

Mini-L^AT_EX-Doku_(work in progress)

Max Schmidt
to.maxschmidt@pm.me

Florian Sagerer
f.sagerer@protonmail.com

7. Juni 2025

Inhaltsverzeichnis

0	Wozu \LaTeX und Installation	3
0.1	Aber ich schreib meine Arbeit gerne in Word!	3
0.2	Ok, ichnehm's ja schon her...	3
0.3	Installation	3
0.3.1	Windows	4
0.3.2	Linux	4
0.3.3	MacOS	4
1	<i>Los geht's!</i> - Ein Dokument erstellen	4
2	<i>Worum geht's?</i> - Titelseite und Inhaltsverzeichnis	5
3	<i>Fett, klein, schief,...</i> - Text Formatierung	6
4	<i>Recht und Ordnung!</i> - Sections	6
5	<i>Da, wo der Daumen rechts ist</i> - Links	7
5.1	weblink	7
5.2	Hyper-Referenz	7
5.3	Nummer-Referenz	7
6	<i>Soße?</i> - Quellenverzeichnis	8
7	<i>Eins nach dem Anderen.</i> - Tabellen, Listen	8
7.1	Listen	8
7.1.1	Bulletlist	8
7.1.2	Nummerierte Listen	9
7.1.3	Listen in Listen	9
7.2	Tabellen	9
8	<i>Was zum Anschauen!</i> - Bilder	10
9	<i>Mathezauber</i> - Was man im Infostudium so an Sonderzeichen braucht	10
9.1	Grundlagen	11
9.2	Mehrzeilige Gleichungen	11
9.3	Megen und Logische Formeln	12
9.4	Graphen	12
10	<i>Programmieren.</i> - Code Snippets	12
10.1	verbatim	13
10.2	listings	13
10.3	Algorithmen	14
11	<i>Ich mal mir die Welt, wie sie mir gefällt...</i> - Farben	14

**Lass dich nicht abschrecken! Lies gerne einfach nur das, was du willst/brauchst.
Es werden Bezüge auf Abschnitte genommen, wo es notwendig/sinnvoll sind.**

0 Wozu \LaTeX ? und Installation

\LaTeX ist der de facto Standard für Wissenschaftliche Arbeit und jeder, der für sowas was anderes (wie zum Beispiel Word) benutzt, tut seinen potentiellen Lesern und vorallem sich selbst absolut keinen Gefallen. Der Anfang ist nicht so einfach, aber ist einmal das Dokument ge-set-up-t, dann ist alles viel einfacher.

Dieses Dokument ist auch in \LaTeX geschrieben und auf floriansagerer.de/mini-LATEX-doku abrufbar. Es wird fortlaufend geändert. Der Quelltext ist auf [GitHub](https://github.com) abrufbar.

Dieses Dokument soll einen einfachen Einstieg und eine kurze Übersicht geben. Eine ausführlichere Dokumentation gibt es auf [Overleaf \(overleaf.com/learn\)](http://overleaf.com/learn) und [Wikibooks \(en.wikibooks.org/wiki/LaTeX\)](http://en.wikibooks.org/wiki/LaTeX). Ansonsten wird man eigentlich immer fündig, wenn man in einer Suchmaschine »*latex wie mach ich das und das*« eingibt.

0.1 Aber ich schreib meine Arbeit gerne in Word!

Folgendes geht in \LaTeX quasi von selbst oder sehr einfach und in Word nur mit sehr, sehr viel Arbeit oder gar nicht:

- Überschriften werden immer richtig nummeriert (stell dir vor du musst alles ändern, wenn du nur eine Überschriftfügst)
- Automatische Seitennummerierung (auch wenn sie erst auf Seite 2 oder so starten soll)
- Automatisches Inhaltsverzeichnis mit richtiger Nummerierung und Seitenzahl
- Referenzen auf Sektionen/Subsektionen/Abbildungen/... aktualisieren sich von selbst
- Abbildungen einfügen ohne dass Word den gesamten Text über drei Seiten durcheinander schiebt, dein Haus abfackelt, dein Hund kidnapt und dein Auto mit einer Kartoffel zerkratzt
- ein gut aussehendes Quellenverzeichnis
- ein gutes, simples Titelblatt
- sehr viel mehr

0.2 Ok, ich nehme's ja schon her...

Ok, super! Zum mal Ausprobieren ist [Overleaf \(overleaf.com\)](http://overleaf.com) gut. Einfach dort anmelden und anfangen. Es gibt dort auch [Vorlagen \(overleaf.com/latex/templates\)](http://overleaf.com/latex/templates), die man einfach öffnen und sofort benutzen kann.

Wenn du vollkommen offline arbeiten möchtest, kannst du \LaTeX auch auf deinem PC installieren (siehe Unterabschnitt 0.3).

0.3 Installation

Hier installieren wir \LaTeX auf unserem PC. Alternativ einfach Overleaf nutzen (siehe Unterabschnitt 0.2).

0.3.1 Windows

Falls diese Anleitung nicht hilft, habe ich ein [Youtube Tutorial \(youtube.com/watch?v=NnqxgMVeMiw\)](https://www.youtube.com/watch?v=NnqxgMVeMiw) erstellt.

Die drei Sachen herunterladen und installieren:

- [Visual Studio Code \(code.visualstudio.com\)](https://code.visualstudio.com)
- [MiKTeX \(miktex.org/download\)](https://miktex.org/download)
- [Strawberry Perl \(strawberryperl.com\)](https://strawberryperl.com)

In VS-Code die Extension [LaTeX Workshop](#) installieren. MikTeX Console öffnen (wenn es nicht geht, dann im Icon Tray rechts in der Taskbar schließen und wieder öffnen). In der MikTeX Console unter *Updates* → *Check for Updates* und dann *Update now*. Fertig.

0.3.2 Linux

- [TeX Live \(tug.org/texlive\)](https://tug.org/texlive)
- [Visual Studio Code \(code.visualstudio.com\)](https://code.visualstudio.com)
- [LaTeX Workshop VSCode Extension](#)
- Alternativ: um in Vim zu kompilieren, siehe [meine vimrc](#)

0.3.3 MacOS

Diese beiden Apps / Packages herunterladen und installieren:

- [Visual Studio Code \(code.visualstudio.com\)](https://code.visualstudio.com)
- [MacTeX \(https://www.tug.org/mactex/\)](https://www.tug.org/mactex/)

Am einfachsten geht das mit [brew \(brew.sh\)](#):

```
brew install --cask mactex
brew install --cask visual-studio-code
```

Ich musste noch von Hand den TeX Compiler zum PATH hinzufügen (wenn du Zsh nutzt, geht das so):

```
echo 'export PATH="/Library/TeX/texbin:$PATH"' >> ~/.zshrc
```

Dann die TeX Live Utility App öffnen und die vorgeschlagenen Updates für die TeX-Packages installieren. Zu guter Letzt in VS Code die Extension [LaTeX Workshop](#) installieren.

Sonderzeichen Weil Apple sich zu fein ist, alle Zeichen auf die Tastatur zu drucken hier eine Zusammenfassung der Zeichen die man für \LaTeX braucht:

Zeichen	Tastenkombination auf Mac
\	Option + Shift + 7
[/]	Option + 5 / 6
{ / }	Option + 8 / 9

1 *Los geht's!* Ein Dokument erstellen

Sobald du in **Overleaf** angemeldet bist, kannst du ein Projekt öffnen (z.B ein Template oder eine hochgeladene .zip) oder ein neues Projekt anlegen. Wenn das Projekt geöffnet ist, siehst du ganz Links eine **Übersicht**

über die **Dateien** in dem Projekt, rechts davon den **Editor** mit der geöffneten Datei und ganz rechts den **PDF-Viewer**. Es sollte mindestens eine Datei mit der Endung `.tex` existieren. Das ist die Hauptdatei mit dem \LaTeX Code.

Um **lokal auf deinem PC** ein Dokument zu erstellen (Unterabschnitt 0.3 notwendig), einfach einen neuen Ordner erstellen und dann in **VS-Code** eine neue Datei mit der Endung `.tex` erstellen.

Folgender \LaTeX -Code ist eigentlich immer ein guter Start:

```
% mit % kann man ueberall Kommentare hinzufuegen, die nicht im PDF auftauchen
\documentclass{article}
\usepackage[ngerman]{babel} %damit von LaTeX erstellte Texte auf Deutsch sind
\usepackage[left=2.5cm, right=2.5cm, top=2.5cm, bottom=2cm]{geometry} %Seitenraender
\usepackage[onehalfspacing]{setspace} %Zeilenabstaende

\begin{document}
Hier koennte Ihre Werbung stehen!
\end{document}
```

In **Overleaf** kann mit dem **grünen (Re-)Compile** oder auch mit **Strg + S** das PDF erstellt bzw. aktualisiert werden.

In **VS-Code** kann mit dem **grünen Pfeil** oben rechts das Dokument das erste Mal kompiliert (d.h. das PDF erstellt) werden; ab dann kann immer mit **Strg + S** gespeichert und kompiliert werden.

Tipp: Man kann in VS-Code Fenster per Drag-and-Drop nebeneinander schieben. Dann kannst du links tippen und rechts gucken.

2 *Worum geht's?* Titelseite und Inhaltsverzeichnis

Damit \LaTeX weiß, was auf der Titelseite stehen soll, muss das zuerst definiert werden: Es kann alles in die Klammern geschrieben werden.

```
\title{Titel des Dokuments}
\author{Max Schmidt}
\date{Juli 23} % weglassen um automatisch das heutige Datum zu verwenden
```

Um die Titelseite dann zu erstellen:

```
\maketitle
\thispagestyle{empty} %wenn auf der Titelseite keine Seitenzahl stehen soll
\newpage %fuer einen Seitenumbruch
```

Um das Inhaltsverzeichnis zu erstellen:

```
\tableofcontents
\thispagestyle{empty} %wenn auf der Seite keine Seitenzahl stehen soll
\newpage %fuer einen Seitenumbruch
```

Um Seitenzahlen von hand zu Modifizieren, kann man einfach

```
\setcounter{page}{x}
```

auf der Seite einfügen auf der die Seitenzahl x sein soll, alle weiteren Seiten werden dann ganz normal weiter hochgezählt.

3 *Fett, klein, schief,...* Text Formatierung

Formatierung	Beispielcode
Fett	<code>\textbf{Fetter Text}</code>
<i>Kursiv</i>	<code>\textit{Kursiver Text}</code>
<u>Unterstrichen</u>	<code>\underline{Unterstrichener Text}</code>
Durchgestrichen	<code>\sout{Durchgestrichener Text}</code>
s c h r i f t g r ö ß e	<code>\tiny, \scriptsize, \footnotesize, \small, \normalsize, \large, \Large, \huge, \Huge</code>
Schriftart	<code>\textsf{Serifenlose Schrift}</code> <code>\texttt{Monospaced Schrift}</code>
Schriftfarbe	<code>\textcolor{Farbe}{Text}</code>
Hoch- / Tiefstellung	<code>Hochgestellter Text</code> <code>\textsubscript{Tiefgestellter Text}</code>
Zeilen- Umbruch	<code>\\</code> oder <code>\newline</code>

L^AT_EX ist kein „What you see is what you get“ also hilft es dir nix im Code Zeilenumbrüche zu machen.

← So eine Einrückung nach einem Zeilenumbruch kann mit `\noindent` entfernt werden.

4 *Recht und Ordnung!* Sections

Sections sind die einzelnen Abschnitte des Dokuments, sie werden automatisch nummeriert und im Inhaltsverzeichnis aufgeführt. Einen neuen Abschnitt erstellt man so:

```
\section{Name des Abschnitts}\label{kurzer-name}
```

Das Label ist da, um den jeweiligen Abschnitt an anderer Stelle zu referenzieren (siehe Unterabschnitt 5.2 und Unterabschnitt 5.3).

Will man einen **Unterabschnitt** erstellen, schreibt man ein `sub` davor. Das kann man beliebig oft machen:

```
\section{Abschnitt}\label{Abschnitt}
  \subsection{Unterabschnitt}\label{Unterabschnitt}
    \subsubsection{Unterunterabschnitt}\label{Unterunterabschnitt}
  \subsection{Zweites Unterabschnitt von Abschnitt}\label{zweites-Unterabschnitt}
```

Ein (Unter-)Abschnitt mit **Sternchen** wird nicht vom Inhaltsverzeichnis und der Nummerierung beachtet:

```
\subsection*{Geheim}
```

5 *Da, wo der Daumen rechts ist* Links

Damit wir Links benutzen können und auch die Abschnitte im Inhaltsverzeichnis anklickbar sind, brauchen wir das Package *hyperref*:

```
\usepackage{hyperref}
```

Wir können das Aussehen von Links anpassen:

```
\hypersetup{
  colorlinks,
  citecolor=blue,
  filecolor=blue,
  linkcolor=black,
  urlcolor=hellblau,
  pdftitle={Mini-LaTeX-Doku}
}
```

5.1 Website verlinken

Code:

```
Die Dokumentation ist auf \href{https://floriansagerer.de/}{meiner Website} aufrufbar.
```

Aussehen:

Die Dokumentation ist auf [meiner Website](https://floriansagerer.de/) aufrufbar.

5.2 Hyper-Referenz

Der Abschnitt muss hierfür ein Label haben (siehe Abschnitt 4).

Code:

```
Abschnitte im \hyperref[Titelseite und Inhaltsverzeichnis]{Inhaltsverzeichnis}
```

Aussehen:

Abschnitte im Inhaltsverzeichnis

5.3 Nummer-Referenz

Der Abschnitt muss hierfür ein Label haben (siehe Abschnitt 4).

Code:

```
Siehe auch \autoref{Ein Dokument erstellen}.
```

Aussehen:

Siehe auch Abschnitt 1.

6 *Soße?* Quellenverzeichnis

Siehe auch [dieses Video \(youtube.com/watch?v=hqaLeq9huqw\)](https://youtube.com/watch?v=hqaLeq9huqw).

Wir brauchen eine Quelldatei mit der Endung *.bib*, zum Beispiel *quellen.bib*, in der wir die Quellen im *BibTeX*-Format abspeichern:

```
@book{marx1867kapital,  
  title={Das Kapital},  
  author={Marx, Karl},  
  year={1867},  
  publisher={Otto Meissner}  
}
```

Mit der Quellenbezeichnung in der ersten Zeile, zitieren wir später die Quelle.

In unserem *L^AT_EX*-Dokument schreiben wir:

```
\usepackage[]{biblatex}  
\addbibresource{quellen.bib}
```

Um im Text zu zitieren, schreiben wir:

```
\cite[]{marx1867kapital} % in der eckigen Klammer koennen wir noch extra Informationen  
  , wie Seitenzahl angeben
```

Um das Quellenverzeichnis anzugeben, schreiben wir:

```
\printbibliography
```

7 *Eins nach dem Anderen.* Tabellen, Listen

7.1 Listen

Siehe auch: [Overleaf Dokumentation zu Listen \(overleaf.com/learn/latex/Lists\)](https://overleaf.com/learn/latex/Lists).

7.1.1 Bulletlist

So macht man eine einfache Liste aus Bulltpoints:

```
\usepackage{enumitem}  
\setlist{nolistsep} %das reicht einmal im ganzen Dokument und dient nur dazu Abstand  
  vor einer liste kleiner zu machen  
\begin{itemize}[noitemsep] %das [noitemsep] ist optional und dient nur dazu den  
  Abstand zwischen den Bulltpoints huebscher zu machen  
  \item erster Punkt  
  \item zweiter Punkt  
  \item dritter Punkt  
  \item ja so geht's weiter
```



```
\end{itemize}
```

So sieht die dann aus:

- erster Punkt
- zweiter Punkt
- dritter Punkt
- ja so geht's weiter

7.1.2 Nummerierte Listen

Code:

```
\usepackage{enumitem}
\setlist{nolistsep}
\begin{enumerate}[noitemsep]
  \item huhu
  \item das ist
  \item eine nummerierte Liste
\end{enumerate}
```

Aussehen:

1. huhu
2. das ist
3. eine nummerierte Liste

7.1.3 Listen in Listen

Code:

```
\usepackage{enumitem}
\setlist{nolistsep}
\begin{enumerate}[noitemsep]
  \item huhu
  \item das ist
  \begin{itemize}
    \item ist eine
    \item liste in der liste
  \end{itemize}
\end{enumerate}
```

Aussehen:

1. huhu
2. das ist
 - ist eine
 - liste in der liste

7.2 Tabellen

Code:

```
\begin{tabular}{l|c|r}
  links & mitte & rechts \\
\hline
  1 & 2 & 3 \\
  4 & 5 & 6 \\
\end{tabular}
```

Aussehen:

links	mitte	rechts
1	2	3
4	5	6

Zeile	Erklärung
1	in den zweiten Klammer werden die Spalten definiert, jede Spalten wird durch einen Buchstaben repräsentiert, l (linksbündig), r (rechtsbündig) c (mittig)

8 *Was zum Anschauen!* Bilder

Für Bilder brauchen wir das Package *graphicx* und einen Ordner:

```
\usepackage{graphicx}
\graphicspath{ {./images/} }
```

Ein Bild, das in unserem Ordner unter *images/bild.png* gespeichert ist können wir so darstellen und skalieren:

```
\includegraphics[scale=1.5]{bild}
```

Wollen eine eher ordentlicher Darstellung mit Bildunterschrift, können wir eine *figure* benutzen:

```
\begin{figure}
  \includegraphics[scale=1.5]{bild}
  \centering %zentriert die Darstellung mittig
  \caption{Eine passende Bildunterschrift zu unserem Bild}
  \label{figure-passenderName} %ein label mit einem einzigartigem Namen um die
                             %Abbildung spaeter Referenzieren zu koennen
\end{figure}
```

Später im Text kann man dann so auf die Abbildungsnummer zugreifen:

```
siehe Abbildung \ref{figure-passenderName}
```

Mit dem Package *float* und der Option *H* kann man sicher stellen, dass die Darstellung genau an der Stelle im Dokument erscheint, in der sie auch im Code ist: dies kann aber sehr schnell zu Formatierungsproblemen führen.

```
\usepackage{float}

\begin{figure}[H]
  \includegraphics[scale=1.5]{bild}
\end{figure}
```

9 *Mathezauber* Was man im Infostudium so an Sonderzeichen braucht

Siehe auch: [Wikibooks L^AT_EX Guide \(en.wikibooks.org/wiki/LaTeX/Mathematics\)](http://en.wikibooks.org/wiki/LaTeX/Mathematics).

Um in L^AT_EX eine mathematische Formel zu schreiben, muss man dem Kompilierer zuerst mitteilen, dass es jetzt mathematisch wird:

- Man kann die Formel entweder **inline**: $2 + 2 = 3$
- oder als **eigenen Absatz** formatieren lassen: $a \in A$

Inline sieht eine Formel dann so $2 + 2 = 3$ aus und als eigener Absatz

$$a \in A$$

so aus.

9.1 Grundlagen

Name	Aussehen	Code
Addition	$1 + 3$	<code>\$ 1 + 3 \$</code>
Subtraktion	$1 - 3$	<code>\$ 1 - 3 \$</code>
Multiplikation	$1 * 3$	<code>\$ 1 * 3 \$</code>
	$1 \cdot 3$	<code>\$ 1 \cdot 3 \$</code>
	1×3	<code>\$ 1 \times 3 \$</code>
Division	$1/3$	<code>\$ 1 / 3 \$</code>
	$1/3$	<code>\$ ^1/_3 \$</code>
	$1:3$	<code>\$ 1 \colon 3 \$</code>
	$1 \div 3$	<code>\$ 1 \div 3 \$</code>
	$\frac{1}{3}$	<code>\$ \frac{1}{3} \$</code>
Plus-Minus	± 3	<code>\$ \pm 3 \$</code>
Minus-Plus	∓ 3	<code>\$ \mp 3 \$</code>
Index	x_i	<code>\$ x_{i} \$</code>
Exponent	x^2	<code>\$ x^{2} \$</code>
Wurzel	\sqrt{x}	<code>\$ \sqrt{x} \$</code>
	$\sqrt[3]{x}$	<code>\$ \sqrt[3]{x} \$</code>
Summe	$\sum_{i=1}^n x_i$	<code>\$ \sum_{i=1}^n x_i \$</code>
	$\sum_{i=1}^n x_i$	<code>\$ \displaystyle \sum_{i=1}^n x_i \$</code>
Vergleiche	$< \leq \geq >$	<code>\$ < \leq \geq > \$</code>
Betrag	$ x $	<code>\$ \lvert x \rvert \$</code>

9.2 Mehrzeilige Gleichungen

mehrzeilige Gleichungen mit Kommentaren

$$\begin{array}{rcl}
 0 & = & 3x + 5 \\
 0 & = & x + \frac{5}{3} \\
 -\frac{5}{3} & = & x
 \end{array}
 \qquad
 \begin{array}{l}
 //: 3 \\
 // - \frac{5}{3}
 \end{array}$$

```

\usepackage{amsmath}

\begin{align*}
0 & \quad & \& = 3x + 5 & \quad & \& \& \text{//} \colon 3 \quad \backslash\backslash \\
0 & \quad & \& = x + \frac{5}{3} & \quad & \& \& \text{//} - \frac{5}{3} \quad \backslash\backslash
\end{align*}

```

```

-\frac{5}{3} & = x
\end{align*}

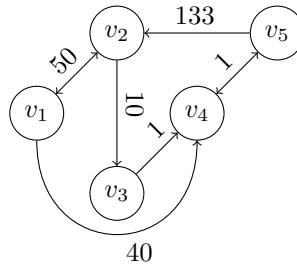
```

9.3 Mengen und Logische Formeln

Aussehen	Code
\in bzw. \ni	<code>\$ \in \$</code> bzw. <code>\$ \ni \$</code>
\notin	<code>\$ \notin \$</code>
\subseteq bzw. \supseteq	<code>\$ \subseteq \$</code> bzw. <code>\$ \supseteq \$</code>
\setminus	<code>\$ \setminus \$</code>

9.4 Graphen

Mit `\usepackage{tikz}` kann man solche Graphen basteln:



```

\begin{tikzpicture}[node distance={15mm}, main/.style = {draw, circle}]
  \node[main] (1) {$g_1$};
  \node[main] (2) [above right of=1] {$g_2$};
  \node[main] (3) [below right of=1] {$g_3$};
  \node[main] (4) [below right of=2] {$g_4$};
  \node[main] (5) [above right of=4] {$s$};

  \draw[<->] (1) -- node[midway, above right, sloped, pos=0.1] {50} (2);
  \draw[->] (2) -- node[midway, above right, sloped, pos=0.2] {10} (3);
  \draw[->] (3) -- node[midway, above right, sloped, pos=0.5] {1} (4);
  \draw[->] (1) to [out=270, in=270, looseness=2] node[midway, below right, sloped,
    pos=0.5] {40} (4);
  \draw[->] (5) -- node[midway, above right, sloped, pos=0.8] {133} (2);
  \draw[<->] (5) -- node[midway, above right, sloped, pos=0.8] {1} (4);
\end{tikzpicture}

```

10 Programmieren. Code Snippets

Siehe auch: [Overleaf Dokumentation zu Code listing \(overleaf.com/learn/latex/Code_listing\)](https://overleaf.com/learn/latex/Code_listing)

10.1 Verbatim

Eine Möglichkeit, **Code Snippets** in einem \LaTeX Dokument einzubauen, ist *verbatim*:

```
\begin{verbatim}
#include <stdio.h>

int main() {
    puts("Hallo Welt!");
    return 0;
}
\end{verbatim}
```

10.2 Listings

Eine andere Möglichkeit, die mehr optische Anpassung bietet, ist *listings*. Dazu muss folgendes in den Kopf des Dokuments:

```
\usepackage{listings}
```

Das Code Snippet kann folgendermaßen eingebunden werden:

```
\begin{lstlisting}
#include <stdio.h>

int main() {
    puts("Hallo Welt!");
    return 0;
}
\end{lstlisting}
```

Die optische Anpassung kann beispielsweise so aussehen:

```
%% gern noch etwas rumspielen bissl richtig schick aussieht
\lstdefinestyle{mystyle2}{
  language={[LaTeX]TeX},
  frame=single,
  rulecolor=\color{rot},
  breaklines=true,
  basicstyle=\tt,
  keywordstyle=\color{hellblau},
  %commentstyle=\color{rgb}{0.13,0.54,0.13},
  commentstyle=\color{gruen},
  %backgroundcolor=\color{dunkelblau},
  columns=flexible,
  morekeywords={
    maketitle,
    tableofcontents,
    subsection,
    subsubsection,
    hypersetup,
    urlstyle,
    href,
    hyperref,
    autoref,
    setlist,
    graphicspath,
    includegraphics}
}
\lstset{style=mystyle2}
```

10.3 Algorithmen

Mit dem package `algorithm2e` kann man Algorithmen schön formatieren. [ausführliche Doku](#)

11 *Ich mal mir die Welt, wie sie mir gefällt...* Farben

Siehe auch [Overleaf Dokumentation zu Farben](https://overleaf.com/learn/latex/Using_colours_in_LaTeX) (overleaf.com/learn/latex/Using_colours_in_LaTeX).

Dazu braucht man das Package:

```
\usepackage{xcolor}
```

Folgende Farben können dann benutzt werden:

	red
	green
	blue
	cyan
	magenta
	yellow
	black
	gray
	white
	darkgray
	lightgray
	brown
	lime
	olive
	orange
	pink
	purple
	teal
	violet

Andere Farben können so definiert werden:

```
\definecolor{rot}{RGB}{239, 71, 111}
\definecolor{gelb}{RGB}{255, 209, 102}
\definecolor{gruen}{RGB}{6, 214, 160}
\definecolor{hellblau}{RGB}{17, 138, 178}
\definecolor{dunkelblau}{RGB}{7, 59, 76}
```