textsatz optimieren

<code>\widowpenalty=<xx></code>	Strafe Hurenkinder
<code>\clubpenalty=<xx></code>	Strafe Schusterjungen

Underfull-/Overfull Warnungen

Underfull Box: zu viel Leerraum in einer Zeile → große Abstände zwischen den Wörtern

Overfull Box: nicht genug Platz vorhanden ist, um den Text innerhalb der Seitenränder zu platzieren → Wörter überschreiten den rechten Rand (Verwendung der Option `[draft]` des Befehls `\documentclass`, um überfüllte Zeilen im Dokument hervorzuheben)

Das `microtype` Paket

- verbessert das allgemeine Erscheinungsbild des Textes erheblich, indem es verschiedene Techniken einsetzt
- ermöglicht es, mehr Wörter in pro Textzeile zu platzieren
- gesamter Wortabstand wird verringert und gleichzeitig gleichmäßiger verteilt
- Reduziert die Anzahl der "under-" und "overfull box"-Warnungen

Horizontaler vs vertikaler vs mathematischer Modus

Horizontaler Modus (Absatzmodus)

- Verarbeitung von gewöhnlichem Text
- Text in Zeilen und die Zeilen in Seiten umgebrochen
- **LR-Modus** (eingeschränkter horizontaler Modus): `\mbox`; keine automatischen Zeilenumbrüche
- `\parbox` und `minipage`: Absatzmodus ohne Seitenumbrüche

Mathe-Modus

- Inline Mathe-Modus
- Display Mathe-Modus

Vertikaler Modus

- Erstellen der Ausgangsseite (Liste der Zeilen, Gleitobjekte,...)
- Modus, in dem sich LaTeX befindet, wenn es ein Dokument startet

Befehle für manuellen Abstand

Befehle für vertikalen Abstand	
<code>\vspace{<skip>}</code>	vertikaler Abstand der Größe <code><skip></code>
<code>\smallskip, \medskip, \bigskip</code>	Abstand der von der jeweiligen Klasse vorgegeben wird; Spezialfälle von <code>\vspace{}</code>
<code>\smallbreak, \medbreak, \bigbreak</code>	ähnlich, entfernt aber vorangehenden Abstand markiert guten Punkt für Seitenumbruch
<code>\addvspace{<skip>}</code>	wird mit anderen Abständen zu Maximum "verschmolzen"
<code>\vfill, \vspace{\fill}</code>	vertikales Füllen mit Whitespace
<code>\vspace*{<skip>}</code>	wird auch nach einem Seitenumbruch eingefügt
<code>\vglue{<skip>}</code>	Abstand am Anfang einer Seite
Befehle für horizontalen Abstand:	
<code>\enskip, \quad, \qqquad</code>	horizontaler Abstand der Länge 0.5em, em, 2em
<code>\hspace{<skip>}</code>	horizontaler Abstand der Länge <code><skip></code>
<code>\hspace*{<skip>}</code>	Analog; wird auch nach Zeilenumbruch (<code>\newline</code>) eingefügt
<code>\hfill, \hspace{\fill}</code>	horizontales Füllen mit Whitespace

Auch negative Abstände sind möglich

Leerzeichen

- `\,` kleiner Abstand (z.B. zwischen Zahl und Einheit)
- `~` normales Leerzeichen ohne Zeilenumbruch: W.~R.~Hamilton

Leerzeichen nach Satzende (Punkt nach Kleinbuchstabe) größer als zwischen Wörtern

⇒ Unterdrücken mit Backslash: Mr.\ Smith, etc.\

⇒ erzwingen mit `\@`: Vitamin A\@.

Manuelle Umbrüche

Zeilenumbrüche	
<code>\newline</code>	Zeilenumbruch
<code>\\[<skip>]</code>	kurze Version für Zeilenumbruch in manchen Umgebungen anders definiert optionaler vertikaler Abstand <code><skip></code>
<code>*[<skip>]</code>	verhindert Seitenumbruch nach dem erzwungenen Zeilenumbruch optionaler vertikaler Abstand <code><skip></code>
<code>\linebreak[<number>]</code>	Zeilenumbruch der Priorität <code><number></code> von 0 (kann ignoriert werden) bis 4 (Umbruch erzwingen) Zeile wird bis an den Rand gezogen
<code>\break</code>	Zeilenumbruch ohne Zeile zu füllen
<code>\par</code>	neuer Paragraph
Seitenumbrüche	
<code>\newpage</code>	Seitenumbruch
<code>\pagebreak[<number>]</code>	Seitenumbruch der Priorität <code><number></code>
<code>\nopagebreak[<number>]</code>	Seitenumbruch verbieten mit Priorität <code><number></code>
<code>\clearpage</code>	Seitenumbruch; Floats werden alle hier platziert
<code>\cleardoublepage</code>	Seitenumbruch; Floats werden alle hier platziert nächste Seite beginnt rechts (fügt ggf Leerseite ein)

Typische Fehlermeldungen:

- ! LaTeX Error: There's no line here to end.
Ein `\\` oder `\newline` wurde fehlerhaft verwendet (um Leerzeile oder vertikalen Abstand zu erzeugen). Stattdessen: `\vspace`, `\vspace*` oder `\bigskip`

Einzüge

Einrücken verhindern: `\noindent`
 Einrücken erzwingen: `\indent`
 Länge des Einzugs: `\setlength{\parindent}{<length>}`

Zeilenabstand

`\renewcommand{\baselinestretch}{2}` doppelter Zeilenabstand

The `setspace` Paket

```
\usepackage{setspace}
\usepackage[<spacingoption>]{setspace}                      global
\begin{<spacingoption>}... \end{<spacingoption>}          Umgebungen
```

Optionen:

Global	Umgebung	
<code>singlespacing</code>	<code>singlespace</code>	normaler (einfacher) Zeilenabstand
<code>onehalfspacing</code>	<code>onehalfspace</code>	anderthalbfacher Zeilenabstand
<code>doublespacing</code>	<code>doublespace</code>	doppelter Zeilenabstand

```
\begin{spacing}{<Zeilenabstand als Zahl>}... \end{spacing}
```

Text Alignment

Alignment	Umgebung	Befehl
linksbündig	<code>flushleft</code>	<code>\raggedleft{<text>}</code>
rechtsbündig	<code>flushright</code>	<code>\raggedright{<text>}</code>
zentriert	<code>center</code>	<code>\centering{<text>}</code>

Schriften

Schriftarten und -größen

Schriftarten	Lokale Form	Globale Form
roman serif	<code>\textrm{<text>}</code>	<code>{\rmfamily <text>}</code>
sans serif	<code>\textsf{<text>}</code>	<code>{\sffamily <text>}</code>
typewriter	<code>\texttt{<text>}</code>	<code>{\ttfamily <text>}</code>
Schriftbreite		
fett	<code>\textbf{<text>}</code>	<code>{\bfseries <text>}</code>
medium	<code>\textmd{<text>}</code>	<code>{\mdseries <text>}</code>
Schriftformen		
gerade	<code>\textup{<text>}</code>	<code>{\upshape <text>}</code>
kursiv	<code>\textit{<text>}</code>	<code>{\itshape <text>}</code>

schräg	<code>\textsl{<text>}</code>	<code>{\slshape <text>}</code>
Kapitälchen	<code>\textsc{<text>}</code>	<code>{\scshape <text>}</code>
Default Schrift	<code>\textnormal{<text>}</code>	<code>{\normalfont <text>}</code>

`\emph{<text>}` hervorheben (in der Regel kursiv)

Schriftgrößen

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

Unterstreichen:

`\usepackage{ulem}` **Achtung:** überschreibt `\emph{}`
`\normalem` zurücksetzen auf von `\emph{}` auf kursiv

`\uline{<text>}` unterstreichen
`\uwave{<text>}` Wellenlinie
`\sout{<text>}` durchstreichen

Systemschriften verwenden

fontspec Paket

`\setmainfont{}[]`
`\setsansfont{}[]`
`\setmonofont{}[]`

Symbole und Icons

fontawesome5 Paket

Vollständige Symbolliste in der Paketdokumentation

Listen

Tabulator Umgebung

```
\begin{tabbing}
text \= text \>[5pt]
text \> text \>[5pt]
\end{tabbing}
```

`\kill` beendet *dummy* Zeile, die nicht gedruckt wird (Festlegen der Tabpositionen)
`\=` Tabpositionen können erneut geändert werden
`\+` am Ende einer Zeile; nachfolgende Zeilen beginnen mit dem ersten Tab
`\-` `\+` zurücksetzen

Aufzählungslisten

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

Individualisieren:

```
\renewcommand{\label<itemi>}{<character>}
Level:          itemi, itemii, itemiii, itemiv
Auzählungszeichen: \label<level>
```

Nummerierte Listen

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

Individualisieren:

```
\renewcommand{\the<level>}{\<zaehler>{<level>}}
\renewcommand{\label<level>}{\the<level>}
Level:          enumi, enumii, enumiii, enumiv
Zähler:        \arabic, \alph, \Alph, \roman, \Roman, \fnsymbol
```

Das enumerate Paket:

für mehr Individualisierung der Nummerierung

A, a, 1, I und i sind Platzhalter für die jeweiligen Zähler (sonst Klammern verwenden)

```
\begin{enumerate}[i.]
\item ...
\end{enumerate}
```

Das enumitem Paket

Verschiedene Optionen, die einer Liste übergeben werden können:

`itemsep`, `leftmargin`, `labelsep`, `itemindent`, `nosep`

Hier können sowohl Aufzählung als auch Verweise festgelegt werden:

```
label=<aufzaehlungsformat>
ref=<verweisformat>
```

Typische Fehlermeldungen:

- ! LaTeX Error: Too deeply nested.
→ **Schachtelung zu tief (>4 Ebenen)**
- ! LaTeX Error: Something's wrong--perhaps a missing \item.
→ **Umgebung ohne den Befehl \item verwendet.**
- ! LaTeX Error: \begin{document} ended by \end{itemize}
→ **ein \end{itemize} zu viel**
- ! LaTeX Error: \begin{itemize} on input line XX ended by \end{document}.
→ **\end{itemize} fehlt.**
- ! LaTeX Error: Lonely \item--perhaps a missing list environment.
→ **\item Befehl außerhalb der itemize Umgebung**
- ! Missing \endcsname inserted.
→ **\begin{\itemize}**
- ! LaTeX Error: Command \end{itemize} invalid in math mode.
→ **Innerhalb der itemize Umgebung wurde die Mathematikumgebung begonnen, aber nicht beendet.**

Beschreibungslisten

```
\begin{description}
\item[label] ...
\end{description}
```

Zitate

Umgebung	Verwendung
<code>\begin{quote} ...</code> <code>\end{quote}</code>	<ul style="list-style-type: none"> • für kurze Zitate, die aus einem einzelnen Absatz bestehen • von links und rechts etwas eingerückt, d.h. Zeilenlänge gegenüber dem restlichen Text etwas verringert
<code>\begin{quotation}</code> ... <code>\end{quotation}</code>	<ul style="list-style-type: none"> • für längere Zitate, die aus mehr als einem Absatz bestehen. • ebenfalls von links und rechts etwas eingerückt • erste Zeile eines neuen Absatzes leicht eingerückt
<code>\begin{verse} ...</code> <code>\end{verse}</code>	<ul style="list-style-type: none"> • für Gedichte • einzelne Zeilen durch <code>\\</code> explizit beendet • sehr lange Zeilen werden in den nachfolgenden Zeilen zusätzlich eingerückt und so als zusammengehörig gekennzeichnet

Fußnoten, Randnotizen etc.

Fußnoten

`\footnote{<text>}`

- Fußnoten in `minipage`-Umgebung haben eigenen Zähler (`mpfootnote`) und werden am unteren Rand der Minipage gesetzt
- Fußnoten in Tabellen werden nur in `tabularx`- und `lontable`-Umgebung korrekt gesetzt

Individualisieren:

Zähler	
<code>\renewcommand{\thefootnote}{\<zaehler>{footnote}}</code>	Zähler ändern. Mögliche Zähler, siehe nummerierte Listen
<code>\setcounter{footnote}{<zahl>}</code>	Zähler auf bestimmten Wert setzen
<code>\@addtoreset{footnote}{section}</code>	Zähler mit jeder neuen <code>section</code> wieder auf 0 setzen
<code>\footnote[<num>]{<text>}</code>	<code>num</code> ersetzt eigentlichen Zähler der Fußnote; Zähler wird nicht erhöht
Formatierung	
<code>\footnotesize</code>	Schriftgröße
<code>\footnotesep</code>	Zusätzlicher Abstand über Fußnote
<code>\setlength{\skip\footins}{<height>}</code>	Abstand zwischen Haupttext und Fußnoten
<code>\footnoterule</code>	Linie, die die Fußnoten vom Haupttext trennt

Auf vorherige Fußnote referenzieren

Funktioniert nur in den KOMA-Script Klassen

`\footref{<key>}`

Das `endnotes` Paket:

`\endnote{text}` statt `\footnote{text}`

oder:

`\renewcommand{\footnote}{\endnote}`

Notizen am Ende eines Kapitels etc gruppieren via `\theendnotes\bigskip`

Randnotizen

`\marginpar{text}`

Individualisieren

`\reversemarginpar` Randnotizen auf den anderen Seitenrand (innen)

`\normalmarginpar` Randnotizen auf den default Seitenrand (außen)

`\marginparwidth` Breite der Randnotizen

`\marginparsep` Abstand zwischen Randnotiz und Haupttext

`\marginparpush` minimaler Abstand zwischen zwei Randnotizen

Farben

`\usepackage{color}`

`\usepackage[usenames,dvipsnames,svgnames]{xcolor}`

`usenames` Namen der 16 Grundfarben (HTML)

`dvipsnames` weitere 64 Farben

`svgnames` weitere 150 Farben

`x11names` weitere 300 Farben

eigene Farben definieren:

`\definecolor{<mycolor>}{<model>}{<color-spec>}`

Farbmodels: `gray, rgb, RGB, HTML, cmyk`

`color-spec` je nach Farbmodel unterschiedlich

`\colorlet{<mycolor>}{<color1>!<percentage>!<color2>}`

Mixt `percentage%` der `color1` mit `(100-percentage)%` der `color2`

Farben verwenden:

`\textcolor{<color>}{<text>}`

`\pagecolor{<color>}`

Boxen

Boxen werden von TeX wie Zeichen behandelt

Positionsoptionen

`t` top

`b` bottom

`l` left

r right
c center
s stretch

LR (left-right) Boxen

`\makebox[<width>][<pos>]{<text>}` <width> als Längenmaß
`\makebox(<width>,<height>)[<pos>]{Text}` <width> und <height> als Zahl
`\mbox{<text>}`

Box mit Rahmen

`\framebox[<width>][<pos>]{<text>}`
`\fbox{<text>}`

`\fboxrule` Linienbreite (default 0.4pt)
`\fboxsep` Abstand zwischen Rahmen und Inhalt (default 3pt)

`\raisebox{<lift>}[<height>][<depth>]{<text>}`

Par (paragraphs) Boxen

Inhalt wird im Paragraph Modus gesetzt

`\parbox[<pos>][<width>]{<text>}`

Position:

t oberste Zeile an Grundlinie ausgerichtet
m Mitte an Grundlinie ausgerichtet
b unterste Zeile an Grundlinie ausgerichtet

`\begin{minipage}[<pos>][<width>]`
text
`\end{minipage}`

Balken

`\rule[<lift>][<width>][<height>]`

Farbige Boxen

`\colorbox{<color>}{<text>}`
`\colorbox{<background_color>}{\textcolor{text_color}{<text>}}`
`\fcolorbox{<frame_color>}{<background_color>}{<text>}`

Das tcolorbox Paket

```
\begin{tcolorbox} [<optionen>]  
<inhalt>  
\tcblower  
<inhalt>  
\end{tcolorbox}
```

`\tcblower` erzeugt gestrichelte Trennlinie (optional)

Optionen:

<code>colback=<color></code>	Hintergrundfarbe
<code>colframe=<color></code>	Rahmenfarbe
<code>colbacktitle=<color></code>	Farbe des Titels
<code>coltext=<color></code>	Farbe des Textes
<code>title=<text></code>	Titel
<code>fontupper=</code>	Schriftart e.g. <code>\Large\bfseries</code>
<code>arc=<length></code>	Bogenkrümmung
<code>boxrule=<line width></code>	Linienstärke

Kapitel 3: Abbildungen und Tabellen

Bilder importieren

Das `graphicx` Paket

- Unterstützte Bildformate abhängig vom Treiber, den `graphicx` verwendet \Rightarrow wird automatisch je nach Compiler ausgewählt
 - `latex` Compiler kann EPS und PS importieren
 - `pdflatex` Compiler kann JPG, PNG und PDF importieren

```
\graphicspath{{dir1/}{dir2/}}  
\includegraphics[<options>]{<image>}
```

Optionen:

<code>width=xx</code>	Breite des Bildes
<code>height=xx</code>	Höhe des Bildes
<code>keepaspectratio</code>	Bild wird beim skalieren nicht verzerrt
<code>scale=xx</code>	Skalierungsfaktor
<code>angle=xx</code>	Rotation um <code>xx</code> Grad (gegen den Uhrzeigersinn)
<code>trim=l b r t</code>	Zurechtschneiden des Bildes (<code>trim = 10mm 80mm 20mm 5mm</code>)
<code>clip</code>	damit <code>trim</code> funktioniert

```
\reflectbox{} Bild spiegeln
```

Gleitobjekte (Floats)

- Optimierte Platzierung von Gleitobjekten für schönes Seitenlayout im Fließtextmodus
- Vermeidet verunstaltete Absätze und halb-leere Seiten

```
\begin{<float>[<placement>]  
<content>  
\end{<float>}
```

Platzierungsoptionen:

<code>h</code>	Gleitobjekt wird ungefähr an der aufgeführten Stelle im Text eingefügt
<code>t</code>	Platzierung oben auf der Seite
<code>b</code>	Platzierung unten auf der Seite
<code>p</code>	Platzierung auf spezieller Seite nur für Gleitobjekte
<code>!</code>	Überschreiben der internen LaTeX-Parameter
<code>H</code>	Platzierung an genau dieser Stelle. Erfordert das <code>float</code> Paket.

Gleitobjekte am unteren Seitenrand können unterhalb von Fußnoten erscheinen

⇒ Abhilfe schafft das Paket `\usepackage[bottom]{footmisc}`

Typische Fehlermeldungen:

- ! LaTeX Error: Not in outer par mode.
Ein Gleitobjekt wurde innerhalb einer anderen Umgebung verwendet (z.B. innerhalb von `minipage`, `parbox`, `tabular`, `framebox`, `table`, `figure`, mathematische Umgebungen)

Abbildungen

```
\begin{figure}[<placement>]
\includgraphics{<figure>}
\caption{text}
\label{key}
\end{figure}
```

Tabellen

```
\begin{table}[placement specifier]
\caption{text} \label{key}
\begin{tabular}{...}
... table data ...
\end{tabular}
\end{table}
```

Eigene Gleitobjekte definieren

Das float Paket

```
\newfloat{<type>}{<placement>}{<ext>}[<within>]
type          Name des neuen Gleitobjekt-Befehls
placement    default Platzierung (t, b, p, h) — kann bei jedem Aufruf überschrieben
              werden
ext          Dateiendung für Objektliste (entspricht lof, lot bei figure und table)
within       Nummerierung
```

`\floatname{<type>}{<floatname>}` Tatsächlicher Name des neuen Gleitobjekts

`\floatstyle{<style>}` wirkt sich auf alle folgenden `newfloat`-Befehle aus

⇒ um Name des Objekts auch in die Referenzierung zu bekommen:

```
\newcommand{\<type>autorefname}{<floatname>}
```

Style-Optionen:

<code>plain</code>	<code>normal</code>
<code>boxed</code>	Gleitobjekt umrahmt; Caption unterhalb des Rahmens
<code>ruled</code>	Caption über Gleitobjekt; Abgrenzung durch horizontale Linien

`\restylefloat{<type>}` um den neuen Style auch auf nicht neudefinierte Gleitobjekte anzuwenden (zum Beispiel auch für `figure` und `table`)

Platzierung von Gleitobjekten

- für zweispaltigen Text: `table*` und `figure*` Umgebungen überspannen beide Spalten
- `\label{<key>}` nach `\caption{}`; kann mittels `\ref{<key>}` oder `\pageref{<key>}` referenziert werden
- ohne `\caption{}` Befehl entsteht unnummerierte Abbildung/Tabelle
- mehrere `\caption{}` Befehle erzeugen mehrere Abbildungen/Tabellen innerhalb eines Gleitobjekts
- `\caption[<short>]{<long>}` optionales Argument für kürzeren Text in `\listoffigures` oder `\listoftables`
- Zähler `figure` and `table` können über `\setcounter{}{}` verändert werden
- Gleitobjekte können nicht innerhalb einer `parbox` oder `minipage` verwendet werden

`\clearpage` fügt alle Gleitobjekte und einen Seitenumbruch ein
`\FloatBarrier` fügt alle Gleitobjekte ein ohne Seitenumbruch (`placeins` Paket)

Das `placeins` Paket:

`\usepackage[section]{placeins}` `\FloatBarrier` Befehl vor jeder `section`
`\usepackage[below]{placeins}` ähnlich, aber weniger restriktiv

Platzierungsparameter anpassen:

`\setcounter{<parameter>}{<number>}`
`topnumber` max Anzahl an Gleitobjekten am oberen Seitenrand (default 2)
`bottomnumber` max Anzahl an Gleitobjekten am unteren Seitenrand (default 1)
`totalnumber` max Anzahl an Gleitobjekten pro Seite (default 3)

Gleitobjekt-Text-Verhältnis anpassen:

`\renewcommand{}{}`
`\textfraction` minimaler Anteil einer Textseite, der mit Text gefüllt sein muss (default 0.2)
`\topfraction` maximaler Anteil des oberen Teils der Seite, der mit Gleitobjekten gefüllt sein darf (default 0.7)

`\bottomfraction` maximaler Anteil des unteren Teils der Seite, der mit Gleitobjekten gefüllt sein darf (default 0.3)
`\floatpagefraction` minimaler Anteil einer Gleitobjektseite, der mit Gleitobjekten gefüllt sein **muss** (default 0.5)

Das `wrapfig` Paket

```
\begin{wrapfigure} [<numberoflines>] [<placement>] [<overhang>] [<width>]  
...  
\end{wrapfigure}
```

Ebenso gibt es eine `wraptable` Umgebung

`<numberoflines>` Anzahl Zeilen die überspannt werden sollen
`<overhang>` Über Texttrand hinaus

Platzierungsoptionen: Großbuchstaben \Rightarrow Gleitobjekt; Kleinbuchstaben \Rightarrow exakte Platzierung

r R	rechts
l L	links
i I	innen (für twoside)
o O	außen (für twoside)

Beschriftung von Gleitobjekten

Das `subcaption` Paket

```
\begin{subfigure} [<pos>] [<width>] ... \caption{<text>} \end{subfigure}  
\begin{subtable} [<pos>] [<width>] ... \caption{<text>} \end{subtable}
```

```
\subcaption{<text>}
```

Das `caption` Paket

ermöglicht zusätzliche `\caption*{...}` neben der `\caption{...}`
(z.B. als Tabellen-Legende)

Das `sidecap` Paket

Beschriftung rechts/links vom Gleitobjekt

```
\begin{SCfigure} [<relativecaptionwidth>] [<placement>] ... \end{SCfigure}  
\begin{SCtable} [<relativecaptionwidth>] [<placement>] ... \end{SCtable}
```

`relativecaptionwidth` Breite der Beschriftung im Verhältnis zur Größe des Gleitobjekts

Paketoptionen:

<code>rightcaption / leftcaption</code>	Beschriftung rechts / links
<code>outercaption / innercaption</code>	Beschriftung außen / innen
<code>wide</code>	Gleitobjekt darf sich über den Rand hinaus erstrecken
<code>raggedright / raggedleft / ragged</code>	Bessere Textalignment bei schmalen Beschriftungen

Tabellen

`\begin{tabular}[<position>]{<spalten> <zeilen> \end{tabular}`

Positionsparameter:

	vertikale Positionierung der Tabelle
<code>t</code>	oberer Rand mit Baseline aligniert
<code>b</code>	unterer Rand mit Baseline aligniert
<code>c</code>	zentriert

Spalten

ein Argument pro Spalte

<code>l</code>	linksbündige Spalte
<code>r</code>	rechtsbündige Spalte
<code>c</code>	zentrierte Spalte
<code>p{<width>}</code>	Paragraph Spalte mit fester Breite (linksbündig)
<code> </code>	vertikale Linie (auch doppelt möglich)
<code>@{<text>}</code>	text zwischen den Spalten z.B. <code>@{\hspace{width}}</code> oder <code>r@{.}l</code> für Dezimalstellen
<code>*{<num>}{<spalten>}</code>	mehrere gleiche Spalten nacheinander, z.B. <code>*{15}{c}</code>

Spalten-Modifier:

Wirken sich auf gesamte Spalte aus

<code>>{<command>}</code>	Befehle, die vor dem Spalteninhalt ausgeführt werden
<code><{<command>}</code>	Befehle, die nach dem Spalteninhalt ausgeführt werden

Wenn Alignment der Spalten geändert wird (z.B. `\raggedright`), wird `\arraybackslash` benötigt, um die Funktion von `\` wiederherzustellen (array Paket)

Das array Paket

ermöglicht Paragraph Spalten

`p{<width>}` Paragraph Spalte mit Text oben aligniert

`m{<width>}` Paragraph Spalte mit Text vertikal mittig aligniert

`b{<width>}` Paragraph Spalte mit Text unten aligniert

Das tabularx Paket

Eigene Spaltentypen definieren

```
\newcolumntype{<name>}[<arg>]{<column definition>}
```

tabularx Umgebung

X Spalten werden auf Breite der Tabelle ausgedehnt

```
\begin{tabularx}{<width>}{ |X|X|X|X| } ... \end{tabularx}
```

The tabulary Paket

```
\begin{tabulary}{<width>}{<col>} ... \end{tabulary}
```

L linksbündig, balancierte Spalte

C zentrierte, balancierte Spalte

R rechtsbündig, balancierte Spalte

J Blocksatz, balancierte Spalte

Das dcolumn Paket

Dezimalstellen `D{<separator>}{<output separator>}{<decimal places>}`

`D{.}{,}{-1}` -1 steht hier für beliebig viele Nachkommastellen

Typische Fehlermeldungen:

- Extra alignment tab has been changed to `\cr`.
mehr Spalten verwendet als definiert wurden. Evtl wurde ein nicht maskiertes `&` Zeichen verwendet.
- ! LaTeX Error: Missing p-arg in array arg.
Bei p Spalte fehlt die Breitenangabe
- ! LaTeX Error: Missing @-exp in array arg.
Dem @ Spalten Element fehlt das Argument
- ! LaTeX Error: Illegal character in array arg.
Innerhalb der Spaltendefinition einer Tabelle wurde ein unzulässiger Buchstabe verwendet

Zeilen

<code>&</code>	Separator zwischen den Spalten
<code>\\</code>	neue Zeile
<code>\hline</code>	horizontale Linie
<code>\cline{i-j}</code>	horizontale Linie von Spalte i bis Spalte j
<code>\vline</code>	vertikale Linie der Höhe der Zeile

Zellen verbinden

Zellen über Spalten hinweg verbinden

```
\multicolumn{num_col}{alignment}{content}
```

Das `multirow` Paket

```
\multirow{<num_rows>}{<width>}{<content>}
```

* natürliche Breite übernommen

folgende Zeilen müssen einen leeren Eintrag für die jeweiligen "überspannten" Zellen enthalten

```
\renewcommand{\multirowsetup}{\centering}     default: \raggedright
```

Tabellen individualisieren

Tabellenparameter ändern:

```
\setlength{}{}
```

```
\tabcolsep             Hälfte des Abstands zwischen den Spalten
```

```
\arrayrulewidth     Breite der vertikalen und horizontalen Linien
```

```
\doublerulesep     Abstand zwischen zwei Linien
```

```
\renewcommand{\arraystretch}{<factor>}     Abstand zwischen den Zeilen
```

Das `booktabs` Paket: für schönere Linien

```
\toprule
```

```
\midrule
```

```
\bottomrule
```

Das `colortbl` Paket

- Für farbige Tabellen
- Kann auch über die Option `table` mit dem `xcolor` Pakte geladen werden

```
\rowcolors{<starting_row>}{<odd_color>}{<even_color>}
```

vor `tabular` Umgebung definieren

```
\hiderowcolors             deaktiviert Tabellen Färbung; am Anfang der Zeile
```

```
\showrowcolors             aktiviert Tabellen Färbung; am Ender der Zeile
```

`\cellcolor{color}`

`\rowcolor{color}` farbige Zeile
`>\columncolor{<color>}` farbige Spalte

`\arrayrulecolor[<model>]{<color-spec>}` Linienfarbe

Große Tabellen

Das `longtable` Paket

statt `\begin{tabular} ... \end{tabular}`

```
\begin{longtable}
... \endfirsthead ... \endhead ... \endfoot ... \endlastfoot
\end{longtable}
```

`\endfirsthead` Tabellenkopf auf erster Seite
`\endhead` Tabellenkopf auf den Folgeseiten
`\endfoot` Tabellenfuß auf allen Seiten
`\endlastfoot` Tabellenfuß auf der letzten Seite

Das `pgfplotstable` Packet

Tabellen aus `.csv`-Dateien importieren

<https://www.latex-tutorial.com/tutorials/pgfplotstable/>

Tabellenkalkulation

Das `spreadtab` Paket

```
\usepackage{spreadtab}
```

```
\begin{spreadtab}{{<tabletype>}{<parameters>}}
...
\end{spreadtab}
```

Tabellenzelle wird durch das Paar `<colref><rowref>` (absolut) oder `[<horizontal offset>,<vertical offset>]` (relativ) identifiziert

Textfeld durch `@` kennzeichnen

Gemischte Zellen: `<text> :={\<number>} <text>`

Rotation

Das rotating Paket

<code>\begin{sideways} ... \end{sideways}</code>	Drehung 90 Grad gegen den Uhrzeigersinn
<code>\begin{turn}{<angle>} ... \end{turn}</code>	Drehung um <angle> Grad; fügt notwendigen Abstand ein
<code>\begin{rotate}{<angle>} ... \end{rotate}</code>	Drehung um <angle> Grad; ohne extra Abstand
<code>\begin{sidewaysfigure} ... \end{sidewaysfigure}</code>	
<code>\begin{sidewaystable} ... \end{sidewaystable}</code>	

werden auf extra Seite gedruckt

Paketoptionen:

<code>counterclockwise</code>	in einseitigen Dokumenten; gegen den Uhrzeigersinn
<code>anticlockwise</code>	Synonym für <code>counterclockwise</code>
<code>clockwise</code>	in einseitigen Dokumenten; im Uhrzeigersinn
<code>figuresright</code>	in doppelseitigen Dokument; alle auf rechter Seite
<code>figuresleft</code>	in doppelseitigen Dokument; alle auf linker Seite

Das rotfloat Paket

ermöglicht H als Platzierungsargument zu verwenden

```
\begin{sidewaystable}[H] ... \end{sidewaystable}
```

Kapitel 4: Mathematik und Informatik

AMS (American Mathematical Society)

`\usepackage{amsmath}` für mathematische Formeln

`\usepackage{amssymb}` für mathematische Symbole

Mathe Umgebungen

- LaTeX setzt die mathematische Notation anders als normalen Text
- die eingegebenen Buchstaben und Symbole werden als Teil einer Formel interpretiert
- Buchstaben werden in Formeln grundsätzlich als Namen von Variablen interpretiert und daher kursiv gedruckt
- Leerzeichen innerhalb von Formeln werden ignoriert und müssen bei Bedarf manuell mittels Abstands-Anweisungen wie `\,` oder `\;` gesetzt werden
- Leerzeilen sind in Matheumgebungen nicht erlaubt
- LaTeX bestimmt Abstand innerhalb von Formeln anhand eigener interner Regeln

Modus		Umgebung	LaTeX Kurzform	TeX Kurzform
Text / inline	Textformeln werden inline innerhalb des Textes angezeigt	<code>\begin{math}</code> <code>\end{math}</code>	<code>\(...\)</code>	<code>\$...\$</code>
Display	Formel als eigenständiger Absatz und zentriert	<code>\begin{displaymath}</code> <code>\end{displaymath}</code>	<code>\[...\]</code>	<code>\$\$...\$\$</code>
Equation	Fügt auch Nummerierung hinzu	<code>\begin{equation}</code> <code>\end{equation}</code>		

Kommas in Mathe Umgebungen sind Aufzählungskommas
Dezimalkommata müssen mit `{,}` gesetzt werden

Umgebungen für Gleichungen

jede der Umgebungen jeweils als nummerierte und unnummerierte Variante (*) verfügbar
Erzeugen automatisch den Mathemodus

`equation` Eine Formel, einzeilig, einfach nummeriert
Kein Zeilenumbruch möglich

`gather` Mehrere Formeln, eine Formel pro Zeile, mehrfach nummeriert
Zeilenumbruch erfolgt mit `\\`

`multline` Eine Formel verteilt auf mehrere Zeilen; einfach nummeriert
Zeilenumbruch erfolgt mit `\\`
erste Zeile nach links ausgerichtet; zweite Zeile nach rechts ausgerichtet
bei mehr als zwei Zeilen werden die mittleren Zeilen zentriert

`align` Eine Formel verteilt auf mehrere Zeilen, einfach nummeriert
Ausrichtung erfolgt in zwei Spalten durch `&` (möglichst nach dem `=`)
Zeilenumbruch erfolgt mit `\\`

`\nonumber \\` unterdrückt Nummerierung für diese Zeile
`\numberwithin{equation}{section}` Abschnittsweise Nummerierung

Alle nummerierten Umgebungen für Gleichungen können mit einem `\label{<key>}` versehen werden und im Text mittels `\ref{<key>}` oder `\eqref{<key>}` (Gleichungsnummer in runde Klammern gesetzt) referenziert werden.

Typische Fehlermeldungen:

- `! Missing $ inserted.`
→ z.B. wenn Leerzeilen in eine Matheumgebung eingefügt wurden

Matrizen

`\begin{array}{ccc} ... \end{array}`

`\begin{matrix}`
`... & ... & ... \\`
`... & ... & ... \\`
`\end{matrix}`

`vmatrix` `||`
`Vmatrix` `|||`
`pmatrix` `()`
`bmatrix` `[]`
`Bmatrix` `{ }`

`\bordermatrix{}`

`\begin{smallmatrix} ... \end{smallmatrix}`

`\begin{cases} ... \end{cases}`

Matrizen im Text
für Fallunterscheidung

Schriften und Symbole

Symbole in LaTeX

Die wichtigsten: [LATEX Mathematical Symbols](#)
alle: [The Comprehensive LATEX Symbol List](#)

griechische Buchstaben: `\alpha` `\Alpha` ...

Schrift im Mathemodus

Schriftformatierung im Mathemodus

<code>\mathrm{...}</code>	Roman
<code>\mathit{...}</code>	kursiv
<code>\mathbf{...}</code>	fett
<code>\mathsf{...}</code>	serifenlos
<code>\mathtt{...}</code>	Typewriter
<code>\mathcal{...}</code>	Calligraphy (nur für Großbuchstaben)
<code>\mathbb{...}</code>	"Tafel" Buchstaben
<code>\mathfrak{...}</code>	Frakturen
<code>\mathscr{...}</code>	benötigt <code>\usepackage{mathrsfs}</code>

Text im Mathemodus

Text muss im Mathemodus extra gekennzeichnet werden

`\mbox{<text>}` oder `\text{<text>}`

Formeln

Griechische Buchstaben

`\alpha` `\beta` `\gamma` `\rho` `\sigma` `\delta` `\epsilon`

Relations-Operatoren

`<` `>` `\subset` `\supset` `\subseteq` `\supseteq`

Binäre Operatoren

`\bigoplus` `\bigotimes` `\bigodot` `\bigcup` `\bigcap` `\biguplus` `\bigsqcup`
`\bigvee` `\bigwedge`

Weitere Funktionen

`\sum` `\prod` `\coprod`
`\int` `\oint` `\iint` `\iiint` `\idotsint`

`\arccos \arcsin \arctan \arg \cos \cosh \cot \coth \csc \deg \det \dim`
`\exp \gcd \hom \inf \ker \lg \lim \liminf \limsup \ln \log \max \min`
`\Pr \sec \sin \sinh \sup \tan \tanh \to \infty`

`\substack` ermöglicht Zeilenumbruch via `\\` um Grenzen (von...bis) über mehrere Zeilen zu schreiben

`\frac{<numerator>}{<denominator>}`
`\cfrac{}{}` für verschachtelte Brüche
`\binom{}{}
\sqrt[<n>]{<expression>}
\neq
\limits setzt Grenzen in z.B. Brüchen auch über/unter die Summe,...`

Hoch-/Tiefstellen:

`x_{y}`
`x^{y}`

Über-/Untersatz

`\overset{}{}
\underset{}{}
\alpha, \beta, \Gamma, \gamma, ...`

Griechisches Alphabet

Auslassungspunkte:

`\ldots` Auslassungspunkte an der Grundlinie des Texts ausgerichtet
`\cdots` Auslassungspunkte an der Mittellinie des Texts ausgerichtet
`\dotsc` Auslassungspunkte zwischen Kommas
`\dotsb` Auslassungspunkte für Operatoren
`\dotsm` Auslassungspunkte in Multiplikationen
`\dotsi` Auslassungspunkte für Integrale

Klammern:

`\Biggl(\biggl(\Bigl(\bigl(\bigr) \Bigr) \biggr) \Biggr)`
`\Biggl| \biggl| \Bigl| \bigl| \bigr| \Bigr| \biggr| \Biggr|`
`\Biggl\{ \biggl\{ \Bigl\{ \bigl\{ \bigr\} \Bigr\} \biggr\} \Biggr\}`

oder Latex selbst die Größe entscheiden lassen:

`\left| \right|`
`\left(\right)`
`\left\{ \right\}`

Eigene Operatoren definieren

```
\DeclareMathOperator{\<operatorname>}{\<operatordefinition>}
```

Theoreme

- andere Schriftart, um sie vom umgebenden Text zu unterscheiden
- Name und Nummer für die nachfolgende Referenz

Theorem-Umgebung definieren: (Lemma, Aufgaben, ...)

```
\newtheorem{\<mytheorem>}[\<zaehler>]{\<bezeichnung>}[\<gliederung>]
```

<zaehler> Name einer bereits definierten Theoremumgebung, falls der Zähler der gleiche sein soll

<gliederung> optionales Argument für die Nummerierung z.B. section

```
\begin{\<mytheorem>}[\<note>] ... \end{\<mytheorem>}
```

Das amsthm Paket

enthält mehrere Styles für Theoreme

```
\theoremstyle{\<stylename>}
```

alle nachfolgend definierten Theoreme verwenden diesen Stil

Styles:

plain	üblich für Theoreme, Lemmas, Sätze, ...	Theorem 1. <i>Theorem text.</i>
definition	üblich für Definitionen, Beispiele	Definition 2. Definition text.
remark	üblich für Anmerkungen	<i>Remark 3.</i> Remark text.

eigenes Styles definieren:

alle Argumente, die leer gelassen werden, nehmen default-Werte an

```
\newtheoremstyle
```

{name}%	Name des neuen Stils
{abovespace}%	vertikaler Abstand zum vorherigen Text
{belowspace}%	vertikaler Abstand zum folgenden Text
{bodyfont}%	Schriftart für den Text (z.B. <code>\scshape</code> oder <code>\bfseries</code>)
{indent}%	Einrücken des Kopfes
{headfont}%	Schriftart für den Kopf
{headpunct}%	Zeichensetzung nach dem Kopf
{headspace}%	(horizontaler) Abstand zwischen Kopf und Text
{headspec}%	individueller Kopf; enthält <code>\thmname{#1}</code> , <code>\thmnumber{ #2}</code> und <code>\thmnote{ #3}</code>

Beweise

```
\begin{proof}[\<name>] ... \end{proof}
```

Informatik

Gleitobjekt für Algorithmen

Das `algorithm` Paket

```
\usepackage[<numbering>]{algorithm}  
\begin{algorithm} ... \end{algorithm}
```

<numbering> Nummerierung der Gleitobjekte nach `section`, `chapter` etc
default ist fortlaufend über das gesamte Dokument

`\listofalgorithms` erzeugt ein Verzeichnis für die `algorithm`-Umgebungen

Pseudocode Umgebung

Das `algpseudocode` Paket

```
\begin{algorithmic}[<step>] ... \end{algorithmic}
```

<step> Abstände zwischen Zeilennummern

Grundlegende Befehle der Pseudocode-Umgebung:

```
\State <text>  
\If{<condition>} \State{<text>} \Else \State{<text>} \EndIf  
\For{<condition>} \State{<text>} \EndFor  
\For{<condition> \To <condition> } \State{<text>} \EndFor  
\ForAll{<condition>} \State{<text>} \EndFor  
\While{<condition>} \State{<text>} \EndWhile  
\Repeat \State{<text>} \Until{<condition>}  
\Loop \State{<text>} \EndLoop  
\Require <text>  
\Ensure <text>  
\Return <text>  
\Print <text>  
\Comment{<text>}  
\And, \Or, \Xor, \Not, \To, \True, \False
```

Sourcecode

Formatierter Text

```
\begin{verbatim} ... \end{verbatim}
```

LaTeX Befehle werden ignoriert; Leerzeichen werden alle gedruckt

Das `listings` Paket

```
\begin{lstlisting}[<options>]
...
\end{lstlisting}
```

Quelltext aus Datei importieren

```
\lstinputlisting[language=<sprache>, firstline=<start>,
lastline=<stop>]{<source-file>}
```

Code Darstellung individualisieren

Mögliche Einstellungen:

<code>\lstset{</code>	
<code>backgroundcolor=\color{white},</code>	Hintergrundfarbe
<code>basicstyle=\footnotesize,</code>	Schriftgröße
<code>commentstyle=\color{mygreen},</code>	Kommentare
<code>escapeinside={\%*}{*},</code>	um LaTeX Code hinzuzufügen
<code>extendedchars=true,</code>	Ermöglicht die Verwendung von nicht-ASCII-Zeichen (nur für 8-Bit-Codierungen)
<code>breaklines=true,</code>	automatische Zeilenumbrüche
<code>keepspaces=true,</code>	Bewahrt Leerzeichen (Code Einrückung)
<code>title=\lstname,</code>	Dateinamen angeben
<code>captionpos=b,</code>	Untertitel
<code>frame=single,</code>	Rahmen
<code>rulecolor=\color{black},</code>	Rahmenfarbe
<code>language=Octave,</code>	Sprache des Codes
<code>keywordstyle=\color{blue},</code>	Stil der Keywords
<code>deletekeywords={...},</code>	Keywords der Sprache löschen
<code>otherkeywords={*,...},</code>	weitere Keywords hinzufügen
<code>stringstyle=\color{mymauve},</code>	Stil der strings
<code>numbers=left,</code>	Platzierung der Zeilennummerierung (none, left, right)
<code>numbersep=5pt,</code>	Abstand zwischen Zeilennummerierung und Code
<code>numberstyle=\tiny\color{mygray},</code>	Stil der Zeilennummerierung
<code>stepnumber=2,</code>	Abstand zwischen Zeilennummern
<code>showspaces=true,</code>	Leerzeichen durch Unterstriche kennzeichnen
<code>showstringspaces=false,</code>	nur innerhalb von strings
<code>showtabs=false,</code>	Tabs in Strings durch Unterstriche kennzeichnen
<code>tabsize=2</code>	Tabgröße (Anzahl Leerzeichen)
<code>}</code>	

Das `minted` Paket

- benötigt das Programm Pygments (Syntax-Highlighter) <https://pygments.org/>
- In Overleaf funktioniert das Paket “out of the box”

```
\begin{minted} [<optionen>] {<sprache>}  
...  
\end{minted}
```

```
\inputminted [<optionen>] {<sprache>} {<datei>}
```

Optionen

`frame=lines`

`framesep=2mm`

`baselinestretch=1.2`

`bgcolor=<color>`

`fontsize=\<fontsize>`

`linenos`

`mathescape`

ermöglicht Mathe-Modus in Kommentaren

`rulecolor=<color>`

`showspaces`

`firstline=<linenumber>`

nur für `\inputminted`

`lastline=<linenumber>`

nur für `\inputminted`

```
\usemintedstyle{<style>}
```

Gleitobjekt

```
\begin{listing}
```

...

```
\caption{<caption>}
```

```
\label{<key>}
```

```
\end{listing}
```

```
\renewcommand\listoflistingscaption{List of source codes}
```

```
\listoflistings
```

Kapitel 5: Großprojekte

Große Projekte verwalten

chapter1.tex, chapter2.tex, ... einzelne Dateien je Kapitel
main.tex enthält den "Rahmen"

`\input{<chapter1>}` liest die Datei `chapter1.tex` an dieser Stelle
funktioniert auch rekursiv
`\include{<chapter1>}` liest die Datei `chapter1.tex` an dieser Stelle
funktioniert nicht rekursiv
kann mit `\includeonly` kombiniert werden
`\includeonly{<chapter5>}` in die Präambel
kompiliert nur die Datei `chapter5.tex` neu

`\include` unterscheidet sich wie folgt von `\input`:

- kann nur in der Hauptdatei verwendet werden (funktioniert nicht rekursiv)
- `\input` kann in `\include` Datei verwendet werden
- funktioniert wie `\clearpage \input{} \clearpage`
- kein Fehler, wenn Datei nicht vorhanden
- `\include` erzeugt eigene Hilfsdatei im jeweiligen Ordner der eingelesenen Datei

`\input` und `\include` können auch dazu benutzt werden, Befehle oder Teile der Präambel in einer gesonderten Datei zu definieren

Das `pdfpages` Paket

Optionen: `final`, `draft`

`\includepdf[<options>]{<filename>}`
`\includepdfmerge[<options>]{<file-page-list>}`

Optionen:

`pages={3, {}, 5-7}` Seiten, die importiert werden sollen; `{}` für leere Seite
`angle=90` Dreht die Seiten um 90 Grad
`addtolist={page number, type, heading, label}`
Macht eingefügte Seite zum entsprechenden Gleitobjekt
`pagecommand` Befehl, der auf den eingefügten Seiten ausgeführt wird

Buchstruktur

```
\documentclass[<options>]{scrbook}
```

default Optionen: letterpaper, 10pt, twoside, onecolumn, final, openright

Aufbau

<code>\begin{document}</code>	
<code>\frontmatter</code>	römische Seitenzahlen
<code>\maketitle</code>	Titel
<code>\chapter{Preface}</code>	unnummerierte Kapitel möglichst keine sections einfügen
<code>\tableofcontents</code>	Inhaltsverzeichnis
<code>%</code>	
<code>\mainmatter</code>	arabische Seitenzahl; fängt wieder mit Seite 1 an
<code>\chapter{First chapter}</code>	alle Kapitel, die den tatsächlichen Inhalt enthalten
<code>...</code>	
<code>%</code>	
<code>\appendix</code>	Anhang; Kapitel und Sections werden als Appendix nummeriert (A,...)
<code>\chapter{First Appendix}</code>	
<code>%</code>	
<code>\backmatter</code>	unnummerierte Kapitel
<code>\chapter{Last note}</code>	
<code>\bibliographystyle{abbrv}</code>	
<code>\bibliography{buch}</code>	Literaturverzeichnis

Titelseiten gestalten

Eigene Titelseite gestalten (anstelle von `\maketitle`)

```
\thispagestyle{empty}
```

... selbstgestaltete Titelseite ...

```
\cleardoublepage
```

Seitenumbruch, nächste Seite beginnt rechts (wenn nötig Leerseite)

Hyperlinks und Metadaten

Das hyperref Paket

- Links und Verweise innerhalb von PDF Dokumenten
- funktioniert nur für pdflatex
- Änderung von Einstellungen innerhalb des PDF Dokumentes
- übernimmt bereits bestehende Verweise (`\label` und `\ref` System, Überschriften, das Literaturverzeichnis, etc.)
- möglichst als letztes Paket einbinden (aber noch vor `geometry`)

```
\usepackage[<options>]{hyperref}
\hypersetup{<options>}
```

Optionen:

Option	Beschreibung	Mögliche Werte	Default Wert
Dokument Metadaten			
pdftitle=<text>	Titel		
pdfauthor=<text>	Autor bzw. Autoren		
pdfkeywords=<text>	Stichwörter		
pdfsubject=<text>	Um was geht es		
Link Optionen:			
colorlinks=<boolean>	farbige Linktext statt Rahmen		false
linkcolor=<color>	Dokument-interne Links		red
anchorcolor =<color>	Link Text		black
citecolor=<color>	Literaturverweise		green
filecolor=<color>	lokale Dateien		magenta
pagecolor=<color>	andere Seiten		red
urlcolor=<color>	externe URL Links		cyan
frenchlinks=<boolean>	small caps, statt Farbe		false
Optionen Ansicht:			
bookmarks=<boolean>	erstellt Bookmarks (Inhaltsverzeichnis)		false
bookmarksopen=<boolean>	Inhaltsverzeichnis beim Öffnen der Datei öffnen		false
pdfstartview=<value>	Ansicht beim Öffnen (z.B. an Fenstergröße anpassen)	Fit, FitH, FitV, FitR, FitB, FitBH, FitBV	FitB
pdfpagelayout=<value>	Seitenlayout beim Öffnen	SinglePage, OneColumn, TwoColumnLeft, ...	empty

Nur Beispiele; weitere Optionen in Paketbeschreibung

bessere Referenzierung mit `hyperref`

`\autoref{<key>}` gibt nicht nur Nummerierung sondern auch Typ des referenzierten Objektes wieder (z.B. Abbildung)
`\nameref{<key>}` gibt nicht die Nummerierung sondern den Namen des referenzierten Objektes wieder (z.B. die Abschnittsüberschrift)

Anpassung der Standardnamen für `\autoref`:

```
\renewcommand{\figureautorefname}{<text>}
```

Literaturverzeichnis

```
\begin{thebibliography}{<widest-label>}  
\bibitem{<key1>} <reference1>  
\bibitem{<key2>} <reference2>  
\end{thebibliography}
```

`<widest-label>` nicht die Zahl selbst ist der Parameter, sondern die Anzahl der Stellen.

```
\cite{<key>}  
\cite[<text>]{<key>}  
\cite{<key1>,<key2>}
```

Literaturverwaltung mit JabRef

automatisch generierte Keys:

<code>[auth]</code>	Nachname des Erstautors
<code>[authorLast]</code>	Nachname des Letztautors
<code>[shorttitle]</code>	die ersten drei Wörter des Titels
<code>[veryshorttitle]</code>	das erste Wort des Titels
<code>[year]</code>	Publikationsjahr
<code>[shortyear]</code>	die letzten beiden Ziffern des Jahres

<code>:abbr</code>	Abkürzung
<code>:lower</code>	Kleinbuchstaben
<code>:upper</code>	Großbuchstaben

Literaturverzeichnis mit BibTeX

```
\bibliographystyle{<citation_style>}  
\bibliography{<bibliography_file>}
```

Laden mehrerer Dateien möglich

Zitationsstile:

- Format der Abkürzung im Fließtext: z.B. Nummern, Text,...
- Format der Einträge im Literaturverzeichnis: z.B. sortiert, unsortiert, Abkürzungen,...

abbrv, acm, alpha, apalike, ieetr, plain, siam, unsrt, ...

eigene bibtex Styles generieren:

makebst

Hilfsprogramm um interaktiv benutzerdefinierte Bibliographiestildateien zu erstellen

Erzeugen des Dokuments:

pdflatex <mydoc>	sammelt welche Literatur wurde verwendet (generiert .aux Datei)
bibtex <mydoc>	thebibliography Umgebung wird im eingestellten Stil erstellt
pdflatex <mydoc>	Literaturverzeichnis wird dem Dokument hinzugefügt
pdflatex <mydoc>	Verweise im Text werden dem Dokument hinzugefügt

\nocite{} Eintrag im Verzeichnis ohne Referenz im Text

\nocite{*} Alle Einträge (auch ohne Referenz) aus .bib Datei ins Literaturverzeichnis

Das natbib Paket

Für mehr Einstellungen am Zitierformat

```
\usepackage[options]{natbib}
```

Optionen:

round	()
square	[]
curly	{}
angle	<>
semicolon	Referenzen mit Semikolon getrennt
comma	Referenzen mit Komma getrennt
authoryear	Autor-Jahr Referenzen
numbers	Numerische Referenzen
super	hochgestellte Referenzen
sort	mehrere Referenzen sortiert [1,5,6,7]
sort&compress	und zusätzlich zusammengezogen [1,5-7]
compress	nur zusammengezogen (unüblich)
sectionbib	mehrere Literaturverzeichnisse

Format Literaturverzeichnis:

plainnat, abbrvnat, unsrnat, ...

Zitierbefehle:

<code>\citet[<davor>][<dahinter>]{<key>}</code>	Alex et al. (<davor> 1991, <dahinter>)
<code>\citep[<davor>][<dahinter>]{<key>}</code>	(<davor> Alex et al., 1991, <dahinter>)
<code>\citet*{<key>}</code>	Alex, Mathew, and Ravi (1991)
<code>\citep*{<key>}</code>	(Alex, Mathew, and Ravi, 1991)

Format Zitate im Dokument:

<code>\bibpunct[<nr1>]{<nr2>}{<nr3>}{<nr4>}{<nr5>}{<nr6>}{<nr7>}</code>	
<nr1>	Zeichen vor dahinter Information
<nr2>	Klammer auf
<nr3>	Klammer zu
<nr4>	Zeichen zwischen mehreren Zitaten (Komma, Semikolon,...)
<nr5>	Referenz: n - numerisch; s - superscript; a - author-year sonst
<nr6>	Zeichen zwischen Autor und Jahr
<nr7>	Zeichen zwischen mehreren Zitaten, wenn gleicher Autor

Das biblatex Paket

Paketoptionen:

<code>style=numeric,</code>	
<code>citestyle=numeric-comp,</code>	
<code>sorting=none,</code>	
<code>backend=bibtex,</code>	
<code>maxcitenames=1,</code>	Anzahl Namen im Text
<code>maxbibnames=3,</code>	Anzahl Namen im Verzeichniss
<code>minbibnames=3,</code>	Anzahl Namen im Verzeichniss
<code>natbib=true,</code>	um natbib Zitierbefehle zu verwenden
<code>url=false,</code>	
<code>isbn=false,</code>	
<code>doi=false,</code>	
<code>autocite = superscript,</code>	
<code>giveninits=true</code>	

<code>\addbibresource{<bibfile>}</code>	Datei mit Literaturdatenbank (kann schon in Präambel aufgerufen werden)
<code>\printbibliography</code>	an dieser Stelle wird das Literaturverzeichnis erzeugt

Literaturverzeichnis mit biber

- Alternative zu BibTeX
- Literaturverzeichnisprozessor speziell für biblatex
- verwendet BibTeX nur zum Sortieren, nicht aber zum Formatieren
- benötigt keine .bst Datei

- vollständige Unicode-Unterstützung, die mit BibTeX nur schwer zu erreichen ist
- löst alle BibTeX-Probleme (richtige Sortierung da Unicodeunterstützung, Speicherbedarf, Kodierungen etc.)
- mehrere Bibliographielisten möglich
- manche LaTeX-Pakete hängen explizit von BibTeX ab und funktionieren nicht mit biber
- statt natbib muss hier das Paket biblatex verwendet werden
- in <file>.tex muss biblatex mit backend=biber geladen sein, damit foo.bcf geschrieben wird
- <file>.bcf steht für „biber control file“ und enthält Anweisungen (welche bib-Datei, welche Sortierung usw.)

```
\usepackage[backend=biber]{biblatex}  
\addbibresource{<bibfile>}  
\printbibliography
```

Abkürzungsverzeichnis

Das acronym Paket

Optionen:

footnote	Langform als Fußnote ausgeben
nohyperlinks	verhindern der automatischen Verlinkung durch hyperref
printonlyused	nur Abkürzungen auflisten, die tatsächlich verwendet werden
smaller	verkleinert Anzeige der Abkürzung im Dokument
dua	Ausgabe der Abkürzung erfolgt immer in Langform
nolist	kein Verzeichnis aller Abkürzungen erstellt

```
\begin{acronym}[<längsteKurzform>]  
\acro{<Kürzel>}[<Kurzform>]{<Langform>}  
\end{acronym}
```

Im Text:

\ac{<Kürzel>}	erste Verwendung wird ausgeschrieben, alle weiteren nicht
\acs{<Kürzel>}	Kurzform
\acf{<Kürzel>}	Langform
\acl{<Kürzel>}	Langform ohne Kurzform in Klammern
\acp{<Kürzel>}	Plural (funktioniert auch für \acsp, \acfp, \aclp) → wird immer mit s ergänzt; alternativ: \acroplural{<Kürzel>}[<KurzPlural>]{<LangPlural>}

Index anlegen

```
\usepackage{makeidx}  
\makeindex      (muss in Präambel aufgerufen werden)
```

Indexeinträge werden im Text erstellt durch:

```
\index{<Bezeichnung>}  
\index{<Bezeichnung>!<Unterbezeichnung>}  
\index{<Bezeichnung>!<Unterbezeichnung>!<Unterunterbezeichnung>}
```

erzeugt einen Eintrag <Bezeichnung> im Index, der auf die aktuelle Seite verweist

Index ausgeben

```
\printindex
```

Erzeugen des Dokuments:

Ähnlich zum Erzeugen des Literaturverzeichnisses:

```
makeindex <mydoc>      Anlegen der .idx Datei
```

Kapitel 6: Spezielle LaTeX Dokumente

Briefe mit LaTeX

Die g-brief Klasse

```
\documentclass[<optionen>]{g-brief}  
\Name{ }  
\Strasse{ }  
...  
  
\trennlinien  
\lochermarke  
\faltmarken  
\fenstermarken
```

Lebenslauf mit LaTeX

Lebenslauf frei gestalten

- Lebensläufe können in LaTeX frei gestaltet werden
- eine geeignete Klasse ist beispielsweise `scrartcl`
- hilfreich ist vor allem die `tabular`-Umgebung

Lebenslauf Vorlagen

Eine große Auswahl an Vorlagen für Lebensläufe findet sich auf Overleaf:

<https://www.overleaf.com/latex/templates/tagged/cv>

Wissenschaftliche Publikationen mit LaTeX

- je nach Fachbereich und Zeitschrift ist es gängig bis unerwünscht Publikationen in LaTeX einzureichen
- eine große Auswahl an Vorlagen findet sich auf Overleaf:
<https://www.overleaf.com/latex/templates/tagged/academic-journal>
- Auch Publikationen können natürlich frei gestaltet werden

Präsentationen mit LaTeX

Die beamer Klasse

```
\documentclass{beamer}  
\usetheme{<beamertheme>}
```

Folienaufbau

```
\frame{  
\frametitle{<Titel der Folie>  
... }
```

`\section{}` und `\subsection{}` zur Strukturierung (außerhalb der Folien)

“Animation” / zeitliche Abfolge

mittels `<>` oder `\pause`

Blöcke

```
\begin{block}{<Beschriftung>  
<Inhalt des Blocks>  
\end{block}
```

<code>block</code>	für Infos
<code>exampleblock</code>	für Beispiele
<code>alertblock</code>	für besonders hervorzuhebendes

Audio und Video Material

Das multimedia Paket

```
\movie[<attributes>]{<movie-label>}{<moviefile>}  
\sound[<attributes>]{<sound-label>}{<soundfile>}
```

Notizen:

<https://www.scivision.dev/beamer-latex-dual-display-pdf-notes/>