

Anleitung und Hilfestellungen zur Vorlage für  
Projekt- und Abschlussarbeiten von Herrn Prof.  
Haffner in L<sup>A</sup>T<sub>E</sub>X

Florian Meier, B.Eng.

Betreuer:

Prof. Dr. rer. nat. E.-G. Haffner

Datum:

28. Februar 2022



# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>1</b>
1.1. T <sub>E</sub> X . . . . .	1
1.2. L <sup>A</sup> T <sub>E</sub> X . . . . .	2
<b>2. Grundlagen von L<sup>A</sup>T<sub>E</sub>X</b>	<b>3</b>
2.1. L <sup>A</sup> T <sub>E</sub> X-Editor . . . . .	3
2.2. nützliche Befehle . . . . .	4
2.2.1. Grafiken/visuelle Darstellungen . . . . .	4
2.2.2. Tabellen . . . . .	6
2.2.3. Code-Auszüge . . . . .	7
2.2.4. Auflistungen . . . . .	10
2.2.5. Zitieren . . . . .	11
2.2.6. Mathematik und Formeln . . . . .	12
2.2.7. Sonstiges . . . . .	15
2.3. packages/Pakete . . . . .	15
<b>3. Umgang mit der Vorlage</b>	<b>17</b>
3.1. <i>main.tex</i> . . . . .	18
3.2. <i>not-body</i> . . . . .	18
3.2.1. <i>a_Nutzereingaben.tex</i> . . . . .	18
3.2.2. <i>b_packages.tex</i> . . . . .	19
3.2.3. <i>c_Code-Darstellung.tex</i> . . . . .	19
3.2.4. <i>d_newcommands.tex</i> . . . . .	19
3.2.5. <i>e_Titelseite.tex</i> . . . . .	19
3.2.6. <i>f_Erklärung.tex</i> . . . . .	20
3.2.7. <i>g_Danksagung</i> . . . . .	20
3.2.8. <i>h_Abstract.tex</i> . . . . .	20
3.2.9. <i>i_Abkürzungsverzeichnis.tex</i> . . . . .	20
3.2.10. <i>j_Verzeichnisse.tex</i> . . . . .	20
3.2.11. <i>z_AnhangA.tex</i> und <i>z_AnhangB.tex</i> . . . . .	20
3.3. <i>stlyes</i> . . . . .	21
3.4. <i>chapters</i> . . . . .	21
3.5. <i>appendix, code</i> und <i>images</i> . . . . .	22
3.6. <i>literatur.bib</i> . . . . .	22
<b>A. Anhang - L<sup>A</sup>T<sub>E</sub>X-Referenzen</b>	<b>23</b>
<b>Verzeichnisse</b>	<b>28</b>
Literaturverzeichnis . . . . .	28
Abbildungsverzeichnis . . . . .	31
Tabellenverzeichnis . . . . .	33
Code-Auszugs-Verzeichnis . . . . .	35



*Aller Anfang ist schwer.*

Publius Ovidius Naso,  
römischer Epiker

# 1

## Einleitung

Dieses Dokument dient dazu Studenten, die bei Prof. Dr. rer. nat. E.-G. Haffner eine Projekt- oder Abschlussarbeit schreiben, einen Überblick über die Handhabung der bereitgestellten L<sup>A</sup>T<sub>E</sub>X-Vorlage zu geben. Dabei sei erwähnt, dass es sich bei dieser Vorlage um eine im Februar 2022 neu erstellte beziehungsweise überarbeitet und angepasst Version einer viel älteren Vorlage handelt. Ist dies beim Lesen dieses Dokuments bereits wieder einige Jahre her, so kann es sein, dass ein Aktualisieren der Vorlage wieder nötig ist.

Zunächst werden Informationen zu T<sub>E</sub>X und L<sup>A</sup>T<sub>E</sub>X selbst gegeben. Danach wird erläutert wie mit der Vorlage umzugehen ist. Daraufhin werden die wichtigsten Befehle erläutert, die L<sup>A</sup>T<sub>E</sub>X selbst und die eingebundenen Pakete bereitstellen. Als Letztes folgt dann eine Auflistung der Links zu allen verwendeten Paketen, sowie einigen nützlichen Dokumenten.

**Dabei stellt diese Anleitung keinerlei Anspruch auf Vollständigkeit.**

### 1.1. T<sub>E</sub>X

„L<sup>A</sup>T<sub>E</sub>X ist eine Sammlung von Makros, welche die Benutzung des Textverarbeitungssystems T<sub>E</sub>X vereinfacht. T<sub>E</sub>X wurde von Donald Knuth, einem emeritierten Professor der Stanford University, von 1977–1986 entwickelt und ist hauptsächlich ein Programm, das aus Quellcode eine Binärdatei generiert, welche dann in Textdokumente umgewandelt werden kann. T<sub>E</sub>X dient der Erstellung von jeglicher Art von Texten: kürzere Dokumente, Briefe, Dissertationen sowie Bücher. Was T<sub>E</sub>X-Dokumente charakterisiert ist, dass sie von außerordentlicher Qualität sind und dass die Handhabung mathematischer und technischer Formeln und Ausdrücke sehr vereinfacht wird. Mittels T<sub>E</sub>X wird aus dem Quellcode eine DVI-Datei erzeugt. Hierbei steht DVI für „Device Independent“, was darauf zurückzuführen ist, dass T<sub>E</sub>X und die generierten Dateien betriebssystemunabhängig sind. Um die Dateien dann auszudrucken oder anzusehen, müssen diese in ein Format, wie z. B. PS oder PDF, umgewandelt werden.“ ((ÖÖ15))

## 1.2. $\LaTeX$

„ $\LaTeX$  wurde von Leslie Lamport entwickelt und ist im Grunde eine Sammlung von  $\TeX$ -Makros, also eine Sammlung von Befehlen, welche eine Folge von Anweisungen in  $\TeX$  ausführen. Somit stellt  $\LaTeX$  den Benutzern einfache Befehle bereit, mit denen komplexe Operationen ausgeführt werden können. Die in  $\LaTeX$  eingebauten Makros dienen z. B. dazu, um Dokumentklassen zu definieren, Absätze und Texte zu strukturieren, Tabellen zu generieren und Referenzen einzubauen. Diese können durch zahlreiche Pakete erweitert werden, sodass es für fast jeden Formatierungsfall ein Paket mit den notwendigen Makros gibt. Die derzeitige Version von  $\LaTeX$  ist  $\LaTeX_{\epsilon}$ , welche zuerst 1994 erschien. Da  $\LaTeX$  eine Makrosammlung von  $\TeX$ -Befehlen darstellt, wird hier auch eine DVI-Datei generiert. Um eine PDF-Datei zu erzeugen, kann man die Funktion *dvi2pdf*, welche üblicherweise in einer  $\LaTeX$ -Installation vorhanden ist, benutzen. Alternativ kann man auch – anstatt mit  $\LaTeX$  die Datei zu kompilieren –  $\text{PDF}\LaTeX$  benutzen, womit direkt eine PDF-Datei erzeugt wird.“ ((ÖÖ15))

## 2

Grundlagen von  $\text{\LaTeX}$ 2.1.  $\text{\LaTeX}$ -Editor

Um mit  $\text{\LaTeX}$  Dokumente erstellen zu können reicht es aus einen einfachen Texteditor zum schreiben des Codes und Kenntnisse über  $\text{\TeX}$ -spezifischen Konsolenbefehle zu haben.

Dies ist allerdings äußerst unkomfortabel. Daher sei hier dringlichst dazu geraten einen  $\text{\LaTeX}$ -Editor zu verwenden. Da es viele potentielle Kandidaten gibt finden hier zwei Ausgewählte Erwähnung:

**Overleaf** ist ein Cloud-basierter Editor der Firma *WriteLaTeX Limited*<sup>®</sup>. Es ist von Nöten sich einen Account anzulegen. Zu den Stärken des Editors zählen gute Befehl-Vervollständigungsvorschläge, eine online Historie um Vergangene Versionen, sowie Änderungen (gewollt wie ungewollt) anzusehen und wenn gewünscht auch rückgängig zu machen, die Möglichkeit zeitgleich mit mehreren Personen an einem Projekt zu arbeiten und das Kompilieren der Ausgabe-Datei unabhängig davon, welche Quelldatei gerade ausgewählt ist.

Webseite: <https://www.overleaf.com/>

**$\text{\TeX}$ -Maker** hingegen ist ein Editor, der auf einem lokalen Rechner installiert werden muss. Allerdings muss zunächst eine 'Basis' geschaffen werden. Diese 'Basis' wird durch eine  $\text{\TeX}$ -Distribution zur Verfügung gestellt. Dabei kann auch wieder eine von vielen ausgewählt werden. Der Autor dieser Anleitung empfiehlt  $\text{MiK}\text{\TeX}$ , da mit dieser Distribution bereits gute Erfahrungen gesammelt werden konnten.  $\text{MiK}\text{\TeX}$  stellt dabei nicht nur  $\text{\TeX}$  an sich zur Verfügung sondern besitzt auch einen integrierten Manager für  $\text{\TeX}$ -Pakete (engl. 'packages'). Diese packages stellen dabei öffentlich verfügbare Ergänzungen zu  $\text{\LaTeX}$  dar, die viele weitere Funktionen implementieren (siehe dazu Kapitel REF).  $\text{MiK}\text{\TeX}$  lädt alle im Dokument angegebenen packages herunter, falls diese noch nicht auf dem Rechner existieren. Außerdem können alle heruntergeladenen Pakete über einen Knopfdruck aktualisiert werden. Regelmäßiges aktualisieren wird empfohlen um eventuellen, irreführenden Fehlermeldungen vorzubeugen.  $\text{\TeX}$ -Maker selbst sieht vom grundlegenden Aufbau ähnlich aus wie Overleaf. Es gibt einen Bereich für den  $\text{\LaTeX}$ -Code und einen Bereich für die dargestellte Datei. Auch der Log kann dargestellt werden. Es bieten sich weitere Einstellungsmöglichkeiten um das Aussehen

des Editors weiter anzupassen.

Webseite-MikT<sub>E</sub>X: <https://miktex.org/>

Webseite-T<sub>E</sub>X-Maker: <https://www.xmlmath.net/texmaker/>

## 2.2. nützliche Befehle

Die erste Zeile eines L<sup>A</sup>T<sub>E</sub>X-Projekts sollte immer die Definition der Dokumentenklasse beinhalten. Dies geschieht mittels des Befehls `\documentclass[...]`. In den geschweiften Klammern muss dabei die Klasse des Dokuments angegeben werden und in den eckigen können mit Parametern Einstellungen innerhalb der ausgewählten Klasse vorgenommen werden. Die Vorlage enthält bereits eine solche Zeile, die die Darstellung entsprechend der Vorstellungen von Prof. Dr. rer. nat. E.-G. Haffner einstellt.

Eines der Kernbefehlspaare in L<sup>A</sup>T<sub>E</sub>X ist `\begin{document}` und `\end{document}`. Zwischen diesen beiden Befehlen wird der Text des Dokuments angegeben. Vor dem ersten der beiden Befehle befindet sich die Präambel. In dieser werden Einstellungen vorgenommen, Befehle definiert und Pakete geladen, die für das Dokument von Nöten sind. Den überwiegenden Großteil der eigenen Arbeit wird also zwischen den beiden Befehlen geschrieben.

Nicht von allen in L<sup>A</sup>T<sub>E</sub>X-Schreibern genutzt, aber dennoch sehr hilfreich und in der Vorlage viel benutzt ist der Befehl `\input{...}`. Dieser ermöglicht es andere Dateien im Quelltext einzubinden. Außerdem kann er verschachtelt verwendet werden. Somit kann eine höhere Übersichtlichkeit geschaffen werden, da so die einzelnen Teile des L<sup>A</sup>T<sub>E</sub>X-Projekts (Definitionen, Paketeinbindungen, Kapitel, Anhang, ...) in eigenen Dateien abgelegt werden können.

Vergleichbar dazu ist der Befehl `\include{...}`. Mit diesem ist Ähnliches wie mit dem vorangegangenen Befehl möglich, allerdings bietet dieser Befehl keine Möglichkeit zur verschachtelten Verwendung, erzeugt aber einen Seitenvorschub. Interessant ist noch, dass `\include{...}` nur zwischen `\begin{document}` und `\end{document}` verwendet werden sollte.

### 2.2.1. Grafiken/visuelle Darstellungen

Oft ist es nötig Grafiken, Bilder oder andere visuelle Darstellungen in der eigenen Dokumentation einzubinden. Dazu wird das Befehlspaar `\begin{figure}` und `\end{figure}` verwendet. Innerhalb dieser so definierten *figure*-Umgebung kann dann mittels des Befehls `\includegraphics[...]` eine Bilddatei eingebunden werden. Dabei wird in den geschweiften Klammern der Pfad zu der Datei angegeben, die eingebunden werden soll. Es ist nicht immer nötig, aber dennoch immer sinnvoll die Endung der Datei mit anzugeben. Die eckigen Klammern können zusätzliche



Parameter enthalten, beispielsweise die Breite bei der Einbindung. Oft wird auch der Befehl `\centering` verwendet um dafür zu sorgen, dass die Grafik mittig im Textbereich dargestellt wird. Abbildung 2.1 zeigt eine häufig verwendete Kombination der beschriebenen Befehle.

```
\begin{figure}[ht]
  \centering
  \includegraphics[width=\textwidth]{images/includegraphics.PNG}
  \caption{Bilddateien in \LaTeX\ einbinden}
  \label{fig:ingr}
\end{figure}
```

**Abbildung 2.1.:** Bilddateien in  $\LaTeX$  einbinden

Um Bezüge in der eigenen Dokumentation realisieren zu können, ohne bei Änderungen in der Reihenfolge des Dokuments auch die Bezüge ändern zu müssen, wird der Befehl `\label{...}` verwendet. In den geschweiften Klammern muss eine im Dokument einzigartige Bezeichnung angegeben werden. Somit ist es möglich sich mittels des Befehls `\ref{...}` auf das Objekt zu beziehen, welches das in den geschweiften Klammern angegebene Label enthält. In Abbildung 2.1 wurde ein Label gesetzt, auf das sich bezogen wird, wenn der Befehl `\ref{fig:ingr}` eingegeben wird.

Diese Möglichkeit Objekte zu referenzieren beschränkt sich allerdings nicht nur auf *figures* sondern auch auf Kapitel (*chapter*), Sektionen (*section*), Untersektionen (*subsection*), Tabellen (*table*), Code-Auszüge (*listing*) und Formeln (*equation*). Es ist nicht verpflichtend aber ratsam sich an die Konvention zu halten jeder Gruppe von Objekten Labels zu geben, die mit je einem gleichen Bezeichner beginnen. Beispielsweise 'chap:...' für Kapitel oder 'fig:...' für Grafiken.

# 42

**Abbildung 2.2.:** 42

für die *figure* vorgesehene Breite.

Abbildung 2.2 stellt eine solche *wrapfigure* dar. Der zur Darstellung verwendete Code wird in Abbildung 2.3 gezeigt.

Auch ist es möglich Abbildungen im Fließtext einzubinden, sodass der Text neben der Abbildung entlang läuft. Dazu muss das Befehlspaar `\begin{wrapfigure}{...}{...}` und `\end{wrapfigure}` genutzt werden. Im zweiten Paar geschweifter Klammern muss dabei die Orientierung der einzufügenden Grafik angegeben werden (siehe dazu <https://www.overleaf.com/wrapfigure>). Das dritte Paar geschweifter Klammern dient zur Angabe der Breite des

Auch ist es möglich in einer *figure* mehrere *subfigures* anzulegen. Dazu wird das

```

\begin{wrapfigure}{L}{0.3\textwidth}
  \centering
  \includegraphics[width=0.2\textwidth]{images/42.png}
  \caption{42}
  \label{fig:42}
\end{wrapfigure}

```

Abbildung 2.3.: L<sup>A</sup>T<sub>E</sub>X-Code zur Verwendung von *wrapfigure*

Befehlspaar `\begin{subfigure}` und `\end{subfigure}` innerhalb einer *figure*-Umgebung pro einzufügender Grafik einmal verwendet. Es sei auch hier wieder auf eine Overleaf-Seite verwiesen: <https://www.overleaf.com/subfigures>

Wie in Abbildung 2.1 zu sehen können dem `\begin{figure}`-Befehl mittels der eckigen Klammern Parameter übergeben werden. Diese geben an, wo sich diese *figure* bevorzug befinden soll. Dabei gibt es folgende Möglichkeiten, die kombiniert werden können:

Tabelle 2.1.: Parameter zu Positionierung von *floats*

Symbol	Bedeutung
h	Platziert die <i>float</i> -Umgebung hier (ungefähr die gleiche Position, an der die Umgebung im Quelltext steht, allerdings nicht exakt an dieser Stelle)
t	Platziert die Umgebung oben an der Seite (top)
b	Platziert die Umgebung unten an der Seite (bottom)
p	Platziert die Umgebung auf einer separaten Seite nur für <i>floats</i>
!	Überschreibt interne Parameter, die L <sup>A</sup> T <sub>E</sub> X verwendet um 'gute' Positionen für <i>floats</i> zu bestimmen
H	Platziert die Umgebung an exakt der Position, an der sie im Quelltext auftritt; benötigt die Zeile <code>\usepackage{float}</code> ; ist ungefähr äquivalent zu h!; bei beiden Varianten können allerdings Fehlermeldungen auftreten, wenn zu viele <i>float</i> -Umgebungen hintereinander folgen.

### 2.2.2. Tabellen

Vielen der für *figures* relevanten Dinge treffen auch auf Tabellen zu. Reguläre Tabellen werden mit dem Befehlspaar `\begin{table}` und `\end{table}` gekennzeichnet. Dieses Befehlspaar stellt die *float*-Umgebung für Tabellen dar.

Innerhalb der benannten Umgebung wird dann mittels dem Befehlspaar `\begin{tabular}` und `\end{tabular}` die eigentliche Tabelle definiert. Das

zweite Paar geschweiften Klammern beim *begin*-Befehl enthält dabei indirekt Angaben darüber wie viele Spalten die Tabelle haben soll und direkte Angaben über die Formatierung der Spalten und das Layout. Tabelle 2.2 ist eine sehr kleine Tabelle, mit sehr einfachem Design, aber umfasst bereits 9 Zeilen. Der Code für diese Tabelle ist in Abbildung 2.4 zu sehen.

Da das händische Anlegen von größeren Tabellen sehr mühselig sein kann, existieren bereits mehrere Online-Werkzeuge zum Erzeugen und vereinfachten Editieren von Tabellen. Eines dieser Werkzeuge ist unter der Adresse [https://www.tablesgenerator.com/latex\\_tables](https://www.tablesgenerator.com/latex_tables) zu erreichen.

**Tabelle 2.2.:** test-table

This	is
a	test-table.

```
\begin{table}[h]
\centering
\caption{test-table}
\label{tab:bsptab}
\begin{tabular}{c|c}
This & is \\
a & test-table.
\end{tabular}
\end{table}
```

**Abbildung 2.4.:** L<sup>A</sup>T<sub>E</sub>X-Code von Tabelle 2.2

Neben der bisher genannten Möglichkeit Tabellen anzulegen gibt es eine weitere: Die *longtable*-Umgebung. Um diese zu nutzen muss das Befehlspaar `\begin{longtable}{...}` und `\end{longtable}` verwendet werden. Zwischen diesen beiden Befehlen befindet sich dann wieder der Inhalt. Die Tabelle 2.1 wurde als *longtable* realisiert. Der Code zu ihr findet sich in Abbildung 2.5 wieder. Die vor einigen Zeilen benannte Webseite zum Erstellen von Tabellen ([https://www.tablesgenerator.com/latex\\_tables](https://www.tablesgenerator.com/latex_tables)) generiert *longtables*, wenn die Option 'Split table in multiple pages' ausgewählt ist. Der Vorteil der *longtable*-Umgebung ist nämlich, dass sich diese über mehrere Seiten erstrecken kann.

### 2.2.3. Code-Auszüge

In vielen Arbeiten ist es nötig mindestens auszugsweise den geschriebenen Code darzustellen. Dazu sei empfohlen die Befehle des packages *listings* zu verwenden. Beispielhaft stellt Code-Auszug 2.1 einen simplen C++ Programmausschnitt dar.

```

\begin{longtable}[c]{|c|m{\textwidth-2.5cm}|}
\caption{Parameter zu Positionierung von \textit{floats}}
\label{tab:posparam}\hline
\hline
\textbf{Symbol} & \textbf{Bedeutung} \\
\endfirsthead
%
\endhead
%
h & Platziert die \textit{float}-Umgebung hier (ungefähr die gleiche Position, an der die Umgebung im Quelltext steht,
allerdings nicht exakt an dieser Stelle) \hline
t & Platziert die Umgebung oben an der Seite (top) \hline
b & Platziert die Umgebung unten an der Seite (bottom) \hline
p & Platziert die Umgebung auf einer separaten Seite nur für \textit{floats} \hline
! & Überschreibt interne Parameter, die LATEX verwendet um 'gute' Positionen für \textit{floats} zu bestimmen \hline
H & Platziert die Umgebung an exakt der Position, an der sie im Quelltext auftritt; benötigt die Zeile \textbackslash
\textit{float}\}; ist ungefähr äquivalent zu h!; bei beiden Varianten können allerdings Fehlermeldungen auftreten,
wenn zu viele \textit{float}-Umgebungen hintereinander folgen. \hline
\end{longtable}

```

Abbildung 2.5.: L<sup>A</sup>T<sub>E</sub>X-Code von Tabelle 2.1, einer *longtable*

Dazu wird auch wieder eine *float*-Umgebung definiert. Diese wird mit dem Befehlspar `\begin{lstlisting}` und `\end{lstlisting}` angegeben. In den eckigen Klammern des ersten Befehls können mit Parametern verschiedene Einstellungen vorgenommen werden. Unter anderem können Bildunterschrift und Label bestimmt und die Programmiersprache und damit die Darstellung des Codes festgelegt werden. Im Rahmen dieser Vorlage sind 2 Code-Darstellungen definiert. Diese lauten 'chaff', zur Darstellung von C beziehungsweise C++ Code, und 'pyhaff', zur Darstellung von Python-Code. Gerne dürfen auch weitere eigene Stile entworfen oder die bisherigen 'verbessert' werden. Diese neuen Stile bitte mit Prof. Dr. rer. nat. E.-G. Haffner teilen, sodass sie in der Vorlage implementiert werden können. Es können allerdings auch viele andere, bereits definierte, im Paket enthaltene Darstellungsstile verwendet werden. Dazu sei auf die Dokumentation verwiesen, die in KAPITEL HIER EINFÜGEN!!! verlinkt ist.

Der zur Darstellung von Code-Auszug 2.1 verwendete L<sup>A</sup>T<sub>E</sub>X-Code ist in Abbildung 2.6 dargestellt.

```
1 printf("Hello World");
```

Code-Auszug 2.1: simpler C++ Programmausschnitt

```

\begin{lstlisting}[language=c++, caption={simpler C++
Programmausschnitt}, label={cod:cppbsp}]
printf("Hello world");
\end{lstlisting}

```

Abbildung 2.6.: L<sup>A</sup>T<sub>E</sub>X-Code des Code-Ausschnitts 2.1, einem *lstlisting*

Dabei kann auffallen, dass der dargestellte C++ Code in kompletter Rohform im L<sup>A</sup>T<sub>E</sub>X-Dokument niedergeschrieben ist. Bei längeren oder mehreren Code-

Auszügen kann so also der L<sup>A</sup>T<sub>E</sub>X-Code aufgebläht werden.

Um das zu umgehen bietet das *listings*-Package einen weiteren Befehl an: `\lstinputlisting[...]`. Dabei können in den eckigen Klammern die gleichen Einstellungen getroffen werden, wie bei dem vorangegangenen Befehlspaar. In den geschweiften Klammern allerdings muss der Pfad zu der Datei angegeben werden, die in der Dokumentation dargestellt werden soll. Es kann sich einfach um eine \*.txt-Datei handeln, in die der entsprechende Code geschrieben wurde, aber auch Dateien mit Endungen entsprechend der verwendeten Programmiersprache sind möglich (beispielsweise: \*.cpp, \*.py, ...).

```

1  # add a select-field
2  def add_select_field(self, target="", preselect="", content="",
    content_value="", field_type="submit", css_class="linkDefault"):
3
4      if len(content) != len(content_value):
5          result = self.b("Error_in_HTML:_") + "HTML.add_select_field
            _content_and_content_value_have_different_sizes"
6      else:
7          result = "<select_class=\""+css_class+"\" _name=\""+target+"\"
            \"
8          if field_type == "submit":
9              result += "_onChange=\"this.form.submit()\" \"
10             result += ">"
11             for i in range(len(content)):
12                 result += "<option_value=\""+content_value[i]+"\" \"
13                 if content_value[i] == preselect:
14                     result += "_selected_";
15                 result += ">"
16                 result += content[i]
17                 result += "</option>"
18             result += "</select>"
19     self.add_code("\n" + result)

```

Code-Auszug 2.2: Python-Code als *lstinputlisting*

Der L<sup>A</sup>T<sub>E</sub>X-Code zu Code-Auszug 2.2 wird in Abbildung 2.7 gezeigt.

```

\lstinputlisting[language=pyhaff, caption=Python-Code als
\textit{lstinputlisting}, label=cod:lstinp]{code/Test-Code.txt}

```

Abbildung 2.7.: L<sup>A</sup>T<sub>E</sub>X-Code des Code-Ausschnitts 2.2

Berechtigerweise kann sich der Leser an diesem Punkt die Frage stellen, wieso der bisher in diesem Dokument gezeigte L<sup>A</sup>T<sub>E</sub>X-Code als Bilder, also als *figures* dargestellt wurde. Prinzipiell gibt es dafür drei Gründe:

1. Es gibt keine im *listings*-Package vordefinierte Sprache zur Darstellung von L<sup>A</sup>T<sub>E</sub>X-Code
2. Der L<sup>A</sup>T<sub>E</sub>X-Compiler hat Probleme damit L<sup>A</sup>T<sub>E</sub>X-Code innerhalb einer *listings/float*-Umgebung als nicht-auszuführenden L<sup>A</sup>T<sub>E</sub>X-Code zu erkennen.
3. Es wird vom Autor dieser Anleitung als sinnvoller erachtet zunächst *figures* einzuführen und diese dann auch vermehrt zu benutzen, da in Ausarbeitung in aller Regel auch weitaus mehr Abbildung als Code-Auszüge auftreten. In manchen Kreisen gilt das Einbinden von Code im Hauptteil einer Ausarbeitung als 'kein guter Stil'. Wenn es für Ihre Ausarbeitung allerdings sinnvoll ist können (und sollen!) Sie selbstverständlich auch Code-Auszüge verwenden.

### 2.2.4. Auflistungen

In der Untersektion 2.2.3, nur wenige Zeilen über dieser hier, wird eine Auflistung verwendet. Auch in der Subsektion 2.2.7 findet eine Auflistung Anwendung. Allerdings sehen beide unterschiedlich aus, da jeweils verschiedene Umgebungen verwendet wurden.

Für die spätere der beiden wird das Befehlspaar `\begin{itemize}` und `\end{itemize}` verwendet. In dieser Umgebung kann mit dem Befehl `\item` ein neues Element der Auflistung definiert werden. Der Code zur in Untersektion 2.2.7 verwendeten Auflistung wird von Abbildung 2.8 wiedergegeben.

```
\begin{itemize}
  \item \href{https://de.overleaf.com/learn}{https://de.overleaf.com/learn}
  \item \href{https://tex.stackexchange.com/}{https://tex.stackexchange.com/}
  \item \href{https://stackoverflow.com/}{https://stackoverflow.com/}
  \item ...
\end{itemize}
```

Abbildung 2.8.: L<sup>A</sup>T<sub>E</sub>X-Code der Auflistung der Untersektion 2.2.7

In Untersektion 2.2.3 wird hingegen die über ein Package eingebundene Umgebung *outline* verwendet, die mit dem Befehlspaar `\begin{outline}[]` und `\end{outline}` definiert wird. In den eckigen Klammern kann das Aussehen der Auflistung bestimmt werden. Die einzelnen Elemente werden hier stufenweise mit den Befehlen `\1`, `\2`, `\3`, ... angegeben. Dabei sinkt die Stufe in der Hierarchie der Auflistung mit steigender Zahl (`\3` ist in der Hierarchie niedriger als `\2`, etc.). In Abbildung 2.9 ist zu sehen, dass dort nur Elemente der ersten Hierarchiestufe verwendet wurden. Diese werden, wegen der Angabe *enumerate* in den eckigen Klammern, als Zahlen ausgeführt.

Generell gilt bei Auflistungen, aber auch bei Kapitelnummerierungen, die Faustregel, dass man nie nur einen Unterpunkt haben sollte!

```

\begin{outline}[enumerate]
  \1 Es gibt keine im \textit{listings}-Package vordefinierte Sprache zur Darstellung von
  \LaTeX-Code
  \1 Der \LaTeX-Compiler hat Probleme damit \LaTeX-Code innerhalb einer
  \textit{listings/float}-Umgebung als nicht-auszuführenden \LaTeX-Code zu erkennen.
  \1 Es wird vom Autor dieser Anleitung als sinnvoller erachtet zunächst \textit{figures}
  einzuführen und diese dann auch vermehrt zu benutzen, da in Ausarbeitung in aller Regel auch
  weitaus mehr Abbildung als Code-Auszüge auftreten. In manchen Kreisen gilt das Einbinden von
  Code im Hauptteil einer Ausarbeitung als 'kein guter Stil'. \quad Wenn es für Ihre
  Ausarbeitung allerdings sinnvoll ist können (und sollen!) Sie selbstverständlich auch
  Code-Auszüge verwenden.
\end{outline}

```

Abbildung 2.9.:  $\text{\LaTeX}$ -Code der Auflistung der Untersektion 2.2.3

### 2.2.5. Zitieren

Bevor es möglich ist im Text Schriftstücke anderer Autoren zu zitieren, muss zunächst dafür gesorgt werden, dass der  $\text{\LaTeX}$ -Datei diese Schriftstücke auch bekannt sind. Dazu wird eine Datei mit der Endung *.bib* in Kombination mit dem Befehl `\bibliography{*.bib}` verwendet. Dieser Befehl wird üblicherweise kurz vor Ende des Dokuments platziert, da er aus der *\*.bib*-Datei, die im Befehl angegeben ist direkt das Literaturverzeichnis generiert. Dieses soll in aller Regel nach dem Hauptteil mit einigen anderen Verzeichnissen positioniert sein.

In der Vorlage zu den Abschlussarbeiten heißt diese *\*.bib*-Datei *literatur.bib* (Gerne selbst einmal anschauen). Dieser können händisch Einträge für neue Schriftstücke hinzugefügt werden oder aber es werden Dienste wie **JabRef**<sup>®</sup> oder **Citavi**<sup>®</sup> verwendet. Es ist sinnvoll jede dieser Optionen zumindest einmal auszuprobieren, um Erfahrungen zu sammeln und mehr Verständnis zu erlangen, als es eine einfache Dokumentation jemals vermitteln könnte.

Das Zitieren kann mit dem Befehl `\cite{...}` vorgenommen werden. Dazu muss in den geschweiften Klammern ein eindeutiger Begriff zur Identifizierung der Quelle angegeben werden, der in der verwendeten *\*.bib*-Datei definiert ist. Hinter diesem Satz wird eine Zitierung vorgenommen, die angibt, dass dieser Satz (inhaltlich) aus dem angegebenen Schriftstück stammt (was in diesem Fall nicht stimmt, aber das Zitieren demonstrieren soll). (**Haf18**)

Mit dem Befehl `\textquoteleft[\cite{...}]{...}` können Sätze wortgetreu zitiert werden. Dabei werden diese in Anführungszeichen gesetzt. In den geschweiften Klammern steht dabei das Zitat, die eckigen hingegen können mit dem `\cite{...}`-Befehl die Quelle oder Quellen enthalten. So ergibt sich beispielsweise folgendes Zitat:

„Der Begriff »*linear*« lässt sich auf das griechische *linea* zurückführen und bedeutet eine gerade Linie, was sich auch im deutschen Wort »*Lineal*« widerspiegelt.“  
(**Haf18**)



Es sei an dieser Stelle auch darauf verwiesen, dass es verschiedene Darstellungsstile für die Literaturverzeichnisse gibt. Diese Datei und auch die Vorlage verwenden den Stil *alphadin*, der in DIN-Normen definiert ist. Es gibt allerdings mannigfaltig viele Stile, die hier nicht alle beleuchtet werden sollen. Ein alternativer Stil, der gerade bei Paper-Veröffentlichungen viel Anwendung findet ist der *IEEEtran*-Stil.



**Abbildung 2.10.:** verschiedene Stile des Literaturverzeichnisses

## 2.2.6. Mathematik und Formeln

### Mathematik-Modus (inline)

L<sup>A</sup>T<sub>E</sub>X ist vor allem dafür bekannt, dass es sich dafür eignet Formel und Berechnungen besonders ansprechend darzustellen. Um eben das zu tun muss der *math-mode/Mathematik-Modus* aktiviert werden. Um in den Textzeilen mathematische Formel darzustellen müssen diese zwischen zwei  $\$$ -Symbolen eingegeben werden. Das kann dann beispielsweise so  $U = R \cdot I$  oder so  $\frac{\partial u}{\partial t} = h^2 \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right)$  oder so  $\sum_{k=1}^n k^2 = \frac{1}{2}n(n+1)$  aussehen.

Um die diversen Symbole darzustellen werden, wie für L<sup>A</sup>T<sub>E</sub>X üblich, viele Befehle mit `'\'` verwendet. Eine Auswahl dieser finden sich in der PDF-Datei in Anhang A oder diverse Quellen im Internet wieder.

Auch für die mathematischen Formeln gibt es diverse Online-Editoren, beispielsweise sei hier <https://latexeditor.lagrida.com/> genannt.

### Gleichungen und Formeln mit Nummerierung

Zwar ist eine solche Darstellung im Fließtext oft nützlich, aber eben nicht immer sinnvoll. Viel eher ist es erstrebenswert Gleichungen und Formeln mit Nummerierung und Referenzmarker (label) in einer *float*-Umgebung einzubinden. Genau dazu gibt es mehrere Möglichkeiten. In diesem Dokument werden zwei davon dargestellt.

Das Befehlspaar `\begin{equation}` und `\end{equation}` implementiert eine solche Umgebung. Zwischen den beiden Befehlen kann dann Code eingegeben werden, ähnlich dem, der zuvor eingegeben wurde. Dabei ergeben sich folgende Gleichungen:



$$U = R \cdot I \quad (2.1)$$

$$\frac{\partial u}{\partial t} = h^2 \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) \quad (2.2)$$

$$\sum_{k=1}^n k^2 = \frac{1}{2} n(n+1) \quad (2.3)$$

Wichtig hierbei ist zu erwähnen, dass all diese Gleichungen in separaten *equation*-Umgebungen angegeben sind, da sie sonst nicht separat nummeriert werden würden. Diese separate Nummernierung ist aber aus Gründen der Referenzierbarkeit wünschenswert. So ist es beispielsweise möglich zu sagen, dass Gleichung 2.3 aufwändiger zu schreiben ist als Gleichung 2.1, allerdings ist 2.2 noch aufwändiger zu schreiben.

```
\begin{equation}
  U=R\cdot I
  \label{eq:URI}
\end{equation}

\begin{equation}
  \frac{\partial u}{\partial t} = h^2 \left( \frac{\partial^2 u}{\partial x^2} +
  \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right)
  \label{eq:partial}
\end{equation}

\begin{equation}
  \sum_{k=1}^n k^2 = \frac{1}{2} n (n+1)
  \label{eq:sum}
\end{equation}
```

**Abbildung 2.11.:**  $\text{\LaTeX}$ -Code der drei Gleichungen 2.1, 2.2 und 2.3

Will man allerdings gleich mehrere Gleichung aufschreiben und diese aufeinander beziehen oder Umformungen einer Gleichung strukturiert darstellen, so ist das Befehlspaar `\begin{align}`, `\end{align}`, dass die Umgebung *align* bereitstellt eine bessere Wahl.

$$\frac{V_i - V_1}{R} + \frac{V_o - V_1}{R} - V_1 \cdot 2sC = 0 \quad | \cdot R \quad (2.4)$$

$$V_i - V_1 + V_o - V_1 - V_1 \cdot 2s_n = 0 \quad | - (V_o + V_i) \quad (2.5)$$

$$-V_1 - V_1 - V_1 \cdot 2s_n = -V_o - V_i \quad | \cdot (-1) \quad (2.6)$$

$$V_1 \cdot 2(1 + s_n) = V_o + V_i \quad | : 2(1 + s_n) \quad (2.7)$$

$$V_1 = \frac{V_o + V_i}{2(1 + s_n)} \quad (2.8)$$

Die über dieser Textpassage stehenden Gleichungen sind in einer solchen *align*-Umgebung definiert. Dabei fällt auf, dass alle = Zeichen übereinander stehen, dass kann dank der geschickten Verwendung von &-Symbolen erreicht werden. Siehe dazu Abbildung 2.12.

```
\begin{align}
\frac{V_i - V_1}{R} + \frac{V_o - V_1}{R} - V_1 \cdot 2sC &= 0 && &| \cdot R \\
V_i - V_1 + V_o - V_1 - V_1 \cdot 2s_n &= 0 && &| - (V_o + V_i) \\
-V_1 - V_1 - V_1 \cdot 2s_n &= -V_o - V_i && &| \cdot (-1) \\
V_1 \cdot 2(1 + s_n) &= V_o + V_i && &| : 2(1 + s_n) \\
V_1 &= \frac{V_o + V_i}{2(1 + s_n)} \\
&\label{eq:p_v1}
\end{align}
```

Abbildung 2.12.: L<sup>A</sup>T<sub>E</sub>X-Code der *align*-Gleichungen

## SI-Einheiten

Das Ergebnis langer Berechnungen ist oft eine Zahl mit einer Einheit. Diese Einheiten sollen nach dem *Internationalen Einheitensystem* / *Système international d'unités* (**SI**) dargestellt werden. Dies kann auch wieder mühsam per Hand oder mit Hilfe eines Befehls (durch das package *siunitx* implementiert) geschehen. Der zu verwendende Befehl lautet `\SI{...}`. Dabei wird in den geschweiften Klammern angegeben welche Einheiten verwendet werden sollen.

$$F_1 = 5 \text{ kgm/s}^2 \quad (2.9)$$

$$F_2 = 5 \text{ kg m/s}^2 \quad (2.10)$$

Formel 2.9 verwendet keine SI-konforme Darstellung der Gleichung, Formel 2.10 hingegen schon (der Code dazu ist in Abbildung 2.13 zu sehen). Das Problem in der ersten der beiden Formel ist, dass sowohl Formelzeichen, als auch Einheiten gleich formatiert sind. Auch ist der Abstand der einzelnen Symbole in SI-Darstellung verkürzt.

```

\begin{align}
F_1 &= 5 \backslash \text{ kg m/s}^2 \backslash \text{label}\{eq:F_{noSI}\} \\
F_2 &= \backslash SI{5}\{kg.m/s^2\} \backslash \text{label}\{eq:F_{SI}\}
\end{align}

```

Abbildung 2.13.: L<sup>A</sup>T<sub>E</sub>X-Code der Gleichungen 2.9 und 2.10

## 2.2.7. Sonstiges

Weitere wichtige Befehle sind beispielsweise im Buch (ÖÖ15), sowie diversen Webseiten zu finden:

- <https://www.ctan.org/>
- <https://de.overleaf.com/learn>
- <https://golatex.de/>
- <https://tex.stackexchange.com/>
- <https://stackoverflow.com/>
- ...

Dabei sei erwähnt, dass oft eine Suche mit einer Suchmaschine der Wahl, dazu führt, dass man auf einer der Seiten landet und Antworten auf die formulierte Frage erhält.

## 2.3. packages/Pakete

In Teilen wurde bereits in Sektion 2.2 indirekt auf einige verwendete *packages* eingegangen. Allerdings umfasst die Vorlage weitaus mehr und auch das Einfügen weiterer gestaltet sich mit dem Befehl `\usepackage{...}` einfach.

Allerdings sollen hier gar nicht im Detail alle verwendeten oder gar alle existierenden *packages* und ihre Befehle erklärt werden. Diese Sektion dient nur zum Auflisten der Dokumentationen aller verwendeten Pakete.

Tabelle 2.3.: Dokumentationen zu allen verwendeten *packages*

<i>package</i>	Dokumentation
<b>Text- und Sprach-packages</b>	
inputec	<a href="https://www.ctan.org/pkg/inputenc">https://www.ctan.org/pkg/inputenc</a>
fontec	<a href="https://www.ctan.org/pkg/fontenc">https://www.ctan.org/pkg/fontenc</a>
babel	<a href="https://www.ctan.org/pkg/babel">https://www.ctan.org/pkg/babel</a>
<b>Layout-packages</b>	
geometry	<a href="https://www.ctan.org/pkg/geometry">https://www.ctan.org/pkg/geometry</a>
xcolor	<a href="https://www.ctan.org/pkg/xcolor">https://www.ctan.org/pkg/xcolor</a>

Fortführung von **Tabelle 2.3**

<i>package</i>	Dokumentation
hyperref	<a href="https://ctan.org/pkg/hyperref">https://ctan.org/pkg/hyperref</a>
chapterthumb	Keine Dokumentation vorhanden; bei Fragen: komascript.de
quotchap	<a href="https://www.ctan.org/pkg/quotchap">https://www.ctan.org/pkg/quotchap</a>
hhline	<a href="https://www.ctan.org/pkg/hhline">https://www.ctan.org/pkg/hhline</a>
multicol	<a href="https://www.ctan.org/pkg/multicol">https://www.ctan.org/pkg/multicol</a>
<b>Grafik-packages</b>	
graphicx	<a href="https://www.ctan.org/pkg/graphicx">https://www.ctan.org/pkg/graphicx</a>
wrapfig	<a href="https://www.ctan.org/pkg/wrapfig">https://www.ctan.org/pkg/wrapfig</a>
caption	<a href="https://www.ctan.org/pkg/caption">https://www.ctan.org/pkg/caption</a>
float	<a href="https://www.ctan.org/pkg/float">https://www.ctan.org/pkg/float</a>
afterpage	<a href="https://www.ctan.org/pkg/afterpage">https://www.ctan.org/pkg/afterpage</a>
tikz	<a href="https://www.ctan.org/pkg/pgf">https://www.ctan.org/pkg/pgf</a>
pgfplots	<a href="https://www.ctan.org/pkg/pgfplots">https://www.ctan.org/pkg/pgfplots</a>
pdfpages	<a href="https://www.ctan.org/pkg/pdfpages">https://www.ctan.org/pkg/pdfpages</a>
<b>Tabellen-packages</b>	
array	<a href="https://www.ctan.org/pkg/array">https://www.ctan.org/pkg/array</a>
longtable	<a href="https://www.ctan.org/pkg/longtable">https://www.ctan.org/pkg/longtable</a>
<b>Code-Auszugs-packages</b>	
listings	<a href="https://www.ctan.org/pkg/listings">https://www.ctan.org/pkg/listings</a>
<b>Auflistungs-packages</b>	
outline	<a href="https://www.ctan.org/pkg/outline">https://www.ctan.org/pkg/outline</a>
<b>Zitier-packages</b>	
csquotes	<a href="https://www.ctan.org/pkg/csquotes">https://www.ctan.org/pkg/csquotes</a>
natbib	<a href="https://www.ctan.org/pkg/natbib">https://www.ctan.org/pkg/natbib</a>
<b>Mathematik-packages</b>	
siunitx	<a href="https://www.ctan.org/pkg/siunitx">https://www.ctan.org/pkg/siunitx</a>
emheq	<a href="https://www.ctan.org/pkg/empheq">https://www.ctan.org/pkg/empheq</a>
amssymb	<a href="https://de.wikibooks.org/wiki/LaTeX-Kompendium:_amssymb">https://de.wikibooks.org/wiki/LaTeX-Kompendium:_amssymb</a>
mathrsfs	<a href="https://www.ctan.org/pkg/mathrsfs">https://www.ctan.org/pkg/mathrsfs</a>
bbm	<a href="https://www.ctan.org/pkg/bbm">https://www.ctan.org/pkg/bbm</a>

## 3

## Umgang mit der Vorlage

Der erste Schritt zur Verwendung der Vorlage ist das Herunterladen von der Alfresco-Seite des Projektteams von Prof. Dr. rer. nat. E.-G. Haffner. Dort muss in der Dokumentenbibliothek der Ordner *Leitfaden und Vorlage* geöffnet werden. In diesem befindet sich die Datei *Vorlage.zip*. Diese ist ein komprimiertes Archiv, dass die Vorlage enthält. Einmal heruntergeladen muss *Vorlage.zip* noch entpackt werden.

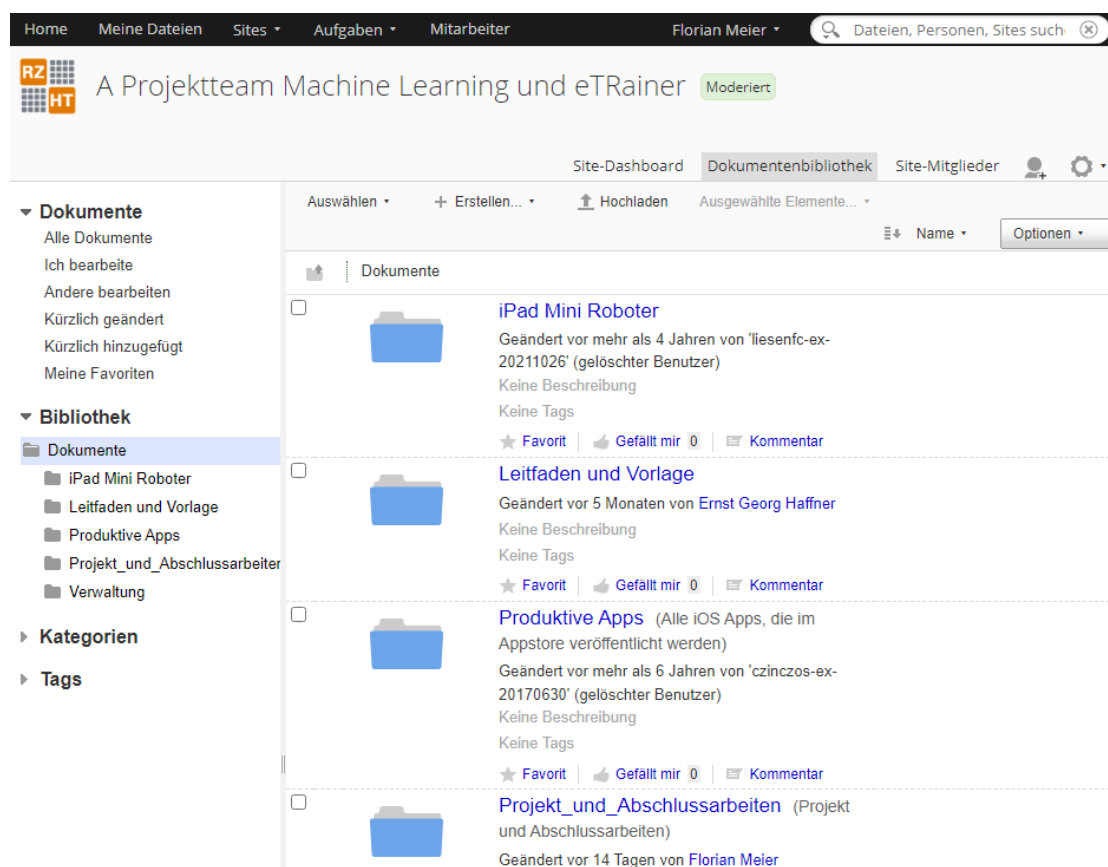


Abbildung 3.1.: Alfresco-Dokumentenbibliothek

In Abbildung 3.1 ist zu sehen, wo sich der Leitfaden-Ordner in der Alfresco-Gruppe befindet. Falls Sie in Besitz dieses Dokuments gekommen sind ohne Teil der Alfresco-Gruppe zu sein ist zwar fraglich, wie Sie überhaupt dieses Datei erhal-

ten haben, aber dennoch sollten Sie sich wegen eines Zugangs zur Alfresco-Gruppe beziehungsweise zur L<sup>A</sup>T<sub>E</sub>X-Vorlage bei Prof. Dr. rer. nat. E.-G. Haffner melden.

In den folgenden Subsektionen wird die Handhabung der Vorlage gegliedert nach *\*.tex*-Dateien erläutert. Dabei wird darauf verzichtet den Code der einzelnen Dateien abzubilden, weil der Code in der Vorlage zu sehen ist.

## 3.1. *main.tex*

In aller Regel muss in dieser Datei nicht oder kaum etwas geändert werden. *main.tex* ist mit Kommentaren der Form `% =====`... in verschiedene Bereiche eingeteilt. Diese sind:

- *documentclass*
- Präambel
- Formalien
- Inhalt
- Anhang und Verzeichnisse

Da die Datei bereits in den Kommentaren viel erläutert ist und hauptsächlich andere Dateien eingebunden werden wird nicht weiter auf *main.tex* eingegangen.

## 3.2. *not-body*

Im Ordner *not-body* befinden sich all die Dateien, die nicht zum Hauptteil der Ausarbeitung gehören, sondern Dinge vor oder nach diesem beinhalten. Es folgt eine alphabetische Auflistung der Dateien mit zugehöriger Erläuterung ihrer Daseinsberechtigung beziehungsweise ihrer Funktion. Dabei sei angemerkt, dass diese Reihenfolge auch die ist, mit der sie in *main.tex* aufgerufen werden.

### 3.2.1. *a\_Nutzereingaben.tex*

In dieser Datei kann der Nutzer Änderungen an Einstellungen beziehungsweise Angaben für das Deckblatt vornehmen, ohne in den jeweiligen Dateien, die eigentlich für diese zuständig sind etwas ändern zu müssen.

Dabei muss im ersten Abschnitt (*Angaben für das Deckblatt*) nur jeweils der Eintrag im zweiten Paar geschweifter Klammern geändert werden. Der Eintrag im jeweiligen ersten Paar geschweifter Klammern enthält den jeweiligen Befehl, der aufgerufen werden kann um den Eintrag aus dem zweiten Paar geschweifter Klammern darzustellen. Die Befehle (Einträge im ersten `{}`-Paar) dürfen **nicht** verändert werden! Gerne aber dürfen die Befehle, falls nötig im eigens Geschriebenen

aufgerufen werden.

Im zweiten Abschnitt (*Layout-Einstellungen*) werden zwei Zähler (engl. *counters*) definiert. Diesen wird standardmäßig jeweils der Wert 1 zugewiesen. Dies bewirkt (durch den in anderen Dateien geschriebenen Code), dass sowohl für den Hauptteil, als auch für den Anhang Kapitelbalken am rechten Rand rechter/ungerade Seiten erscheinen. Wird *akb* zu 0 gesetzt, so erhält der Anhang keine Kapitelbalken mehr. Mit *kb* und den Kapitelbalken im Hauptteil verhält es sich gleich.

### 3.2.2. *b\_packages.tex*

Wie der Name vermuten lässt enthält die Datei *b\_packages.tex* alle eingebundenen Pakete. Nachdem sich entsprechend in die Dokumentation der einzelnen Pakete eingesehen wurde dürfen deren Parameter geändert werden. Auch das Hinzufügen weiterer Pakete ist möglich und unter Umständen auch nötig. Dabei sollte darauf geachtet werden, die thematische Struktur zu wahren, also die neuen Pakete an entsprechender Stelle einzufügen.

### 3.2.3. *c\_Code-Darstellung.tex*

In der Datei *c\_Code-Darstellung.tex* werden Darstellungen für Code-Auszüge definiert. Dazu werden zunächst unter Nutzung des *package xcolor* Farben festgelegt, die zur Darstellung genutzt werden. Darauf folgt jeweils die Definition (Darstellungs-)Programmiersprache, unter Verwendung des `\lstdefinlanguage{...}[] {...}`-Befehls. Im ersten Paar geschweiften Klammern wird der Name der neu definierten Sprache festgelegt. Das zweite Paar dieser Klammern kann die Sprache enthalten, auf der die neu basieren soll.

Zum Zeitpunkt der Verfassung dieser Anleitung (28. Februar 2022) sind zwei Darstellungssprachen definiert: *'pyhaff'* zur Darstellung von Python-Code und *'chaff'* zur Darstellung von C- beziehungsweise C++ Code. Darüber hinaus können weitere im *listings-package* bereits vordefinierte Stile geladen und genutzt werden. Auch können weitere Stile definiert werden. Diese sollten auch in gleicher Art in der *c\_Code-Darstellung.tex*-Datei hinterlegt werden. Gerne dürfen diese neuen Stile auch mit Prof. Dr. rer. nat. E.-G. Haffner geteilt werden, sodass diese in der Vorlage implementiert werden können.

### 3.2.4. *d\_newcommands.tex*

In dieser Datei werden neue Befehle definiert, die zur Nutzung in der eigenen Ausarbeitung gedacht sind. Gerne dürfen auch neue und weitere definiert werden.

### 3.2.5. *e\_Titelseite.tex*

Auch hier gibt der Dateiname wieder, was Inhalt der Datei ist: Es wird die Titelseite definiert. Im Allgemeinen sollte dieses Datei nicht maßgeblich verändert werden. Bei Arbeiten in externen Unternehmen sollen die Zeilen mit *'im Unternehmen*

`|betrieb{}` und `|betreuer{}` aktiviert werden, also die Kommentare deaktiviert werden. Es darf auch das Aussehen der Titelseite verändert werden. Dazu wird allerdings nicht geraten.

### 3.2.6. *f\_Erklärung.tex*

In der Datei *f\_Erklärung.tex* werden Aussehen und Inhalt der Eidesstattlichen Erklärung definiert. Dabei sollte hier höchstens der Text der Erklärung verändert werden.

### 3.2.7. *g\_Danksagung*

In dieser Datei werden Aussehen und Inhalt der Danksagung definiert. Dabei sollte hier höchstens der Text der Danksagung verändert werden.

### 3.2.8. *h\_Abstract.tex*

In *h\_Abstract*-Datei werden Aussehen und Inhalt des *Abstract* definiert. Dabei gibt es ein *Abstract* in englischer Sprache und ein *Abstract* in deutscher Sprache unter dem Namen *Zusammenfassung*. Hier sollen die Texte, also der Inhalt beider *Abstracts* angepasst werden.

### 3.2.9. *i\_Abkürzungsverzeichnis.tex*

Hier wird das tabellarische Abkürzungsverzeichnis definiert. Um dieses auf die eigene Arbeit anzupassen muss der Inhalt verändert werden. Da es sich um eine tabellarische Darstellung handelt kann dazu auch ein Online-Tabellen-Werkzeug verwendet werden. Händisches Eintragen ist allerdings auch möglich.

### 3.2.10. *j\_Verzeichnisse.tex*

In *j\_Verzeichnisse* werden alle Verzeichnisse erzeugt, die am Ende der Arbeit aufgelistet werden. Dies umfasst Literatur-, Abbildungs-, Tabellen-, Code-Auszugs-Verzeichnis und den Glossar. Dabei werden die ersten vier automatisch aus den Daten der entsprechend definierten Umgebungen und deren Unterschriften generiert. Der Glossar muss händisch erstellt werden. Dabei ist er als Auflistung realisiert. Jedes der Aufgeführten Verzeichnisse wird auch im Inhaltsverzeichnis aufgeführt. Ist dies oder Verwenden solcher Verzeichnisse generell nicht nötig oder nicht gewünscht, so müssen die entsprechenden Zeilen auskommentiert werden.

### 3.2.11. *z\_AnhangA.tex* und *z\_AnhangB.tex*

In dieser Datei soll ein Anhang eingebunden werden. Da es vorkommen kann, dass mehrere Dinge angehängen werden müssen, die thematisch nicht zu einander passen wurden zunächst zwei verschiedene Anhangsdateien vorgesehen. Bei



Bedarf können weitere angelegt werden. Beispielhaft wurde in *z\_AnhangA.tex* geschrieben, dass dort Quellcode und in *z\_AnhangB.tex* die Dokumentation zu den erstellten Programmen aufgeführt werden soll. Andere Anhänge sind denkbar.

### 3.3. *stlyes*

Der Ordner *stlyes* enthält die Dateien *Advanced\_I.sty* und *chapterthump.sty*. Dabei ist Letzteres ein *package*, dass in *b\_packages.tex* eingebunden wird. Dieses Verfahren ist nötig, da *chapterthump.sty* nicht Teil der standardmäßig verfügbaren *packages* ist. Die Datei wurde in Version 0.1 verwendet und leicht modifiziert, um der aktualisierten Version der Vorlage gerecht zu werden. Es existieren auch aktuellere Version des *chapterthump-package*, allerdings tut die aktuell verwendete Version ihren Dienst. Es ist allerdings nicht auszuschließen, dass mit künftigen Erweiterungen der Vorlage auch eine Änderung der *chapterthump.sty*-Datei nötig wird.

Die andere der beiden Dateien (*Advanced\_I.sty*) hingegen enthält die Festlegung des Seitenstils, sowie die Definition des Aussehens von Kopf- und Fußzeile.

### 3.4. *chapters*

Im Ordner *chapters* sind insgesamt acht *\*.tex*-Dateien enthalten, von denen sieben beispielhaft für mögliche Teile der Ausarbeitung stehen und eine einfach nur ein wenig Code enthält, an dem manche Befehle und ihre Verwendung erkannt werden können. Diese eine Datei trägt den Namen *test\_chapter.tex*. Die anderen sieben Dateien heißen:

- *01\_Einleitung.tex*
- *02\_Grundlagen.tex*
- *03\_Hauptteil\_1.tex*
- *04\_Hauptteil\_2.tex*
- *05\_Hauptteil\_3.tex*
- *06\_Evaluation.tex*
- *07\_FazitundAusblick.tex*

Sie stellen eine gute Grundstruktur dar und dürfen gerne umbenannt werden. Dabei darf aber nicht vergessen werden die entsprechenden Namen ebenfalls in der *main.tex*-Datei anzupassen.

### 3.5. *appendix, code und images*

Die drei Ordner *appendix*, *code* und *images* sind dazu gedacht jeweils zusätzliche Dateien, die in der eigenen Dokumentation gezeigt werden sollen, zu enthalten. *appendix* ist dabei für Anhänge, *code* für Code beziehungsweise Code-Auszüge und *images* für Bilddateien gedacht. In *code* sind zwei Dateien enthalten, die lediglich zur Demonstration dienen und gelöscht werden dürfen. In *images* liegt nur die Datei *eTRainer-Logo.png*, die auf der Titelseite eingebunden wird.

### 3.6. *literatur.bib*

In der Datei *literatur.bib* sollen alle verwendeten Quelle aufgeschrieben werden und das Bib<sub>T</sub><sub>E</sub>X-konform (siehe dazu <https://latex-tutorial.com/tutorials/bibtex/>). Wie bereits in Subsektion 2.2.5 erwähnt können Dienste wie JabRef<sup>®</sup> oder Citavi<sup>®</sup> verwendet werden, um \*.bib-Dateien automatisiert aus Angaben zu erstellen. Dabei sei gesagt, dass Citavi<sup>®</sup> die klare Empfehlung des Autors ist.



# Anhang - L<sup>A</sup>T<sub>E</sub>X-Referenzen

# Reference Sheet for a Thesis with L<sup>A</sup>T<sub>E</sub>X2e and KOMA-Script

- All examples were tested with pdflatex.
- The package mentioned in the headings has to be included (see B.2).
- Compile three times after last change (esp. docs with references).

## A. L<sup>A</sup>T<sub>E</sub>X Basics

### A.1. Units

- Available units for length and dimensions:  
bp point (typographic) mm millimeter in inch em width of M  
px pixel (1/72in) cm centimeter pc pica ex height of x
- Document dependent units z\textwidth, z\linewidth, z\columnwidth, z\textheight with z a percentage value, e.g. 0.55\textwidth means 55% of the actual width of the text.
- \baselineskip minimum vertical space between the bottom of two successive lines in a paragraph.
- Amounts like \smallskipamount, \medskipamount, \bigskipamount.

### A.2. Reserved Characters (see also E.2, cf. H)

\	introduces a command	\textbackslash
{ }	embraces arguments, creates logical parts	\{ \}
[ ]	embraces <i>optional</i> arguments	[ ]
%	comments: code after % will be ignored.	%
&	separates columns in tabular-like environments	&
#	parameter for own command declarations	#
\$	text style math mode (abbr. for \(...\))	\$
^	index/exponent only valid in math mode, e.g. a <sub>1</sub> <sup>2</sup> see E.2	

## B. Preamble (before \begin{document})

### B.1. Documentclass (necessary)

Use: \documentclass[opt,opt,...]{class}  
Recommended classes: scrartcl, scrreprt, scrbook, scrllttr2  
Non-KOMA-Script classes: beamer, koma-moderncvclassic

Common options with default	Values available (subtotal)
fontsize=11pt	10pt   12pt (e.g. 12.5pt also valid)
paper=a4, paper=portrait	a3   a5   b4   letter, landscape
parskip=no	half   full
headings=big	small   normal
chapterprefix=false	true
open=right (scrbook)	any (scrartcl, scrreprt)   left
captions=oneline	nooneline
captions=tablebelow,figurebelow	tableabove, figureabove
toc=notlistof	listof   listofnumbered
bibliography=totoc   totocnumbered	nottotoc
twoside=true (scrbook)	false (scrartcl, scrreprt)
twocolumn=false	true
draft=false	true (show overfull boxes)

→ Options of document class are passed to every loaded package.  
→ Set or change options later in file, e.g. \KOMAoptions[twoside=true]

### B.2. Loading Packages

\usepackage[options]{package}  
\PassOptionsToPackage[options]{package}

### B.3. Encoding Settings

```
\usepackage[utf8]{inputenc} % most IDEs use UTF8
\usepackage[T1]{fontenc} % most fonts needs T1
```

### B.4. Language Settings with babel

Load: \usepackage[ngerman, main=english]{babel}  
Use: \selectlanguage{language} % foreignlanguage{language}{text}  
  

```
\documentclass[italian]{scrbook} % global option
\usepackage[british,main=italian]{babel} % package option
\usepackage{csquotes} % package csquotes knows italian
```

## C. Layout


### C.1. Changing Page Layout with geometry

- Let KOMA-Script know of geometry by option usegeometry=true.

```
\usepackage[left=2cm, right=2, top=3cm, bottom=4cm,
bindingoffset=1cm, includeheadfoot]{geometry}
```

→ Auto-completion determines unspecified dimensions (under or over specified as well), here width and height of text (see I.9).  
→ Other options: paper=a4paper, landscape|portrait, includehead, includefoot, includeheadfoot, twocolumn  
→ Changing page layout mid document: \newgeometry{opt, opt, ...}

### C.2. Header and Footer of Page (aka running heading)



```
% delete default settings and define your own
\usepackage[automark]{scrlayer-scrpage}
\clearpaïr of page styles
\head[][\headmark] \foot[][\pagemark]{\pagemark}
```

% Variant for a thesis with horizontal rules at head and foot  
\usepackage[headsepline=0.005pt:, footsepline=0.005pt:, plainfootsepline, automark]{scrlayer-scrpage}  
\clearpaïr of page styles  
\head[][\headmark] \foot[][\pagemark]{\pagemark}  
\ModifyLayer[addvoffset=-.6ex]{scrheadings.foot.above.line}  
\ModifyLayer[addvoffset=-.6ex]{plain.scrheadings.foot.above.line}  
\setkomafont{pageheadfoot}{\small}

### C.3. Linespread with setspace

- Load: \usepackage[onehalfspacing]{setspace} for 1.5 line spacing.

## D. Document Structure (see also L)

### D.1. Start Document

\begin{document} Complete document contents. \end{document}

### D.2. Title

simple title: \author{text} \title{text} \date{\today} \maketitle  
title page self designed: \begin{titlepage} text \end{titlepage}

D.3. Table of Contents, List of Figures (for other List of see E.8 & G)  
\tableofcontents \listoftables \listoffigures  
KOMAoption toc=listof (see B.1) generates entries for TOC.

### D.4. Headings

\part{title}	\chapter{title}	
\section{title}	\subsection{title}	\subsubsection{title}
\paragraph{title}	\subparagraph{title}	

→ \chapter only valid in documentclass scrbook and scrreprt  
→ Use \* variants for headings without numbering, no change in counter and no entry in table of contents.  
→ Use the optional parameter for short titles in headings and table of contents, e.g. \section[short title]{title}  
→ Use \addpart, \addchap or \addsec for unnumbered headings, but with running heading and entry in table of contents. The \* variants delete the running heading.  
→ Layout of paragraph and subparagraph similar to other headings:  
\RedeclareSectionCommands[after skip=1sp]{paragraph, subparagraph}  
\setcounter{secnumdepth}{\subparagraphnumdepth}  
\setcounter{tocdepth}{\subparagraphnumdepth}

### D.5. Justification

Environment	Declaration	Other
\begin{center}	\centering	text \par \vfill text
\begin{flushleft}	\raggedright	text \hfill text
\begin{flushright}	\raggedleft	\raggedbottom, \flushbottom

### D.6. Lists

\begin{itemize} with bullets \item or \item[symbol]  
\begin{enumerate} with numbers \item  
\begin{description} with bold words \item[word]  
\begin{labeling}[separator]{labelinglabel} \item[word]  
  
\begin{enumerate}  
  \item First item  
  \item Second item\label{it:second} % see References  
\end{enumerate}

### D.7. Enhanced Lists with enumitem

Load: \usepackage{enumitem}  
Example (for enumerate):  
\setlist[enumerate,1]{label=\Alph\*} A) one  
\setlist[enumerate,2]{label=\alpha\*} a) one  
\setlist[enumerate,3]{label=\roman\*} b) two  
\setlist[enumerate,4]{label=\arabic\*} B) two  
  
Example (for legal list):  
\newlist{legal}{enumerate}{10} 1. one  
\setlist[legal]{label=\arabic\*, .noitemsep} 1.1. two  
Use: \begin{legal} \item ... \end{legal} 1.1.1. three  
1.1.2. strawberry

### D.8. Separate Files

- After preamble within the text place: \include{file} Text starts and ends on a new page. file has to be in the same directory as the master file. Otherwise specify a path: \include{path/file}
- In preamble place: \includeonly{file1,file2} to run only these files.
- Use \input{file} includes a file without starting/ending on a new page (\includeonly not valid).

## E. Text

### E.1. Paragraphs (≈ "new idea in content")

Paragraphs are separated by an empty line in the code or by \par.  
A \ produces a new line – use sparingly, seldom needed outside tabulars.  
Correct Overfull Box Warnings with more than 4pt (look into log file).

### E.2. Text Symbols/Characters (see also A.2)

A lot of diacritic symbols can be typed directly, e.g. è é ê ñ ç  
§ \S \_ \textunderscore{ } ~ \textasciitilde{ }  
^ \textasciicircum{ } ... \ldots | \textbar  
Other symbols need packages, e.g. € \texteuro (textcomp)

### E.3. Fonts

Command	Declaration	Effect
\textrm{text}	\rmfamily text	Roman family
\textsf{text}	\sffamily text	Sans serif family
\texttt{text}	\ttfamily text	Typewriter family
\textmd{text}	\mdseries text	Medium series
\textbf{text}	\bfseries text	<b>Bold series</b>
\textup{text}	\upshape text	Upright shape
\textit{text}	\itshape text	<i>Italic shape</i>
\textsl{text}	\slshape text	<i>Slanted shape</i>
\textsc{text}	\scshape text	SMALL CAPS SHAPE

More general commands:  
\emph{text} \em text *Emphasized*  
\textnormal{text} \normalfont text Document font

Example: \setkomafont{section}{\scshape}

### E.4. Font Size

Font size is relative to the base font size, specified in the document class.

\tiny	<small>tiny</small>	\Large	Large
\scriptsize	<small>scriptsize</small>	\LARGE	LARGE
\footnotesize	<small>footnotesize</small>		
\small	<small>small</small>	\huge	huge
\normalsize	<small>normalsize</small>		
\large	<small>large</small>	\Huge	Huge

Use: \small text or \huge text\par to limit the size change.  
Example: \setkomafont{pageheadfoot}{\small}

### E.5. Colors with xcolor

```
\usepackage{xcolor}
\definecolor{DarkBlue}{RGB}{0, 115, 207}
\colorlet{col.section}{DarkBlue}
\textcolor{red}{text in red} or {\color{red}text}
\colorbox{gray!25}{\color{gray} faded by 25\%}
```

Predefined colors:  
white gray black red green blue cyan magenta yellow  
Fade a color with color!value between 0 and 100  
Headings in color: \setkomafont{disposition}{\color{color}}

### E.6. Footnotes

\footnote{text}	Print footnote marker in text and footnote at bottom of page
\footnotemark	Print footnote marker in text (e.g. within tabular or caption)
\footnotetext{text}	Print footnote at bottom of page

### E.7. References with hyperref (loads url implicitly)

\autocite{citekey}	Cite a bibliographic reference (package biblatex)
\label{marker}	Set a marker for cross reference, often if the form \label{sec:item} or \label{fig:diag1}
\autoref{marker}	Give type name and number of marker
\autopageref{marker}	Give abbreviation of "page" and page number of marker
\url{url}	Print clickable web page
\href[options]{url}{text}	Print clickable link
\hyperref[marker]{text}	Print clickable reference

Style: \urlstyle{xx} with xx a style like "tt", "rm", "sf" or "same".  
Names for autoref (package babel):  
\renewcaptionname{language}{\typenameautorefname}{text},  
e.g. \renewcaptionname{english}{\subsectionautorefname}{section}

### E.8. Acronyms with acro

```
\usepackage{acro,hyperref,longtable,tabu} \next 5 to praeambel
\acsetup{list-style=longtabu,list-heading=addchap}
\DeclareAcroListStyle{longtabu}{table}{table=longtabu,
table-spec=@{}{\bfseries}\X@{}}
\DeclareAcronym{ecm}{short=EM,long=Electro Machining}
...
\ac{EM} or \Ac{EM} for capitalized first letter
\printacronyms
```

## F. Figures & Tables (floating environments)

### F.1. Figures with graphicx

Load: \usepackage{graphicx}  
Use: \includegraphics[opt]{file} (png, jpg, pdf)  
With 'figure' the environment to place a graphic is meant. The figure caption is printed where the caption command is placed in the input. Extra vertical space is controlled by the KOMAoption captions (see B.1).  
Use: \begin{figure}[pos] ... \caption{...} \label{fig:x} \end{figure}  
Parameter: pos is a suggestion for placing, it can be ignored by T<sub>E</sub>X.  
Possible values are combinations of t (top), h (here), b (bottom), ! (try harder), p (separate page).  
Hint: Define a path to the graphic files (no blanks in folder names; no special characters in file names) \graphicspath{ {folder/} {folder/} ... }  

```
\graphicspath{ {img/} } %subfolder for images; set in preamble
\begin{figure}\centering
\includegraphics[width=.8\columnwidth]{pic.jpg}
\caption[Short title]{Long title}\label{fig:ff}
\end{figure}
```

→ Numbering throughout the whole document (scrbook) with package chngcntr: \counterwithout{figure}{chapter} (same for table)  
→ Figure name: \renewcaptionname{language}{\figurename}{text}  
\renewcaptionname{language}{\figureautorefname}{text}

## F.2. Subfigures with subcaption

Load: `\usepackage{subcaption}`

Use: `\begin{subfigure}[pos]{width} ... \end{subfigure}`

```
\begin{figure}[ht] \centering
\begin{subfigure}[t]{0.5\textwidth}
\centering \includegraphics[height=1.2in]{figure-a}
\caption{Subcaption 1}\label{fig:SubFig1}\end{subfigure}
\begin{subfigure}[t]{0.5\textwidth}
\centering \includegraphics[height=1.2in]{figure-b}
\caption{Subcaption 2}\label{fig:SubFig2}\end{subfigure}
\caption{Caption of complete figure}\label{fig:Fig1}
\end{figure}
```

## F.3. Tables width aligned material

With ‘table’ the environment to place aligned material is meant. The table caption is printed where the caption command is placed in the input. For positioning options see [F.1](#).

```
\KOMAOptions{captions=tableabove} % move to praeambel
\begin{table}[htbp] \centering
\caption{Table caption}\label{tab:exp}
\begin{tabular}{@{}l@{}}
\emph{Name} & \emph{Desc.}\\ \hline
tikz2pdf & Python script\\
LaTable & visual table editor
\end{tabular}
\end{table}
```

Use: `\begin{tabular}[c b t]{@{} l r c | p{unit}}{}`

Column separation: `@{\hspace{unit}}` or `\setlength{\tabcolsep}{unit}`

Row separation: `\[unit]` or `\renewcommand{\arraystretch}{unit}`

Partial lines: `\cline{2-3}` instead of `\hline`

Additional packages: array, longtable, booktabs, tabu, xcolor with option table, tabularx, tabulary

## F.4. Colored Table

```
\usepackage[table]{xcolor} % move to praeambel
\rowcolors{1}{}{lightblue} % {start row}{odd-row}{even-row}
\begin{tabular}{clr} ... \end{tabular}
```

## F.5. Suppress Floating with float

For a thesis most students want to control the placing of figures and tables themselves. One way is more control with placeins. Another way is to avoid the environments figure and table using \captionof. Quick and dirty is an additional positioning parameter using float:

Load: `\usepackage{float,scrhack}`

Use: `\begin{figure}[H]`, `\begin{table}[H]`

## F.6. Source Code Listings with listings

Load: `\usepackage{listings}`

Options: `\lstset{ basicstyle=\ttfamily\small, language=Python, numbers=left, keywordstyle=\color{blue}\bfseries }`

See option literate for Umlauts (`\literate=ä{\{"a"}1`)

Languages: C, C++, Java, Matlab, Python, HTML, XML, ...

Use: Environment: `\begin{lstlisting} code \end{lstlisting}`

In line: `\lstinline+code+ (same start- and end char)`

File: `\lstinputlisting{filename}`

```
1 @ Python selection
2 secret=42
3 guess=init(input("Enter number: "))
4 if guess==secret:
5     print ("You won!")
6 elif guess==secret:
7     print ("No, secret number bigger.")
8 else:
9     print ("No, secret number is smaller.")
```

## F.7. Boxes and Rules

Normal: `\parbox[pos][height][contentpos]{width}{text}` or

`\begin{minipage}[pos][height][contentpos]{width}text\end{minipage}`

Lift Text: `\raisebox[lift][height][depth]{text}`

Framed Box: `\fbox{text}` or `\framebox[width][pos]{text}`

Colored Box (xcolor): `\colorbox{backgroundcolor}{text}`

Framed Colored Box: `\colorbox{bordercolor}{backgroundcolor}{text}`

Resize (graphicx): `\scalebox{10}{giant}`

Lengths: `\setlength{\fboxsep}{unit}`, `\setlength{\fboxrule}{unit}`

## G. Bibliography with biblatex & External Processor biber

### G.1. Entry types

@article	@book	@inbook
@collection	@incollection	@manual
@misc	@online	@patent
@phdthesis	@proceedings	@periodical
@report	@techreport	@thesis

### G.2. Entry Fields (example see L)

author	title	journal	year	volume
editor	publisher	institution	school	series
pages	organization	number	note	key

### G.3. Styles

alphabetic authoryear authortitle numeric mla verbose

chem-acs phys nature science ieee apa

See [https://de.sharelatex.com/learn/Biblatex\\_bibliography\\_styles](https://de.sharelatex.com/learn/Biblatex_bibliography_styles)

### G.4. Example

```
% in preamble
\usepackage[autostyle=true]{csquotes} % Load
\usepackage[backend=biber,style=nature,language=british]{biblatex} % Load
\addbibresource{mybibliographyfile.bib} % Define
% anywhere within the document
\autocite{citekey} % Use
\printbibliography % Print
```

KOMAOption bibliography (see [B.1](#)) generates entry for TOC.

### G.5. External Processor

IDEs like T<sub>E</sub>Xstudio include the external processor, select biber as bibliography tool for ‘build’ in preferences, otherwise run biber explicitly.

## H. Math

### H.1. Math mode (Standard L<sup>A</sup>T<sub>E</sub>X)

Textstyle: `\(x^2 + 4\)`  $\rightsquigarrow$   $x^2 + 4$  as part of the text.

Displaystyle: `\[ x^2 + 4 \]`  $\rightsquigarrow$  separat line, centered

Equation: `\begin{equation} ... \end{equation}` \label{name}

$$\lambda := \lim_{x_1 \rightarrow \infty} \int_{x_0}^{x_1} \frac{f\left(\frac{t}{2}\right)}{\sqrt{t^2 + \sin^2(t)}} dt \stackrel{!}{\leq} 1 \quad (1)$$

→ Use \* variant for unnumbered equation (without label).

→ Package option for equation position: fleqn fixed indent from the left margin instead of centered.

→ Options for positions of equation number: leqno or reqno.

### H.2. Important Symbols in Math

+	+	-	-	±	pm	≠	\mp
<	<	≤	\le	≪	\ll	·	\cdot
>	>	≥	\ge	≫	\gg	×	\times
=	=	≠	\ne	≡	\equiv	≈	\approx
		⊥	\perp	∣	\mid	∥	\parallel
f′	f′	∇	\nabla	Δ	\Delta	∂	\partial
∈	\in	∀	\forall	∃	\exists	∄	\nexists
∩	\cap	∪	\cup	⊄	\notin		\setminus
ℓ	\ell	∠	\angle	○	\circ	∅	\emptyset
∨	\lor	∧	\land	¬	\neg	∅	\varnothing
⊤	\top	⊥	\bot	∞	\infty	∝	\propto

### H.3. Math Functions (upright typeface)

`\arccos \arcsin \arctan \arg \cos \cosh \cot \coth \csc \deg \det`  
`\dim \exp \gcd \hom \inf \ker \lg \lim \liminf \limsup`  
`\ln \log \lmax \lmin \lPr \sec \sin \sinh \sup \tan \tanh`

For other functions use (package amsmath): `\operatorname{name}`, e.g. `\operatorname{arcsinh}` (see also [J.2](#)).

### H.4. More Math Functions

$\sum$	<code>\sum</code>	$\prod$	<code>\prod</code>	$\coprod$	<code>\coprod</code>
$\int$	<code>\int</code>	$\iint$	<code>\iint</code>	$\iiint$	<code>\iiint</code>
$\vec{a}$	<code>\vec{a}</code>	$\dot{a}$	<code>\dot{a}</code>	$\ddot{a}$	<code>\ddot{a}</code>
				$\oint$	<code>\oint</code>

### H.5. Fonts and Sizes in Math Mode (some from $\mathcal{M}$ Math)

`\mathrm{}`, `\mathit{}`, `\mathbf{}`, `\mathsf{}`, `\mathtt{}`, `\boldmath{}`  
`\mathbb{}` e.g.  $\mathbb{Z}$ ,  $\mathcal{M}$  e.g.  $\mathcal{Z}$ ,  $\mathfrak{M}$  e.g.  $\mathfrak{M}$   
`\displaystyle`, `\scriptstyle`, `\scriptscriptstyle`, `\textstyle`  
`\boldsymbol{}`

### H.6. Often used Math Expressions

$x^{n+1}$	$x^{n+1}$	$E_{\text{kin}}$	$E_{\text{kin}}$
$\frac{a+b}{2}$	<code>\frac{a+b}{2}</code>	$\sqrt[n]{a^2+b^2}$	<code>\sqrt[n]{a^2+b^2}</code>
$x_1, \dots, x_n$	<code>x_1, \ldots, x_n</code>	$x_1 + \dots + x_n$	<code>x_1 + \cdots + x_n</code>

$\left(a + \frac{1}{2}\right)^2$  `\left(a + \frac{1}{2}\right)^2`  
 $\sum_{i=1}^N \prod_{i=1}^N$  `\sum_{i=1}^N \prod_{i=1}^N`  
 $\lim_{a \rightarrow \infty}$  `\lim_{a \rightarrow \infty}`  
 $\int_b^a x^2 dx$  `\int_a^b x^2 dx`  
 $\left.\frac{df}{dx}\right|_{x_0}$  `\left.\frac{df}{dx}\right|_{x_0}`  
 $\vec{F}_\perp$  `\myvec{F}_{\perp}`  
 $\vec{A}^\top$  `\myvec{A}^{\top}`  
 $\stackrel{above}{mid}$  `\overset{above}{mid}`  
 $\stackrel{mid}{below}$  `\underset{below}{mid}`  
 $\mathbf{1}_{\text{newcommand}\{\myvec\}[1]{\ensuremath{\underline{\boldsymbol{\#1}}}}}$

### H.7. Math with amsmath (replacing standard Environments)

`equation` `equation*` One line, one equation  
`multline` `multline*` One unaligned multiple-line equation, one number  
`gather` `gather*` Several equations without alignment  
`align` `align*` Several equations with multiple alignments  
`alignat` `alignat*` Multiple alignments, choose spacing between cols  
`flalign` `flalign*` Several equations: horizontally spread form of align  
`cases` Alignment for cases  
`split` A simple alignment within a multiple-line equation  
`aligned` A “mini-page” with multiple alignments  
`gathered` A “mini-page” with unaligned equations

→ The content is automatically placed in math mode.

→ Use `\intertext{text}` to set text within an amsmath environment

→ Length parameter to influence vertical spacing within any amsmath environment: `\jot` (e.g. `\addtolength{\jot}{1ex}`)

→ Add singular vertical space for a line via `\[<amount>]` (see [A.1](#))

→ Use the `spreadlines` environment from the `mathtools` package

→ Length parameters (with standard values) to influence vertical white space around displayed math formulas: `\abovedisplayskip=12pt`, `\belowdisplayskip=12pt`, `\abovedisplayskip=0pt`, `\belowdisplayskip=0pt`

### H.7.1. $\mathcal{M}$ Math align

$y = d$	$y = d$	$z = 1$
$y = d$	$y = cx + d$	
$y = bx^2 + cx + d$	$y = bx^2 + cx + d$	

$y = d$	$z = 1$
$y(x) = cx + d$	$z = x + 1$
$y_{12} = bx^2 + cx$	$z = x^2 + x + 1$

### H.7.2. $\mathcal{M}$ Math alignat

`\begin{alignat}{3} % 2x3-1 ' & ' are necessary`  
`i_{11} = 0.25 & i_{12} = i_{21} & i_{13} = i_{23}`  
`i_{21} = \frac{1}{3} i_{11} & i_{22} = 0.5 i_{12} & i_{23} = i_{31}`  
`i_{31} = 0.33 i_{22} & i_{32} = 0.15 i_{32} & i_{33} = i_{11}`  
`\end{alignat}`

### H.7.3. $\mathcal{M}$ Math flalign

`\begin{flalign+}`  
`i_{11} = 0.25 & i_{12} = i_{21} & i_{13} = i_{23}`  
`i_{21} = \frac{1}{3} i_{11} & i_{22} = 0.5 i_{12} & i_{23} = i_{31} \cdot \sqrt{5}`  
`i_{31} = 0.33 i_{22} & i_{32} = 0.15 i_{32} & i_{33} = i_{11}`  
`\end{flalign+}`

### H.7.4. $\mathcal{M}$ Math gather

`\begin{gather}`  
`D(a,r) \equiv \{z \in C : |z-a| < r\}`  
`seg(a,r) \equiv \{z \in C : \exists z \in \mathbb{C}, |z-a| < r\}`  
`C(E, \theta, r) \equiv \bigcup_{e \in E} C(e, \theta, r)`  
`\end{gather}`

### H.7.5. $\mathcal{M}$ Math matrix

`\begin{matrix}` `a & b \\ c & d` `\end{matrix}`  
`\begin{pmatrix}` `a & b \\ c & d` `\end{pmatrix}`  
`\begin{bmatrix}` `a & b \\ c & d` `\end{bmatrix}`  
`\begin{Bmatrix}` `a & b \\ c & d` `\end{Bmatrix}`  
`\begin{vmatrix}` `a & b \\ c & d` `\end{vmatrix}`  
`\begin{Vmatrix}` `a & b \\ c & d` `\end{Vmatrix}`  
$$\begin{matrix} a & b & \begin{pmatrix} a & b \\ c & d \end{pmatrix} & \begin{bmatrix} a & b \\ c & d \end{bmatrix} & \begin{Bmatrix} a & b \\ c & d \end{Bmatrix} & \begin{vmatrix} a & b \\ c & d \end{vmatrix} & \begin{Vmatrix} a & b \\ c & d \end{Vmatrix} \end{matrix}$$
  
`Dots: \dots or \ldots lower dots, \cdots vertically centered dots, \vdots vertical dots, \ddots diagonal dots, \hdotsfor{cols} \{dotspace\} multicolumn dots.`

### H.8. $\mathcal{M}$ Math cases


`\[ f(n) = \begin{cases} n/2 & \text{if } n \text{ is even} \\ -(n+1)/2 & \text{if } n \text{ is odd} \end{cases}`

### H.9. Arrows

$\mapsto$	<code>\mapsto</code>	$\rightsquigarrow$	<code>\leadsto</code>
$\rightarrow$	<code>\rightarrow</code>	$\Rightarrow$	<code>\Rightarrow</code>
$\longrightarrow$	<code>\longrightarrow</code>	$\Longrightarrow$	<code>\Longrightarrow</code>
$\leftarrow$	<code>\leftarrow</code>	$\Leftarrow$	<code>\Leftarrow</code>
$\longleftarrow$	<code>\longleftarrow</code>	$\Longleftarrow$	<code>\Longleftarrow</code>
$\Uparrow$	<code>\Uparrow</code>	$\Uparrow$	<code>\Uparrow</code>
$\Downarrow$	<code>\Downarrow</code>	$\Downarrow$	<code>\Downarrow</code>
$\leftrightharpoons$	<code>\leftrightharpoons</code>	$\rightleftharpoons$	<code>\rightleftharpoons</code>
$\rightleftharpoons$	<code>\rightleftharpoons</code>	$\rightleftharpoons$	<code>\rightleftharpoons</code>
$\rightleftharpoons$	<code>\rightleftharpoons</code>	$\rightleftharpoons$	<code>\rightleftharpoons</code>





<div></div> <div><div>My Title</div><div>My Subtitle</div><div>N.N.</div></div>	<div>Contents</div> <div><div>List of Figures</div><div>List of Tables</div><div>1. Intro</div><div>1.1. Technical introduction</div><div>1.1.1. Heading lower level</div><div>2. Method</div><div>A. Appendix</div><div>Bibliography</div></div> <div><div>iii</div><div>iv</div><div>1</div><div>1</div><div>1</div><div>2</div><div>4</div><div>5</div></div>	<div>List of Figures</div> <div><div>A.1. Example of a caption centeredbeside</div></div> <div><div>4</div></div>	<div>List of Tables</div> <div><div>1.1. Table caption</div></div> <div><div>1</div></div>	<div>1. Introduction</div> <div><div>1.1. Technical introduction</div></div> <div>Heading without number</div> <div>Some text in <b>bold</b>, some more text. Some text in <i>emph</i>, more text. And now some text in German language Hier kommt eine Formel: <math>2 + 2 = 5</math></div> <div>A new paragraph, viz. a new idea, a new thought.</div> <div><div>1.1.1. Heading lower level</div></div> <div>More text <i>emphasized</i> text.<sup>1</sup></div> <div>Table 1.1.: Table caption</div> <div><table><tr><th>Head</th><th>Head</th><th>Head</th></tr><tr><td>Data</td><td>first</td><td>Row</td></tr><tr><td>Data</td><td>second</td><td>Row</td></tr></table></div> <div><div><sup>1</sup>Wikipedia. TeX- Wikipedia, The Free Encyclopedia. <a href="https://en.wikipedia.org/wiki/TeX">https://en.wikipedia.org/wiki/TeX</a>, last referenced 24-June-2017.</div></div> <div><div>1</div></div>	Head	Head	Head	Data	first	Row	Data	second	Row
Head	Head	Head											
Data	first	Row											
Data	second	Row											
<div>2. Method</div> <div>Some text, see Table 1.1 for aligned material.</div> <div>L'Hopital's rule:</div> <div>(2.1) <math>\lim_{x \rightarrow 0} \frac{e^x - 1}{2x} = \lim_{x \rightarrow 0} \frac{\frac{1}{e^x}}{2} = \frac{1}{2}</math></div> <div><div>2</div></div>	<div>2. Method</div> <div>More text. A "quote" and a „Zitat“.</div> <div><div>3</div></div>	<div>A. Appendix</div> <div><div>A BOX</div></div> <div>Figure A.1.: Example of a caption beside the figure</div> <div><div>4</div></div>	<div>Bibliography</div> <div>Wikipedia. TeX- Wikipedia, The Free Encyclopedia. <a href="https://en.wikipedia.org/wiki/TeX">https://en.wikipedia.org/wiki/TeX</a>, last referenced 24-June-2017.</div> <div><div>5</div></div>	<div>License</div> <div>Reference Sheet, Version 1.2, August 2017</div> <div>This Cheat Sheet is free under the terms and conditions of the L<sup>A</sup>T<sub>E</sub>X Project Public License Version 1.3c.</div> <div>Authors: Marion Lammarsch, University of Heidelberg, and Elke Schubert, Stutenrose</div> <div>Credit for the layout and content ideas goes to</div> <div>(a) Emanuel Regnath, "L<sup>A</sup>T<sub>E</sub>X Cheat Sheet" L<sup>A</sup>T<sub>E</sub>X 4E1, 2013 <a href="http://www.latex4ei.de/downloads/LaTeX_CheatSheet.pdf">http://www.latex4ei.de/downloads/LaTeX_CheatSheet.pdf</a></div> <div>(b) Tanno Schreiwel, DHBW Mosbach, "LaTeX Cheat German", Ed. 2014 <a href="http://mirror.ctan.org/tex-archive/info/latexcheat/latexcheat-de/">http://mirror.ctan.org/tex-archive/info/latexcheat/latexcheat-de/</a></div> <div>The logo with the TeX lion is sketched by Dwayne Bibbey, <a href="https://mirror.ctan.org/1306/t11ex">https://mirror.ctan.org/1306/t11ex</a>; the graphic of a header of a page is from the KOMA-Script documentation, file scrlayer-scrpage.tex; the page layout image is done by using the package layout and the command \layout and the red lines and red text added with a graphics program.</div> <div>Acknowledgements</div> <div>We would like to thank Markus Kohn, Ulrike Fischer, Martin Sievers, and Herbert Voß for their useful hints and suggestions.</div> <div>Not naming all the female and male students which with their intrigued questions helped to improve this reference sheet.</div>									





# Literaturverzeichnis

- [Haf18] HAFNER, E.G.: *Lineare Algebra für Dummies*. Wiley, 2018 (Für Dummies). <https://books.google.de/books?id=XfWADwAAQBAJ>. – ISBN 9783527819430
- [ÖÖ15] ÖCHSNER MARCO ; ÖCHSNER ANDREAS: *Das Textverarbeitungssystem LaTeX*. 5. Auflage. Griffith University, Gold Coast, Australia : Springer Vieweg, 2015 (essentials). <https://link.springer.com/book/10.1007/978-3-658-09503-1>. – ISBN 978-3-658-09503-1



# Abbildungsverzeichnis

2.1. Bilddateien in L <sup>A</sup> T <sub>E</sub> X einbinden . . . . .	5
2.2. 42 . . . . .	5
2.3. L <sup>A</sup> T <sub>E</sub> X-Code zur Verwendung von <i>wrapfigure</i> . . . . .	6
2.4. L <sup>A</sup> T <sub>E</sub> X-Code von Tabelle 2.2 . . . . .	7
2.5. L <sup>A</sup> T <sub>E</sub> X-Code von Tabelle 2.1, einer <i>longtable</i> . . . . .	8
2.6. L <sup>A</sup> T <sub>E</sub> X-Code des Code-Ausschnitts 2.1, einem <i>lstlisting</i> . . . . .	8
2.7. L <sup>A</sup> T <sub>E</sub> X-Code des Code-Ausschnitts 2.2 . . . . .	9
2.8. L <sup>A</sup> T <sub>E</sub> X-Code der Auflistung der Untersektion 2.2.7 . . . . .	10
2.9. L <sup>A</sup> T <sub>E</sub> X-Code der Auflistung der Untersektion 2.2.3 . . . . .	11
2.10. verschiedene Stile des Literaturverzeichnisses . . . . .	12
2.11. L <sup>A</sup> T <sub>E</sub> X-Code der drei Gleichungen 2.1, 2.2 und 2.3 . . . . .	13
2.12. L <sup>A</sup> T <sub>E</sub> X-Code der <i>align</i> -Gleichungen . . . . .	14
2.13. L <sup>A</sup> T <sub>E</sub> X-Code der Gleichungen 2.9 und 2.10 . . . . .	15
3.1. Alfresco-Dokumentenbibliothek . . . . .	17



# Tabellenverzeichnis

2.1.	Parameter zu Positionierung von <i>floats</i> . . . . .	6
2.2.	test-table . . . . .	7
2.3.	Dokumentationen zu allen verwendeten <i>packages</i> . . . . .	15



# Code-Auszugs-Verzeichnis

2.1.	simpler C++ Programmausschnitt . . . . .	8
2.2.	Python-Code als <i>lstinputlisting</i> . . . . .	9

