

Yet another \LaTeX -Tutorial...

...using \LaTeX

Tim Prüß

DHBW Ravensburg Campus Friedrichshafen

29. April 2022

- 1 Einführung
- 2 Basics
- 3 Struktur
- 4 Textgestaltung
- 5 Mathematische Ausdrücke
- 6 Bibliographie & Zitieren
- 7 Zusatz
- 8 Hilfsmittel

Einführung

Was ist L^AT_EX?

- T_EX = Satzsystem vom Informatik Professor *Donald Knuth*
- L^AT_EX = Weiterentwicklung durch *Leslie Lamport* (⇒ **Lamport TeX**)
- vereinfacht TeX u.a. durch die Benutzung von Makros
- Entwicklung dauert seit den 1990ern an, allerdings werden nur noch kleine Änderungen vorgenommen, weil L^AT_EX schon relativ fertig ist

Vorteile

- open-source & kostenlos
- professionelles Aussehen
- Automatisierung von Nummerierungen oder Verweisen
- Mathematischer Formelsatz
- Fokus auf Inhalt
- sehr performant, egal wie groß das Dokument ist

Nachteile

- ... die gewohnten Arbeitsabläufe funktionieren hier nicht mehr wie bei Word, weil \LaTeX kein WYSIWYG-Programm ist!
- Anfangs ist ein mühsames Umdenken nötig, aber je mehr man damit arbeitet, desto einfacher wird es
- \LaTeX lohnt sich (*aus meiner Erfahrung*) nur für längere Dokumente, wie wissenschaftliche Arbeiten, Bücher, ...
- Für kurze Dokumente kann Word genau so gut aussehen und vor allem zeiteffizienter sein.

Installation: Distribution

Zunächst benötigt man eine \LaTeX -Distribution

- = \LaTeX -Kernel und Sammlung an Paketen
- **MiKTeX** (Empfehlung), TexLive, ...
- MikTeX installiert u.a. fehlende Pakete automatisch, ist relativ einfach gehalten und betriebssystemunabhängig

Installation: Editor

Da die Distribution wie bei Linux als Konsole im Hintergrund arbeitet, benötigt man noch eine Schnittstelle zum Benutzer, die eine einfachere Bedienung ermöglicht.

- = Bearbeitungsprogramm
- **Texmaker** (Empfehlung), Texworks, ...
- Texmaker hat einen Darkmode und viele weitere nützliche Funktionen

Zur Installation von Texmaker und MiKTeX kann dieses [Videotutorial](#) befolgt werden.

Aktuelle Versionen: [MiKTeX](#), [Texmaker](#)

Alternative: Online-Editor

Der kostenlose Online-Dienst [Overleaf](#) bietet eine Cloud-basierte Bearbeitung von \LaTeX -Dokumenten mit (mehr oder weniger) Echtzeitverarbeitung.

Hierzu ist nur die Erstellung eines Kontos und keine Installation notwendig.

Für das spätere Arbeiten eignen sich beide Varianten.

Basics

Leerzeichen

Bei \LaTeX ist es völlig egal, wie viele Leerzeichen gesetzt werden. Alles wird wie ein Leerzeichen angesehen.

```
1 Bei \LaTeX{} ist es völlig egal,    wie viele  
2   Leerzeichen   gesetzt   werden.
```

Zeilen- & Seitenwechsel

Soll ein **Zeilenwechsel** stattfinden, muss das extra angegeben werden.

Das geht mit `\\`.

Leerzeilen führen zu einem neuen Absatz.

Neue **Absätze** können auch durch `\par` eingefügt werden.

```
1 Soll ein Zeilenwechsel
2 stattfinden, muss das extra angegeben werden. \\
3 Das geht mit \verb|\\|.
4
5 Leerzeilen führen zu einem neuen Absatz. \par\medskip
6 Neue Absätze können aber auch durch \lstinline|\par|
7 eingefügt werden.
```

Ein **Seitenwechsel** wird mit `\newpage` oder `\clearpage` vollzogen.

Bindestriche

Ein horizontaler Bindestrich kann vielfältig angewendet werden, da seine Länge variabel ist:

Binde-strich 6–16 Uhr –10 ja – oder nein ja — oder nein	Binde-strich 6--16 Uhr --10 ja -- oder nein ja --- oder nein	Für Wörter Für Uhrzeiten Für Zahlen, aber \$-10\$ ist besser
---	--	--

Reservierte Zeichen

Folgende Zeichen sind von \LaTeX reserviert: # \$ % ^ & _ { } ~ \

Wenn sie im Text benutzt werden wollen, muss folgende Schreibweise beachtet werden:

Zeichen	Bedeutung	Benutzung im Text
\	Start Befehl	<code>\textbackslash</code>
\\	Neue Zeile (<code>=\newline</code>)	
\$	Mathe-Modus	<code>\\$</code>
&	Tabulator	<code>\&</code>
%	Kommentar	<code>\%</code>
#	Raute	<code>\#</code>
~	Tilde	<code>\~{ }</code>
_	Tiefstellung	<code>_</code>
^	Hochstellung	<code>\^{} </code>
{ }	Argumente	<code>\{ \}</code>
[]	Optionen	<code>\$()\$</code>

Kommentare

Kommentare können mit dem %-Zeichen eingefügt werden.
Alles was danach kommt, wird von \LaTeX nicht beachtet.

Es gibt **keine** Blockkommentare.

```
1 % Dies ist ein Kommentar
2 %%%% Die Anzahl ist egal
3 kein Kommentar % Kommentar % immernoch Kommentar
```

Längenparameter

L^AT_EX hat einige interne Parameter, die das Aussehen des Dokuments bestimmen. Hier sind ein paar wichtige:

```
1 \parindent      % Einzug bei neuem Absatz
2 \parskip        % Abstand nach Absatz
3 \baselineskip    % Abstand zwischen Zeilen
4 \textwidth       % Breite des Textes auf einer Seite
5 \linewidth       % Länge einer Zeile in aktueller Umgebung
6 % ...
```

Die Parameter haben angepasste Standardwerte und müssen somit nicht verändert werden. Falls aber doch geht das so:

```
1 \setlength{\parindent}{0pt}    % Setzt Länge zu 0pt
2 \addtolength{\parindent}{10mm} % Addiert 10mm zu Länge
```


Längeneinheiten

Man kann je nach Situation unterschiedliche Einheiten verwenden:

<code>in</code>	inches bzw. Zoll
<code>mm</code>	Millimeter
<code>cm</code>	Centimeter
<code>pt</code>	points (ca. 1/72 inch)
<code>em</code>	Breite eines "M" in aktueller Schriftart
<code>ex</code>	Höhe eines "x" in aktueller Schriftart

Horizontale Abstände

Mithilfe von `\hspace{}` können Abstände eingefügt werden:

Dies ist ein Abstand von 2cm.

```
1 | Dies ist ein Abstand von \hspace{2cm} 2cm.
```

Mithilfe von `\hfill` wird leerer Platz gleichmäßig gefüllt:

links		rechts
links	mitte	rechts

```
1 | links \hfill rechts \\  
2 | links \hfill mitte \hfill rechts
```

Weitere Abstände:

<code>\,</code>	ca. 1 Leerzeichen
<code>\;</code>	ca. 1,5 Leerzeichen
<code>\quad</code>	ca. 2 Leerzeichen
<code>\qquad</code>	ca. 3 Leerzeichen

Vertikale Abstände

Diese werden mit `\vspace{}` erzeugt:

Das sind

2cm Abstand.

```
1 Das sind \\ \vspace{2cm}  
2 2cm Abstand.
```

Weitere Abstände:

<code>\smallskip</code>	ca. 1/4 Zeilenhöhe
<code>\medskip</code>	ca. 1/2 Zeilenhöhe
<code>\bigskip</code>	ca. 1 Zeilenhöhe
<code>\vfill</code>	analog zu <code>\hfill</code>

Umgebungen

Eine Umgebung (environment) muss immer geöffnet **und** geschlossen werden.

In die Umgebung kommt dann der Inhalt, der von ihr beeinflusst wird.

```
1 \begin{umgebung}[optionen]  
2   % dieser Text/Code wird von der Umgebung beeinflusst  
3 \end{umgebung}
```

Dokumente kompilieren

Nachdem man sein erstes Test-Dokument geschrieben hat, kann man das Dokument einfach über den Knopf "*PDFLaTeX*" kompilieren.

```
1 \documentclass{article}
2 \begin{document}
3   Hallo Welt!
4 \end{document}
```

Struktur

Grundgerüst

Jedes \LaTeX -Dokument muss mindestens aus folgenden Befehlen bestehen:

```
1 \documentclass{article}
2   % Präambel: Konfiguration des Dokuments
3 \begin{document}
4   Hallo Welt! % Dokumentinhalt
5 \end{document}
```

Befehle sind dabei immer so aufgebaut:

```
1 \befehl[option1, opt2, ...]{argument1}{arg2}{...}
```

Präambel

- Präambel ist ähnlich einer Header-Datei bei C
- befindet sich zwischen `\documentclass[]{}` und `\begin{document}`
- beinhaltet alle Pakete, die benötigt werden und konfiguriert das gesamte Dokument
- für die T1000/2000/... existiert eine gute Vorlage

Abschnitte

Abschnitte strukturieren das Dokument und werden automatisch ins Inhaltsverzeichnis eingefügt. Es gibt verschiedene Tiefen von Abschnitten in folgender Reihenfolge (höchste zu kleinste):

```
1 %\chapter{}
2 %\section[Titel im Inhaltsverzeichnis]{Titel}
3 %\subsection{Titel}
4 %\subsubsection{Titel}
5 %
6 %\abschnitt*{} => erscheint nicht im Inhaltsverzeichnis
```

Es gibt noch weitere Abschnitte, aber diese werden sehr selten benutzt und deshalb hier nicht behandelt.

Verzeichnisse

In \LaTeX werden alle Verzeichnisse automatisch beim Kompilieren erstellt.

```
1 \maketitle           % oder \titlepage erzeugt Titelseite
2 \tableofcontents     % Hier wird das Inhaltsverzeichnis
   erstellt
3 \listoffigures       % Hier wird das Abbildungsverz. erstellt
4 \listoftables        % Hier wird das Tabellenverz. erstellt
```

Einfügen von Grafiken

L^AT_EX unterstützt von Haus aus nur .eps-Dateien.

Mit dem Paket `\usepackage{graphicx}` können aber auch .jpg-, .png- und .pdf-Dateien eingebunden werden.

```
1 \includegraphics[width=\textwidth]{images/Bild.png}  
2 % oder  
3 \includegraphics[scale=0.5]{images/Bild.png}
```

Bei der Suche nach Bildern für eine Arbeit sollten möglichst hochauflösende jpg- oder wenn möglich sogar pdf-Dateien verwendet werden, sonst leidet das Aussehen.

Abbildungen

Damit noch eine Bildunterschrift eingefügt und später im Text referenziert werden kann, muss die `figure`-Umgebung benutzt werden:

```

1 \begin{figure}[platzierung]
2   \centering % Zentriert die
   Abbildung
3   \includegraphics{images/
   harold_stock.jpg}
4   \caption{Bildunterschrift}
5   \label{Name} % Referenzmarke
6 \end{figure}

```



Abbildung 1: Dies ist ein Bild

Für die `platzierung` gibt es folgende Angaben: (`float`-Paket nötig!)

<code>h</code>	Platzierung ungefähr an der Stelle im Code (<i>here</i>)
<code>t</code>	Platzierung oben auf der Seite (<i>top</i>)
<code>b</code>	Platzierung auf extra Seite
<code>!</code>	überschreibt alle internen Parameter
<code>H</code>	exakte Platzierung an der Stelle im Code (<i>HERE!</i>)

Referenzierung

Referenzmarken können mit `\label{markenname}` erstellt werden. Wo der Marker im Code steht ist egal, er muss nur in der Nähe von dem Inhalt stehen, der referenziert werden soll.

Um später die Marke wiederzufinden, sollte der Name gut gewählt werden. Folgende Nomenklatur wird häufig verwendet:

```
1 \label{cha:name} % Für Chapters
2 \label{sec:name} % Für Sections
3 \label{sub:name} % Für Subsections
4 \label{fig:name} % Für Abbildungen
5 \label{eq:name}  % Für Gleichungen ...
```

Die spätere Referenzierung im Text sieht dann z.B. so aus:

```
\ref{cha:name}
```

Tabellen

Diese können mit der `tabular`-Umgebung erzeugt werden. Texmaker bietet allerdings auch einen Tabellen-Assistenten an, der Tabellen erzeugen kann.

```

1 \begin{tabular}{spalten}
2 \hline % horizontale Linie
3 Spalte1/Zeile1 & Spalte2/Zeile1 & ... \\
4 \hline
5 Spalte1/Zeile2 & Spalte2/Zeile2 & ... \\
6 ...
7 \end{tabular}

```

Die Spaltenangabe besteht aus folgenden Angaben:

l	linksbündige Spalte
c	zentrierte Spalte
p{Breite}	für Textabsätze
	senkrechte Linie
	doppelte Linie

Folgende Befehle können in der `tabular`-Umgebung angewendet werden:

<code>&</code>	trennt Spalten
<code>\\</code>	neue Zeile
<code>\hline</code>	horizontale Linie
<code>\newline</code>	neue Zeile in p-Spalte

```

1 \begin{tabular}{rlc}
2   1 & 2 & 3 \\ \hline
3   4 & 5 & 6 \\
4   7 & 8 & 9 \\
5 \end{tabular}

```

1	2	3
4	5	6
7	8	9

```

1 \begin{tabular}{l|c|r||}
2   \hline
3   1 & 2 & 3 \\ \hline
4   4 & 5 & 6 \\ \hline
5   7 & 8 & 9 \\ \hline \hline
6 \end{tabular}

```

1	2	3
4	5	6
7	8	9

Ähnlich wie mit der `figure`-Umgebung kann auch eine Tabelle mit der `table`-Umgebung dargestellt werden:

```
1 \begin{table}
2   \centering
3   \begin{tabular}{c|c|c}
4     1 & 2 & 3 \\ \hline
5     4 & 5 & 6 \\ \hline
6     7 & 8 & 9 \\ \hline
7   \end{tabular}
8   \caption{Eine Tabelle}
9 \end{table}
```

1	2	3
4	5	6
7	8	9

Tabelle 1: Eine Tabelle

Tabellen und Abbildungen haben dabei ihre eigenen Zähler.

Listen

Es gibt drei verschiedene Listenumgebungen:

- `itemize`
- `enumerate`
- `description`

```
1 \begin{itemize}
2   \item erster Punkt
3   \begin{itemize}
4     \item Unterpunkt
5   \end{itemize}
6   \item zweiter Punkt
7 \end{itemize}
```

```
1 \begin{enumerate}
2   \item erster Punkt
3   \begin{enumerate}
4     \item Unterpunkt
5   \end{enumerate}
6   \item zweiter Punkt
7 \end{enumerate}
```

```
1 \begin{description}
2   \item[Erstens:] ...
3   \item[Zweitens:] ...
4 \end{description}
```

- erster Punkt
 - Unterpunkt
- zweiter Punkt

- 1 erster Punkt
 - 1 Unterpunkt
- 2 zweiter Punkt

Erstens: ...
Zweitens: ...

Textgestaltung

Textgrößen

Durch verschiedene Größen, können z.B. Überschriften vom restlichen Text getrennt werden. Die Größen werden fast immer automatisch durch \LaTeX festgelegt.

<code>\tiny</code>	Mikroschrift
<code>\scriptsize</code>	Tiefstellung
<code>\footnotesize</code>	Fußnoten
<code>\small</code>	klein
<code>\normalsize</code>	normal
<code>\large</code>	groß
<code>\Large</code>	größer
<code>\LARGE</code>	sehr groß
<code>\huge</code>	riesig
<code>\Huge</code>	sehr riesig

Texthervorhebung

`\textit{}` *kursiv* (it=italic)

`\textsl{}` *angewinkelt*

`\emph{}` *betont* (da sie nicht wirklich unterscheidbar sind, würde ich nur `\textit{}` verwenden)

`\textbf{}` **fett**

`\textsc{}` KAPITÄLCHEN (sc=small caps)

`\textrm{}` serifenschrift (rm=roman)

`\textsf{}` serifenlos (sf=serif)

`\texttt{}` schreibmaschine

Textausrichtung

Text kann folgendermaßen **zentriert** werden:

```
1 {\centering  
2 Dies \\  
3 ist \\  
4 zentriert.\\}
```

Dies
ist
zentriert.

```
1 \begin{center}  
2 Dies \\  
3 ist \\  
4 zentriert.  
5 \end{center}
```

Dies
ist
zentriert

Für **linksbündigen** Text kann `\begin{flushleft} ... \end{flushleft}`,
für **rechtsbündigen** Text `\begin{flushright} ... \end{flushright}` bzw.
`{\raggedright}` verwendet werden.

Mathematische Ausdrücke

Mathematische Ausdrücke

Um diese zu benutzen ist das `amsmath`-Paket nützlich.

\LaTeX unterscheidet zwei Arten der Formeleingabe:

- Inline: innerhalb einer Zeile im Text
- Display: als eigene Gleichung vom Text getrennt

Inline:

```
1 | Hier $x^2 - 1 = 0$ im Text
```

Hier $x^2 - 1 = 0$ im Text

Display:

```
1 | $$x^2 - 1 = 0$$
```

$$x^2 - 1 = 0$$

equation-Umgebung

Mittels der equation-Umgebung ist displaymath auch wie in der figure-Umgebung möglich:

```
1 \begin{equation}
2 a^2 + b^2 = c^2
3   \label{eq:pythagoras}
4 \end{equation}
5 siehe Gleichung
6 \eqref{eq:pythagoras}
```

$$a^2 + b^2 = c^2 \quad (1)$$

siehe Gleichung (1)

Ist die Nummerierung nicht gewünscht, kann

`\begin{equation*} ... \end{equation*}`
verwendet werden.

align-Umgebung

Die `align`-Umgebung sorgt dafür, dass mehrere Gleichungen übereinander an einem Zeichen angeordnet werden können. Dazu wird das `&`-Zeichen verwendet.

```
1 \begin{align}  
2 a^2 + b^2 &= c^2 \\  
3 a^2 &= c^2 - b^2  
4 \end{align}
```

$$a^2 + b^2 = c^2 \quad (2)$$

$$a^2 = c^2 - b^2 \quad (3)$$

Bei nicht gewünschter Nummerierung gilt wieder:

```
\begin{align*} ... \end{align*}
```

Symbole

Es gibt *sehr* viele Symbole in \LaTeX .

Folgende Symbole können direkt eingegeben werden:

+ - = ! / () [] < > | ' :

Es ist nützlich ein *Cheat-Sheet* zu benutzen oder die gewünschten Befehle einfach zu googeln. Im [Wikibooks](#) (oder [Wikipedia](#)) ist eine Liste an Symbolen zu finden.

Über [Detexify](#) kann ein bestimmtes Symbol auch über Zeichnen gefunden werden.

Hoch- & Tiefstellung

Ausdrücke können mit \wedge hoch- und mit $_$ tiefgestellt werden.
Wenn Ausdrücke aus mehr als einem Zeichen bestehen muss dieser mit $\{ \dots \}$ eingeklammert werden.

$$1 \mid a_{n+1} = a_n^2 + (n-1)^2$$

$$a_{n+1} = a_n^2 + (n-1)^2$$

Brüche

Brüche können sowohl mit `\frac{zähler}{nenner}` (entscheidet abhängig von Umgebung, wie Bruch dargestellt werden soll) und mit `\dfrac{zähler}{nenner}` (Bruch im displaystyle).

Wenn kein Platz vorhanden ist, kann ein Bruch auch mit `zähler/nenner` erzeugt werden.

```
1 \frac{n!}{k!(n-k)!} = \binom{n}{k}
```

$$\frac{n!}{k!(n-k)!} = \binom{n}{k}$$

```
1 \frac{\frac{1}{x} + \frac{1}{y}}{y-z}
```

$$\frac{\frac{1}{x} + \frac{1}{y}}{y-z}$$

```
1 1 + \dfrac{1}{1 + \dfrac{1}{1 + \dfrac{1}{1 + \dots}}}
```

$$1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1 + \dots}}}$$

Summen & Integrale

Hier unterscheiden sich wieder die zwei Arten der Formeleingabe:

```
1 % Inline Summe im Text
2 \sum_{n=1}^{\infty} \frac{1}{n^2}
```

$$\sum_{n=1}^{\infty} \frac{1}{n^2}$$

```
1 % Display-Stil in Umgebung
2 \sum_{n=1}^{\infty} \frac{1}{n^2}
```

$$\sum_{n=1}^{\infty} \frac{1}{n^2}$$

```
1 % Inline Integral im Text
2 \int_{-\infty}^{\infty} x^2 \mathrm{d}x
```

$$\int_{-\infty}^{\infty} x^2 dx$$

```
1 % Display-Stil in Umgebung
2 \int_{-\infty}^{\infty} x^2 \mathrm{d}x
```

$$\int_{-\infty}^{\infty} x^2 dx$$

Auf weitere Befehle kann nicht weiter eingegangen werden. Die meisten haben aber eine einfache Syntax und können leicht über die erwähnten Nachschlagewerke gefunden werden.

Korrekte Klammersetzung

Die Größe der Klammern wird von \LaTeX automatisch angepasst, allerdings muss folgendes beachtet werden:

1 | `(\frac{x^2}{y^3})` % Falsch!

$$\left(\frac{x^2}{y^3}\right)$$

1 | `\left(\frac{x^2}{y^3} \right)` % Korrekt

$$\left(\frac{x^2}{y^3}\right)$$

1 | `\left\{ \frac{x^2}{y^3} \right\}` % Korrekt

$$\left\{\frac{x^2}{y^3}\right\}$$

1 | `\left. \frac{x^2}{2} \right|_0^1` % Korrekt

$$\left.\frac{x^2}{2}\right|_0^1$$

Bibliographie & Zitieren

Bibliographie

L^AT_EX kann Literaturverzeichnisse mit einem Tool namens **BibTeX** organisieren und erstellen.

Die Quelleneinträge werden über eine Art "Datenbank" verwaltet und können anschließend im Dokument referenziert werden.

Die Literatureinträge werden in der Präambel eingebunden:

```
1 \usepackage[backend=biber]{biblatex}  
2 \addbibresource{quellen.bib}
```

Das Literaturverzeichnis wird dann durch den Befehl `\printbibliography` an der Stelle im Code erzeugt.

Zitieren

Ein Literatureintrag hat immer einen *Typ* (z.B.: Buch), ein *Kürzel* für den schnellen internen Aufruf in \LaTeX und diverse *Angaben* zur Quelle.

```
1 % Beispieelinhalt einer .bib-Datei
2 @book{kürzel, % Für den Aufruf in LaTeX
3   title={LaTeX-Tutorium},
4   author={P., Tim},
5   year={2022}
6   % mögliche weitere Angaben
7 }
```

Soll die Quelle jetzt referenziert werden, verwendet man einfach den Befehl `\cite{kürzel}` an der Stelle im Text.

Die Einträge können über Literatur-Tools, Online oder auch manuell erstellt werden.

Zusatz

SI-Einheiten

Mithilfe des `siunitx`-Pakets können SI-Einheiten und Zahlen im korrekten Format eingegeben werden. Die Benutzung ist [hier](#) weiter erklärt.

```
1 \num{123,45} % Zahlen
```

```
2  
3 \num{123e45}
```

```
4  
5 \si{\volt\second\per\ampere} % nur  
   Einheit
```

```
6  
7 \SI{100}{\milli\volt} % Einheit mit Zahl
```

```
8  
9 \SIlist{1;3;8}{\newton} % Aufzählung
```

```
10  
11 \SIrange{1}{10}{\newton} % Bereich
```

123,45

$123 \cdot 10^{45}$

$\frac{\text{Vs}}{\text{A}}$

100 mV

1 N, 3 N und 8 N

1 N bis 10 N

Abkürzungen

Mit dem `glossaries`-Paket können Glossare, Indexverzeichnisse und Abkürzungsverzeichnisse erzeugt werden. Eine neue Abkürzung kann mit `\newacronym{kürzel}{kurzform}{langform}` erzeugt werden.

Es ist gut, die Einträge in eine extra Datei auszulagern (z.B. `abkuerzungen.tex`)

Um eine Abkürzung zu benutzen, nimmt man den Befehl `\gls{kürzel}`. Wenn die Abkürzung das erste mal verwendet wird, wird automatisch sowohl die Kurz- als auch die Langform ausgegeben.

Das Abkürzungsverzeichnis wird mit `\printglossaries` erzeugt.

Fußnoten

Folgt noch

Verbatim

Manchmal muss man in \LaTeX Zeichen eingeben, die aber bereits reserviert sind. Für einzelne Zeichen ist dies wie hier gezeigt möglich.

Das wird allerdings schnell nervig, wenn z.B. Quellcode dargestellt werden soll.

Eine Möglichkeit Text "wörtlich" (engl.: *verbatim*) darzustellen, ist die *verbatim*-Umgebung: `\begin{verbatim} ... \end{verbatim}`

oder die Inline-Variante: `\verb# #`

Bei der Inline Variante ist zu beachten, dass das Anfangs- & Endzeichen gleich sein müssen und nicht im Text enthalten sein dürfen.

Das *verbatim*-Paket muss hierzu eingebunden werden!

Quellcode

Bei `verbatim` wird die Syntax des Quellcodes nicht hervorgehoben. Dafür gibt es jedoch das `listings`-Paket.

```
\begin{lstlisting} ... \end{lstlisting}
```

oder die Inline-Variante: `\lstinline# #`

Die gleichen Regeln wie bei `verbatim` gelten auch hier.

Die Sprache kann auch global in der Präambel (`\lstset{language=...}`) festgelegt werden.

Hilfsmittel

Wo finde ich Hilfe?

- Die Kurzeinführung [Docu-l2kurz-german](#) ist ca. 60 Seiten groß und umfasst alle Basics
- Deutsches \LaTeX -Wiki: [GoLaTeX](#)
- Mathematische Symbole: [Wikibooks](#)
- Google ist dein Freund: Wenn Probleme auftreten, einfach die Fehlermeldung in eine Suchmaschine schmeißen...
- Falls nichts funktioniert, könnt ihr mich unter `pruess.tim-it20@it.dhbw-ravensburg.de` erreichen