
Titel der Doku

Art der Doku

erstellt am: 4. Oktober 2021

Department for Computer Science
Eidgenössische Technische Hochschule Zürich

Autor:	Peter Kessler
Matrikelnummer:	1234567
eMail:	peter.kessler@id.ethz.ch
Institut:	Institute for Computing Platforms
Professur:	Data Sciences, Data Systems, and Data Services
Professor:	Prof. Ce Zhang
Studienjahrgang:	2020
Betreuer:	Hans Mustermann
Ko-Examinator:	Bodo Bitbeisser

Abstract / Summary

Dieses Dokument (erstellt mit \LaTeX - siehe Kapitel: Installation der LaTeX-Umgebung auf Seite 1), soll die Erstellung von umfangreichen technischen oder wissenschaftlichen Dokumentationen erleichtern.

Are you ready to leave those "what you see is what you get" word processors behind and to enter the world of real, reliable, and high-quality typesetting[1]?

Ob in Uni, Beruf oder Alltag: Muss eine umfangreiche technische oder wissenschaftlich Dokumentation erstellt werden, führt über kurz oder lang kein Weg an \LaTeX vorbei.

Wofür eignet sich \LaTeX ?

- jede Art wissenschaftlicher Veröffentlichungen
- Bücher (Sachbücher, Romane, Lexika, ...)
- Lebensläufe, Serienbriefe, Vorträge und Poster
- ...

Wofür eignet sich \LaTeX nicht!

- Zeitungssatz
- Desktop Publishing (Plakate, Flyer etc.)
- sehr kurze Texte
- alle Bereiche, in denen Seiten-Elemente völlig frei angeordnet werden sollen
- ...

\LaTeX verführt immer wieder viele Anwender – insbesondere Anfänger – dazu, irgendetwas noch schöner zu machen: 'fancy' Schriften zu verwenden, am Layout herumzubasteln, Bilder punktgenau auf einer Seite zu platzieren, etc. Tatsächlich sind die Möglichkeiten nahezu unbegrenzt. Aber: Layout und Schriften sollen nicht schön sein, sondern den Inhalt der Arbeit aus Sicht des Lesers optimal transportieren.

\LaTeX ist somit nicht geeignet für jemanden, der sagt: 'Ich weiss selber am besten, wie mein Dokument aussehen soll.'. Man kann zwar mit \LaTeX im Prinzip alles machen, aber man muss entweder Glück haben und jemand anders hat schon ein entsprechendes Paket geschrieben, das man verwenden kann, oder man muss sehr viel Wissen über \TeX haben.

\LaTeX ist dagegen sehr gut geeignet für alle die sagen 'Ich möchte mich nur um den Inhalt, aber nicht um die Formatierung kümmern. Es soll aber trotzdem gut aussehen.'.

Widmung / Danksagung

Der Text für die Widmung resp. die Danksagung kann an dieser Stelle eingefügt werden.

Inhaltsverzeichnis

Abstract / Summary	i
Widmung / Danksagung	ii
Inhaltsverzeichnis	iii
Abbildungsverzeichnis	v
Tabellenverzeichnis	vi
Abkürzungsverzeichnis	vii
Symbolverzeichnis	viii
 1. Installation und Konfiguration der LaTeX-Umgebung	 1
1.1. Installation der LaTeX-Umgebung	1
1.2. Konfiguration der LaTeX-Umgebung	3
 2. Trennung von Inhalt und Struktur	 4
 3. Formale Vorgaben abholen / beachten	 5
 4. Text erfassen	 6
4.1. Text formatieren	7
4.1.1. Gliederung / Strukturierung	7
4.1.2. Absätze	8
4.1.3. Schriftarten und Farben	9
4.1.4. Aufzählungen & Bullet-Points	9
 5. Arbeiten mit Tabellen	 11
5.1. Tabellen ohne Schraffierung	11
5.2. Tabellen mit Schraffierung	13
 6. Arbeiten mit Bildern / Grafiken	 15
 7. Arbeiten mit Abkürzungen	 16
7.1. Erfassen von Abkürzungen	16
7.2. Einsetzen von Abkürzungen	17
7.3. Verwenden von Abkürzungen	17
 8. Arbeiten mit Fussnoten	 18
8.1. Erfassen von Fussnoten	18
 9. Arbeiten mit Stichworten	 19
9.1. Erfassen von Stichworten	19

10.Arbeiten mit Quellenangaben	20
10.1. Erfassen von Quellen	20
10.2. Verwenden von Quellen	21
10.3. Quellenverzeichnis erstellen	21
Ergänzende Materialien / Anhang	ix
Literatur- / Quellenverzeichnis	x
Stichwort- / Namensverzeichnis	xi
Fussnoten	xii
Eidesstattliche Erklärung	xiii

Abbildungsverzeichnis

Abb. 1:	Arbeiten mit \LaTeX	1
Abb. 2:	Konfiguration Latex: Befehle	3
Abb. 3:	Konfiguration Latex: Schnelles Übersetzen	3
Abb. 4:	Die Verzeichnis-Struktur des Dokumentes	4
Abb. 5:	Kapitel-Struktur des Dokumentes	6
Abb. 6:	Absätze im Dokument	8
Abb. 7:	Schriftarten und Farben	9
Abb. 8:	Eigene Aufzählungszeichen	9
Abb. 9:	Eigene Aufzählungszeichen	10
Abb. 10:	Eine erste LaTeX-Tabelle	11
Abb. 11:	Eine zweite LaTeX-Tabelle	12
Abb. 12:	LaTeX-Tabellen Präambel	12
Abb. 13:	Eine dritte LaTeX-Tabelle	13
Abb. 14:	Eine vierte LaTeX-Tabelle	13
Abb. 15:	Eine fünfte LaTeX-Tabelle	14
Abb. 16:	Eine sechste LaTeX-Tabelle	14
Abb. 17:	\LaTeX -Löwe ohne Rahmen	15
Abb. 18:	\LaTeX -Löwe mit Rahmen	15
Abb. 19:	Der \LaTeX -Code zu den beiden \LaTeX -Löwen	15
Abb. 20:	Erfassen der Quellen	20
Abb. 21:	Auflistung der Quellen im Literatur- / Quellenverzeichnis	21
Abb. 22:	Erstellen des Literatur- / Quellenverzeichnis	21

Tabellenverzeichnis

Tab. 1: Eine erste LaTeX-Tabelle	11
Tab. 2: Eine zweite LaTeX-Tabelle	12
Tab. 3: Eine dritte LaTeX-Tabelle	12
Tab. 4: Eine vierte LaTeX-Tabelle	13
Tab. 5: Eine fünfte LaTeX-Tabelle	14
Tab. 6: Eine sechste LaTeX-Tabelle	14

Abkürzungsverzeichnis

ETH	Eidgenössische Technische Hochschule
ETHZ	Eidgenössische Technische Hochschule Zürich
EPFL	École polytechnique fédérale de Lausanne
ID	Informatikdienste
PM	Portfolio Management
PPF	Procurement and Portfolio Management

Symbolverzeichnis

\mathbb{N}	Menge aller natürlichen Zahlen ohne die Null
\mathbb{N}_0	Menge aller natürlichen Zahlen mit der Null
π	Die Kreiszahl Pi
Ω	Der elektrische Widerstand Ohm
α	Alpha, der erste Buchstabe des griechischen Alphabetes

1. Installation und Konfiguration der LaTeX-Umgebung

1.1. Installation der LaTeX-Umgebung

LaTeX (sprich: Latech) ist ein mächtiges Werkzeug zum Setzen von Schriftstücken. Gerade dann, wenn die Texte etwas länger werden, spielt es seinen entscheidenden Vorteil aus: Anders als andere Textverarbeitungssysteme stürzt es nie ab, auch bei mehr als 1000-seitigen Büchern nicht. Einer der weiteren grossen Vorteile von LaTeX ist es, dass es sich nach der Logik des Dokumentes richtet. LaTeX sagt man nicht "dieser Text hier soll etwas grösser und fett sein" sondern "dies hier ist eine Überschrift". Alles andere erledigt LaTeX von selbst. LaTeX verarbeitet den Text teilweise vollautomatisch und nach einigen wenigen Anfangskonfigurationen kann man sich voll und ganz auf den Inhalt konzentrieren, statt sich mit der Formatierungen herumzuschlagen. Hinzu kommt, dass in LaTeX ein sehr leistungsfähiger, einfach zu bedienender Formeleditor eingebaut ist, was LaTeX vor allen Dingen für wissenschaftliche Arbeiten interessant macht. LaTeX wurde ausserdem mit dem Ziel geschrieben, dass Schriftstücke auch in 100 Jahren noch gleich aussehen und nicht so wie bei Textverarbeitungssystemen mal die Zeile auf die nächste Seite rutscht, mal die Bilder verrutschen oder gar nicht mehr da sind. LaTeX-Dokumente zu "programmieren" ist vergleichbar mit Internetseiten mit der Seitenbeschreibungssprache HTML zu programmieren. Es ist ähnlich einfach wie HTML und auch ähnlich aufgebaut. Jedoch ist es nicht für Internetseiten programmiert, sondern für beliebige Papierdokumente. Last but not least ist ein ausschlaggebendes Argument für LaTeX, dass es völlig kostenlos ist. Leider bringt LaTeX auch einen grossen Nachteil mit sich, den ich hier nicht verschweigen will. LaTeX muss man sozusagen "programmieren".



Abb. 1: Arbeiten mit \LaTeX

Einige Befehle sind zu erlernen. Wenn man einen gewissen Befehlssatz auswendig kann, dann gewinnt dieses "Programmieren" jedoch einen riesigen Geschwindigkeitszuwachs,

so dass "normale" Textverarbeitungssysteme nicht mehr mithalten können. LaTeX ist auch nicht für hochgradige Designer-Texte geeignet. Der Designer kümmert sich um das Aussehen der Seite. Gerade dies ist bei LaTeX dem LaTeX-Übersetzer überlassen.

LaTeX-Dokumente werden im Allgemeinen mittels einer Entwicklungsumgebung erstellt. Zwar kann man LaTeX-Dokumente auch mit Hilfe eines einfachen Texteditors erfassen und auf der Kommandozeile das Dokument erstellen. Doch bieten die auf LaTeX angepassten Programme mehr Funktionen und Komfort. Viele LaTeX-Befehle, Sonderzeichen und Symbole sind über die grafische Benutzeroberfläche zugänglich, und teilweise lassen sich darüber auch einfache Tabellen erstellen. Für grosse Projekte bieten Entwicklungsumgebungen eine Verwaltung und Strukturdarstellung. Dokumentenvorlagen und PDF-Vorschau finden sich fast überall. Zur Verbesserung der Lesbarkeit des Codes gibt es eine Syntax-Hervorhebung und teilweise auch eine Autovervollständigung von Befehlen. Manche Programme bieten eine Rechtschreibprüfung. Umlaute werden von manchen Entwicklungsumgebungen automatisch in LaTeX-Befehle übersetzt. Der fortschrittlichere Ansatz zur Verwendung von Umlauten ist eine geeignete Zeichenkodierung und ein dazu passendes LaTeX-Paket. Manche Umgebungen unterstützen insbesondere Unicode.

Da LaTeX ein freies Produkt ist, welches auch im Sourcecode verfügbar ist, gibt es viele unterschiedliche sogenannte Distributionen. Das System wurde ehemals für den Unixbereich entwickelt, wurde aber auf sehr viele Systeme übertragen.

MikTeX ist eine sehr gute LaTeX-Distribution für Windows. Während der Installation wird nachgefragt, wie mit fehlenden Packages umgegangen wird. Es ist empfohlen, diese mit Nachfrage installieren zu lassen, da bei der erstmaligen Übersetzung dieses Dokumentes werden so einige Packages nach-installiert. Diese sind jedoch für ein korrektes Layout und für die korrekte Funktionsweise notwendig. Der Installer dieser Distribution ist erhältlich unter <https://miktex.org/>

Texmaker ist ein plattformübergreifender Unicode-Texteditor für die Erstellung von LaTeX-Dokumenten. Die Software wird unter der GNU General Public License veröffentlicht. Texmaker ist sehr gut bedienbar und übersichtlich. Ähnlich zu den anderen bekannten LaTeX - Editoren bietet auch er die grundlegende Unterstützung beim Einfügen von LaTeX-Konstruktionselementen - jedoch noch einige weitere Besonderheiten. Der Editor richtet sich insbesondere an LaTeX-Anfänger, denen mit Hilfe von Assistenten die Erstellung von Dokumenten erleichtert werden soll. Details und weiteres findet man unter <http://www.xmlmath.net/texmaker/>

Diese Dokumentation (dieses Template) ist mit **MikTeX** und **Texmaker** auf Windows 10 erstellt worden.

1.2. Konfiguration der LaTeX-Umgebung

LaTeX nimmt die *.tex Datei(en) und erstellt damit die eigentliche Druck-Datei (z.B. *.pdf). Dieser Vorgang ist nicht trivial. Abhängig von den gewünschten Ergebnissen resp. der zu erstellenden Verzeichnisse, muss der Übersetzungsvorgang mehrfach ausgeführt werden damit letztendlich das Dokument wie gewünscht erstellt wird.

Texmaker ist via **Optionen** > **Texmaker konfigurieren** wie folgt zu konfigurieren:

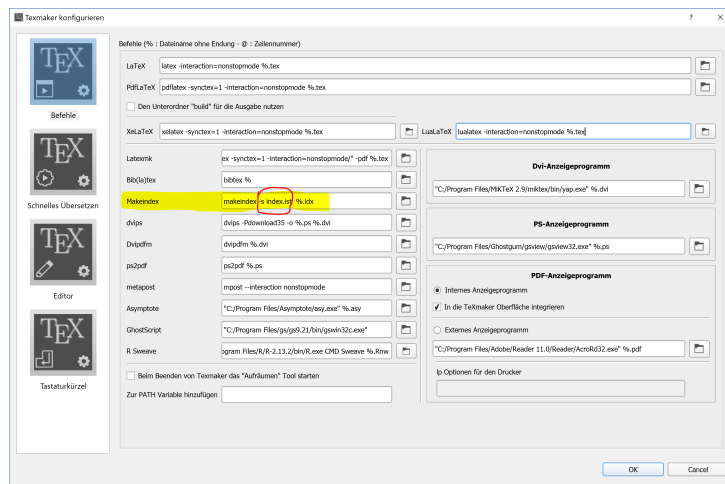


Abb. 2: Konfiguration Latex: Befehle

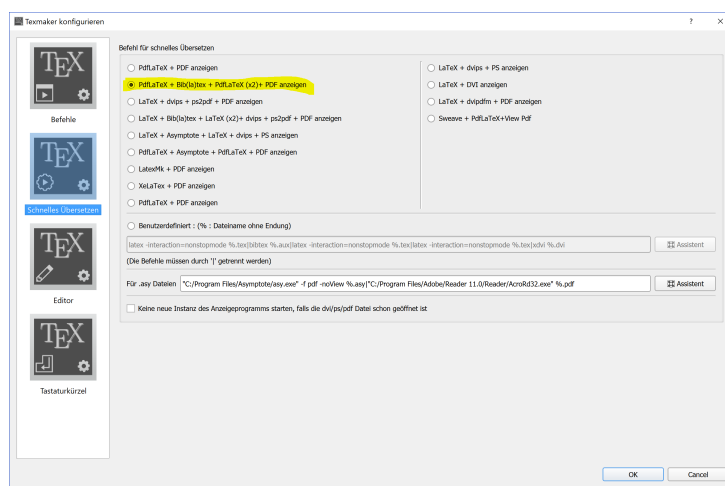


Abb. 3: Konfiguration Latex: Schnelles Übersetzen

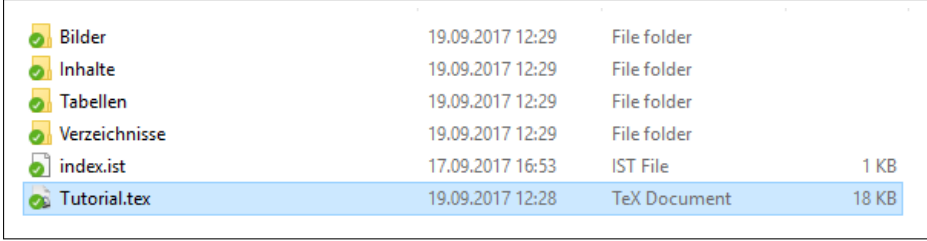
Zusätzlich muss das Stichwortverzeichnis mit **F12** manuell erstellt werden.

Empfehlung:

Mit der Tasten-Folge **F1** **F12** **F1** wird alles korrekt erstellt.

2. Trennung von Inhalt und Struktur

Dieses L^AT_EX Tutorial / Template ist so aufgebaut, dass die Struktur-Informationen des zu erstellenden Dokumentes in der Haupt-Datei *.tex abgelegt ist. Alle Inhalte des Dokumentes sind in verschiedenen Verzeichnissen in weiteren Dateien enthalten.



Bilder	19.09.2017 12:29	File folder	
Inhalte	19.09.2017 12:29	File folder	
Tabellen	19.09.2017 12:29	File folder	
Verzeichnisse	19.09.2017 12:29	File folder	
index.ist	17.09.2017 16:53	IST File	1 KB
Tutorial.tex	19.09.2017 12:28	TeX Document	18 KB

Abb. 4: Die Verzeichnis-Struktur des Dokumentes

Die Datei `Tutorial.tex` ist das Kerndokument. Diese Datei enthält die Struktur (Reihenfolge der Kapitel) des Dokumentes, sowie alle benötigten Befehle für die Formatierung und für die Erstellung der verschiedenen zu generierenden Verzeichnisse (wie z.B. das Inhaltsverzeichnis, das Stichwortverzeichnis, etc.).

Die Datei `index.ist` enthält die Style-Informationen zum Stichwortverzeichnis. Dies wäre nicht zwingend notwendig. Da mir aber das Layout des von LaTeX automatisch erstellten Stichwortverzeichnisses nicht gefällt, habe ich dieses mit dieser Style-Datei angepasst. Dies muss in der Konfiguration von Texmaker entsprechend angegeben werden (siehe auch Abb. 2: Konfiguration Latex: Befehle auf Seite 3).

Sämtliche (fachlichen) Inhalte des Dokumentes werden in separaten Dokumenten innerhalb der jeweiligen Verzeichnisse (Bilder, Inhalte, Tabellen, Verzeichnisse) abgelegt. Diese werden als einfache Text-Dateien mit Endung `*.tex` erfasst.

Auf den folgenden Seiten dieses Tutorials wird gezeigt, wie Text erfasst und in Absätze gegliedert wird, mit welchen Befehlen man Bilder und Grafiken einbinden kann, wie Tabellen erstellt werden, wie Querverweise innerhalb des Dokumentes erstellt werden, wie korrekt mit Abkürzungen gearbeitet wird, wie Begriffe in das Stichwortverzeichnis aufgenommen werden und wie Fussnoten sowie Literatur- / Quellenangaben erfasst und gesetzt werden.

Dieses Tutorial / Template selbst verwendet alle im Inhaltsverzeichnis aufgeführten Punkte & Inhalte. Damit kann der Source-Text dieses Tutorials / Template `Tutorial.tex` ebenfalls als Quelle für die Beantwortung vieler Fragen genommen werden.

3. Formale Vorgaben abholen / beachten

Arbeiten heisst immer auch dokumentieren. Projekt-, Produkt-, System-, Benutzer-Dokumentationen oder spätestens beim Abschluss der Berufslehre die IPA - das Erstellen von Dokumentationen ist (Berufs-) Alltag. Deshalb erhältst Du im Lehlabor verschiedentlich die Aufgabe, eine schriftliche (technische, wissenschaftliche) Dokumentation zu verfassen.

Bevor nun mit der eigentlichen (fachlichen) Dokumentation begonnen wird:

Vergewissere Dich unbedingt vor dem Beginn Deiner Arbeit bei Deiner Betreuungsperson, ob sich die äussere Form (Formatierung, Layout, Reihenfolge der Kapitel, etc.) dieses Dokumentes mit den Vorgaben deckt, bzw. hole unbedingt bei Deiner Betreuungsperson ab, an welche Richtlinien / Vorgaben Du Dich für Deine Dokumentation halten musst.

Das Template ist so aufgebaut, dass Anpassungen vorgenommen werden können, ohne sehr tiefe L^AT_EX-Kenntnisse zu benötigen. Die Verwendung dieses Templates sollte somit keine Kopfschmerzen verursachen.

4. Text erfassen

Dieses Tutorial / Template ist so aufgebaut, dass in der Datei `Tutorial.tex` alle Struktur-Informationen des Dokumentes enthalten sind. Im wesentlichen sind dies die Reihenfolge aller Kapitel (section) und Unterkapitel (subsection).

```
%
% ===== Kapitel 1 =====
%
\pagebreak
\section{Installation und Konfiguration der LaTeX-Umgebung}
%
%
\subsection{Installation der LaTeX-Umgebung} \label{Installation}
\begin{spacing}{1.25}
\input{./Inhalte/Installation.tex}
\end{spacing}
%
%
\subsection{Konfiguration der LaTeX-Umgebung} \label{Konfiguration}
\begin{spacing}{1.25}
\input{./Inhalte/Konfiguration.tex}
\end{spacing}
%
%
%
% ===== Kapitel 2 =====
%
\pagebreak
\section{Trennung von Inhalt und Struktur}
\begin{spacing}{1.25}
\input{./Inhalte/InhaltStruktur.tex}
\end{spacing}
%
%
%
% ===== Kapitel 3 =====
%
\pagebreak
\section{Text erfassen}
\begin{spacing}{1.25}
\input{./Inhalte/TextErfassen.tex}
\end{spacing}
%
```

Abb. 5: Kapitel-Struktur des Dokumentes

Die eigentlichen fachlichen (Text-) Inhalte der Kapitel sind in separaten `*.tex` Dateien im Verzeichnis `Inhalte` erfasst, die mit dem Befehl `\input{...}` eingebunden / eingefügt werden (siehe Abb. 5: Kapitel-Struktur des Dokumentes auf Seite 6).

4.1. Text formatieren

An dieser Stelle sollen exemplarische einige Beispiele gezeigt werden, wie in L^AT_EX Texte **Fett**, **farbig**, *Kursiv*, unterstrichen, doppelt unterstrichen, unterschlängelt, ~~horizontal durchgestrichen~~, ~~schräg/durchgestrichen~~ gesetzt werden und wie Texte formatiert / strukturiert werden können.

Ebenso soll gezeigt werden, wie obige Beispiel-Aufzählung

- **Fett**,
- **farbig**,
- *Kursiv*,
- unterstrichen,
- doppelt unterstrichen,
- unterschlängelt,
- ~~horizontal durchgestrichen~~,
- ~~schräg/durchgestrichen~~,

auch als Liste gesetzt werden kann und wie ein selber definiertes Aufzählungszeichen

- **Fett**,
- **farbig**,
- * *Kursiv*,
- > unterstrichen,
- + doppelt unterstrichen,
- zum Ersten: unterschlängelt,
- zum Zweiten: ~~horizontal durchgestrichen~~,
- als letztes noch: ~~schräg/durchgestrichen~~,

gesetzt werden kann.

4.1.1. Gliederung / Strukturierung

Wie Eingangs dieses Kapitels erwähnt, ist dieses Template so vorbereitet, dass die Gliederung / Strukturierung des Inhaltes in Kapitel (section), Unterkapitel (subsection) und Unterunterkapitel (subsubsection) im Haupt-Dokument gemacht wird. Aus dem Haupt-Dokument werden dann die Inhalte der einzelnen Kapitel (section), Unterkapitel (subsection) und Unterunterkapitel (subsubsection) referenziert (siehe Abb. 5: Kapitel-Struktur des Dokumentes auf Seite 6).

Sollen tiefere Kapitel-Strukturen als Unter-Unter-Kapitel verwendet werden, so ist dies zwar möglich, bedeutet aber einiges an Aufwand, da dies nicht mit den \LaTeX Standardmitteln gemacht werden kann. Dies ist ein bewusster Entscheid von Leslie Lamport¹ - das La in \LaTeX steht für Lamport[2].

LaTeX's set of "sections" stops at the level of `\subsubsection`. This reflects a design decision by Lamport — for, after all, who can reasonably want a section with such huge strings of numbers in front of it[4]?

Somit bleibt für die Gliederung / Strukturierung der referenzierten Dokumente nur noch die Gliederung / Strukturierung in sogenannte Absätze. Es wäre technisch zwar möglich, auch in den aus dem Haupt-Dokument referenzierten Dokumenten Kapitel und Unterkapitel zu definieren. Davon rate ich jedoch dringend ab, da die Übersicht damit garantiert verloren geht und die am Ende resultierende tatsächliche Gliederung / Strukturierung eher ein Zufallsprodukt als wirklich unter Kontrolle ist.

In den Inhalts-Dokumenten wollen wir also nur fachlichen Inhalt, keine Kapitel Strukturierung. Es können weitere Dateien eingebunden werden. Dabei beschränken wir uns jedoch auf inhaltliche Elemente wie Grafiken und Tabellen.

4.1.2. Absätze

Folgendes Beispiel zeigt, wie Absätze erfasst werden:

Somit bleibt für die Gliederung Strukturierung der referenzierten Dokumente nur noch die Gliederung / Strukturierung in sogenannte Absätze. Es wäre technisch zwar möglich, auch in den aus dem Haupt-Dokument referenzierten Dokumenten Kapitel und Unterkapitel zu definieren. Davon rate ich jedoch dringend ab, da die Übersicht damit garantiert verloren geht und die am Ende resultierende tatsächliche Gliederung / Strukturierung eher ein Zufallsprodukt als wirklich unter Kontrolle ist.

In den Inhalts-Dokumenten wollen wir also nur fachlichen Inhalt, keine Kapitel Strukturierung. Es können weitere Dateien eingebunden werden. Dabei beschränken wir uns jedoch auf inhaltliche Elemente wie Grafiken und Tabellen.

Abb. 6: Absätze im Dokument

Ein neuer Absatz wird durch eine Leerzeile im Text erzeugt. Die Anzahl der Leerzeilen und die Anzahl der Leerzeichen zwischen einzelnen Wörtern spielen keine Rolle. \LaTeX formatiert den (Fließ-) Text innerhalb eines Absatzes selber. Einfluss darauf wird über spezielle \LaTeX -Befehle genommen, die in folgenden Kapiteln behandelt werden.

4.1.3. Schriftarten und Farben

Für die bereits Eingangs dieses Kapitels gezeigten Schriftarten und Farben:

An dieser Stelle sollen exemplarische einige Beispiele gezeigt werden, wie in \LaTeX Texte **Fett**, **farbig**, *Kursiv*, unterstrichen, doppelt unterstrichen, unterschlängelt, horizontal durchgestrichen, ~~*schräg durchgestrichen*~~ gesetzt werden und wie Texte formatiert / strukturiert werden können.

ist folgender \LaTeX -Code notwendig:

An dieser Stelle sollen exemplarische einige Beispiele gezeigt werden, wie in \LaTeX Texte `\textbf{Fett}`, `\textcolor{red}{farbig}`, `\textit{Kursiv}`, `\underline{unterstrichen}`, `\uuline{doppelt unterstrichen}`, `\uwave{unter\underslängelt}`, `\sout{horizontal durchgestrichen}`, `\xout{schräg durchgestrichen}` gesetzt werden und wie Texte formatiert / strukturiert werden können.

Abb. 7: Schriftarten und Farben

In diesem Beispiel sieht man auch, wie \LaTeX die korrekte Trennung von Wörtern mitgeben kann (siehe die Worte unterschlängelt und strukturiert).

4.1.4. Aufzählungen & Bullet-Points

Für die ebenfalls Eingangs dieses Kapitels gezeigte Aufzählung mit den Standard Aufzählungszeichen, ist folgende Code notwendig:

```
\begin{itemize}[topsep=0em, partopsep=0em, parsep=0em, itemsep=0em]
  \item \textbf{Fett},
  \item \textcolor{red}{farbig},
  \item \textit{Kursiv},
  \item \underline{unterstrichen},
  \item \uuline{doppelt unterstrichen},
  \item \uwave{unterschlängelt},
  \item \sout{horizontal durchgestrichen},
  \item \xout{schräg durchgestrichen},
\end{itemize}
```

Abb. 8: Eigene Aufzählungszeichen

Über die verschiedenen Separatoren in der ersten Zeile, können die Abstände zwischen den einzelnen Aufzählungspunkten eingestellt werden.

Die mit diesem Code erzeugte Aufzählung ist auf Seite 7 zu sehen.

Für Aufzählung mit selber definierten Aufzählungszeichen, ist folgende Code notwendig:

```
\begin{itemize}[leftmargin=9em, topsep=0em, partopsep=0em, parsep=0em, itemsep=0em]
\item[--] \textbf{Fett},
\item[-] \textcolor{red}{farbig},
\item[*] \textit{Kursiv},
\item[\textgreater] \underline{unterstrichen},
\item[+] \uuline{doppelt unterstrichen},
\item[zum Ersten:] \uwave{unterschlängelt},
\item[zum Zweiten:] \sout{horizontal durchgestrichen},
\item[als letztes noch:] \xout{schräg durchgestrichen},
\end{itemize}
```

Abb. 9: Eigene Aufzählungszeichen

Auch hier werden über die verschiedenen Separatoren in der ersten Zeile die Abstände zwischen den einzelnen Aufzählungspunkten eingestellt. Da in diesem Beispiel jedoch 'sehr lange' Aufzählungszeichen (z.B. 'als letztes noch') verwendet werden, muss die ganze Aufzählung deutlich mehr eingerückt werden (leftmargin=9em).

Auch die mit diesem Code erzeugte Aufzählung ist auf Seite 7 zu sehen.

Dieses Beispiel zeigt, dass die Anzahl der Leerzeichen zwischen Worten resp, zwischen dem Aufzählungszeichen und den Worten keine Rolle spielt, da L^AT_EX die Formatierung selber vornimmt.

5. Arbeiten mit Tabellen

In diesem Kapitel werden einige Beispiele zu Tabellen gezeigt. Um die Darstellung von Tabellen besser zeigen zu können, ist jede Tabelle in einer eigenen Datei definiert, die mit `\input{tabellenDatei.tex}` eingebunden wird.

Hinweis: In vielen LaTeX-Editoren (u.a. auch in Texmaker) gibt es Kommandos oder Makros für die Erstellung von Tabellen. Ebenfalls gibt es Online Generatoren für LaTeX Tabellen².

5.1. Tabellen ohne Schraffierung

Tabellen sind in Latex nicht ganz einfach zu verwalten. Die hier gezeigten Beispiele sollten jedoch vielen Anforderungen genügen. Für kompliziertere Tabellen empfehle ich Google und Youtube, wo sehr viele Tutorials zur Verfügung stehen.

Starten wir mit einem einfachen Beispiel:

Spalte 1	Spalte 2	Spalte 3
Zelle 1.1	Zelle 1.2	Zelle 1.3
Zelle zwei.eins	Zelle zwei.zwei	Zelle zwei.drei

Tab. 1: Eine erste LaTeX-Tabelle

Der für diese Tabelle verwendete Code sieht wie folgt aus:

```
\begin{table}[h!]
\begin{center}
\begin{tabular}{|c|c|c|}
\hline
Spalte 1      &   Spalte 2      &   Spalte 3      \\
\hline
Zelle 1.1     &   Zelle 1.2     &   Zelle 1.3     \\
\hline
Zelle zwei.eins &   Zelle zwei.zwei &   Zelle zwei.drei \\
\hline
\end{tabular}
\caption{Eine erste LaTeX-Tabelle}
\end{center}
\end{table}
```

Abb. 10: Eine erste LaTeX-Tabelle

Die Spalten richten sich nach deren Inhalt aus und sind zentriert (c) ausgerichtet, unabhängig davon, wie die einzelnen Spalten in der `*.tex`-Datei formatiert sind.

In folgender Tabelle wollen wir die Spaltenbreite und die Spaltenausrichtung selber kontrollieren. Das Ergebnis sieht wie folgt aus:

Spalte 1	Spalte 2	Spalte 3
Linksbündig	Zentriert	Rechtsbündig
linksbündige Spalte	zentrierte Spalte	rechtsbündige Spalte

Tab. 2: Eine zweite LaTeX-Tabelle

Der dazu benötigte Code:

```
\begin{table}[h!]
\begin{center}
\begin{tabular}{p{4.5cm}|C{3.5cm}|R{4.5cm}}
Spalte 1 & Spalte 2 & Spalte 3 \\
\hline
Linksbündig & Zentriert & Rechtsbündig \\
\hline
linksbündige Spalte & zentrierte Spalte & rechtsbündige Spalte \\
\end{tabular}
\caption{Eine zweite LaTeX-Tabelle}
\end{center}
\end{table}
```

Abb. 11: Eine zweite LaTeX-Tabelle

Der hier gezeigte Code funktioniert nur, wenn in der Präambel die hier verwendeten Befehle eigenhändig definiert werden.

```
%
% =====
% Package für Tabellen:
% =====
%
\usepackage{tabularx}
\newcolumnntype{R}[1]{>{\raggedleft\arraybackslash}p{#1}}
\newcolumnntype{C}[1]{>{\centering\arraybackslash}p{#1}}
%
```

Abb. 12: LaTeX-Tabellen Präambel

Hier noch ein Beispiel mit verschiedenen Schriftarten in den Zellen und anderen Linien für die Separierung der Zeilen.

Wer	Wo	<i>Was</i>
ich	da	<i>dies</i>
du	hier	das

Tab. 3: Eine dritte LaTeX-Tabelle

Der hierfür benötigte Code:

```
\begin{table}[h!]
\begin{center}
\begin{tabular}{p{2cm} C{2cm} R{2cm}}
\toprule
\textbf{Wer} & & Wo & & \textit{Was} & \\
\midrule
\textbf{ich} & & da & & \textit{dies} & \\
\textbf{du} & & hier & & \text{das} & \\
\bottomrule
\end{tabular}
\caption{Eine dritte LaTeX-Tabelle}
\end{center}
\end{table}
```

Abb. 13: Eine dritte LaTeX-Tabelle

5.2. Tabellen mit Schraffierung

In den nächsten Beispielen wird gezeigt, wie Spalten resp. Zeilen eingefärbt werden:

	Spalte 1	Spalte 2	Spalte 3
Zeile 1	Zelle 1.1	Zelle 1.2	Zelle 1.3
Zeile 2	Zelle 2.1	Zelle 2.2	Zelle 2.3
Zeile 3	Zelle 3.1	Zelle 3.2	Zelle 3.3
Zeile 4	Zelle 4.1	Zelle 4.2	Zelle 4.3
Zeile 5	Zelle 5.1	Zelle 5.2	Zelle 5.3
Zeile 6	Zelle 6.1	Zelle 6.2	Zelle 6.3

Tab. 4: Eine vierte LaTeX-Tabelle

```
\begin{table}[h!]
\begin{center}
\begin{tabular}{lcccl}
\hline
& Spalte 1 & & Spalte 2 & & Spalte 3 & \\
\hline
Zeile 1 & \cellcolor{gray!50}Zelle 1.1 & & \cellcolor{gray!50}Zelle 1.2 & & \cellcolor{gray!50}Zelle 1.3 & \\
\hline
Zeile 2 & Zelle 2.1 & & Zelle 2.2 & & Zelle 2.3 & \\
\hline
Zeile 3 & \cellcolor{gray!50}Zelle 3.1 & & \cellcolor{gray!50}Zelle 3.2 & & \cellcolor{gray!50}Zelle 3.3 & \\
\hline
Zeile 4 & Zelle 4.1 & & Zelle 4.2 & & Zelle 4.3 & \\
\hline
Zeile 5 & \cellcolor{gray!50}Zelle 5.1 & & \cellcolor{gray!50}Zelle 5.2 & & \cellcolor{gray!50}Zelle 5.3 & \\
\hline
Zeile 6 & Zelle 6.1 & & Zelle 6.2 & & Zelle 6.3 & \\
\end{tabular}
\caption{Eine vierte LaTeX-Tabelle}
\end{center}
\end{table}
```

Abb. 14: Eine vierte LaTeX-Tabelle

Ein zweites Beispiel:

	Spalte 1	Spalte 2	Spalte 3
Zeile 1	Zelle 1.1	Zelle 1.2	Zelle 1.3
Zeile 2	Zelle 2.1	Zelle 2.2	Zelle 2.3
Zeile 3	Zelle 3.1	Zelle 3.2	Zelle 3.3
Zeile 4	Zelle 4.1	Zelle 4.2	Zelle 4.3
Zeile 5	Zelle 5.1	Zelle 5.2	Zelle 5.3
Zeile 6	Zelle 6.1	Zelle 6.2	Zelle 6.3

Tab. 5: Eine fünfte LaTeX-Tabelle

```

\begin{table}[h!]
\begin{center}
\begin{tabular}{p{1.5cm}|p{2.5cm}|p{2.5cm}|p{2.5cm}}
& \textbf{Spalte 1} & & \textbf{Spalte 2} & \textbf{Spalte 3} & \\
\hline
\textbf{Zeile 1} & \cellcolor{gray!50}Zelle 1.1 & & Zelle 1.2 & & \cellcolor{gray!50}Zelle 1.3 & \\
\hline
\textbf{Zeile 2} & \cellcolor{gray!50}Zelle 2.1 & & Zelle 2.2 & & \cellcolor{gray!50}Zelle 2.3 & \\
\hline
\textbf{Zeile 3} & \cellcolor{gray!50}Zelle 3.1 & & Zelle 3.2 & & \cellcolor{gray!50}Zelle 3.3 & \\
\hline
\textbf{Zeile 4} & \cellcolor{gray!50}Zelle 4.1 & & Zelle 4.2 & & \cellcolor{gray!50}Zelle 4.3 & \\
\hline
\textbf{Zeile 5} & \cellcolor{gray!50}Zelle 5.1 & & Zelle 5.2 & & \cellcolor{gray!50}Zelle 5.3 & \\
\hline
\textbf{Zeile 6} & \cellcolor{gray!50}Zelle 6.1 & & Zelle 6.2 & & \cellcolor{gray!50}Zelle 6.3 & \\
\end{tabular}
\caption{Eine fünfte LaTeX-Tabelle}
\end{center}
\end{table}

```

Abb. 15: Eine fünfte LaTeX-Tabelle

Ein drittes Beispiel:

Spalte 1	Spalte 2	Spalte 3
Zelle 1.1	Zelle 1.2	Zelle 1.3
Zelle 2.1	Zelle 2.2	Zelle 2.3

Tab. 6: Eine sechste LaTeX-Tabelle

```

\begin{table}[h!]
\begin{center}
\begin{tabular}{c|c|c}
\rowcolor{gray!50} Spalte 1 & Spalte 2 & Spalte 3 & \\
\hline
Zelle 1.1 & Zelle 1.2 & Zelle 1.3 & \\
\hline
Zelle 2.1 & Zelle 2.2 & Zelle 2.3 & \\
\end{tabular}
\caption{Eine sechste LaTeX-Tabelle}
\end{center}
\end{table}

```

Abb. 16: Eine sechste LaTeX-Tabelle

Wichtig: Die Formatierung der Tabellen in den `*.tex`-Dateien dient lediglich der Übersicht, resp. der Lesbarkeit des Codes für den Autor.

6. Arbeiten mit Bildern / Grafiken

In diesem Kapitel wird eine speziellere Variante der Platzierung von Bildern und Grafiken auf einer Seite gezeigt. Das Einbinden / Anzeigen von einzelnen Bildern ist in den vorhergehenden Kapiteln mehrfach zu sehen.

Der \LaTeX -Löwe einmal mit und einmal ohne Rahmen, nebeneinander platziert.



Abb. 17: \LaTeX -Löwe ohne Rahmen



Abb. 18: \LaTeX -Löwe mit Rahmen

Der \LaTeX -Code für die Platzierung der beiden \LaTeX -Löwen nebeneinander:

```
\begin{figure} [htb]
  \centering
  \begin{minipage} [t] {0.45\linewidth}
    \centering
    \includegraphics [width=0.9\linewidth] {./Bilder/LaTeX.jpg}
    \caption{\LaTeX-Löwe ohne Rahmen}
  \end{minipage}
  \hfill
  \begin{minipage} [t] {0.45\linewidth}
    \centering
    \fbox{\includegraphics [width=0.9\linewidth] {./Bilder/LaTeX.jpg}}
    \caption{\LaTeX-Löwe mit Rahmen}
  \end{minipage}
\end{figure}
```

Abb. 19: Der \LaTeX -Code zu den beiden \LaTeX -Löwen

Die vertikale Verteilung über die ganze Seite wird mit `\vfill` erreicht. Der \LaTeX -Code für die Gestaltung der ganzen Seite (inkl. der konkreten Platzierung der `\vfill`) ist in der Tex-Datei zu dieser Seite `./Inhalte/Inhalt_Bilder.tex` zu sehen.

7. Arbeiten mit Abkürzungen

In diesem Kapitel wird gezeigt, wie mit Abkürzungen gearbeitet wird.

In einer wissenschaftlichen Arbeit dürfen oft wiederholte Begriffe durchaus abgekürzt werden. Die Verwendung von Abkürzungen sollte allerdings auf den Leserkreis abgestimmt sein. Neben den im Duden aufgeführten Abkürzungen dürfen auch Abkürzungen spezieller Fachtermini verwendet werden. Seien Sie sparsam mit Abkürzungen. Sie dürfen auf keinen Fall den Fluss Ihrer Arbeit stören.

Wird auf die nachfolgend vorgestellte Art mit Abkürzungen gearbeitet, so kümmert sich LaTeX um die korrekte Verwendung von Abkürzungen (... einen Begriff bei erstmaliger Verwendung im Textabschnitt auszuschreiben und dabei die Abkürzung in Klammer zu setzen, bei allen weiteren Malen nur noch die Abkürzung einzusetzen ...). Natürlich lässt LaTeX verschiedene Arten der Übersteuerung dieses Verhaltens zu.

Anders als Fachausdrücke, deren Bedeutung sich Fachfremden nicht auf den ersten Blick erschliesst, gehören allgemein übliche Ausdrücke wie 'z. B.' oder 'usw.' nicht in das Abkürzungsverzeichnis.

7.1. Erfassen von Abkürzungen

Alle Abkürzungen sind in einer separaten Datei mit Namen 'Abkuerzungen.tex' definiert. Diese Datei befindet sich im Verzeichnis 'Verzeichnisse' des Dokumentenordners. Die Definitionen der Abkürzungen müssen mit folgendem Muster erstellt werden:

Erfassen von Abkürzungen

```
\acro{eth}[ETH]{Eidgenössische Technische Hochschule}
\acro{ethz}[ETHZ]{Eidgenössische Technische Hochschule Zürich}
\acro{epfl}[EPFL]{École polytechnique fédérale de Lausanne}
\acro{id}[ID]{Informatikdienste}
\acro{ppf}[PPF]{Procurement & Portfolio Management}
\acro{pm}[PM]{Portfolio Management}
```

7.2. Einsetzen von Abkürzungen

Soll nun eine solche Abkürzung verwendet werden, geht das wie folgt.

Latex-Text

Ich bin angestellt an der `\ac{eth}`, genauer an der `\ac{ethz}`. Neben der `\ac{ethz}` gibt es auch noch eine `\acs{eth}` in der Westschweiz, nämlich die `\ac{epfl}`. An der `\acs{eth}` bin ich bei den `\ac{id}` im Bereich `\ac{pm}` tätig. Das `\ac{pm}` ist eine Gruppe innerhalb der Sektion `\ac{ppf}`.

Die 'Norm' für die Verwendung von Abkürzungen sieht man mit der Verwendung der Abkürzung `\ac{ethz}` an der zweiten und dritten Stelle im Text.

Print-Text

Ich bin angestellt an der Eidgenössische Technische Hochschule (ETH), genauer an der Eidgenössische Technische Hochschule Zürich (ETHZ). Neben der ETHZ gibt es auch noch eine ETH in der Westschweiz, nämlich die École polytechnique fédérale de Lausanne (EPFL). An der ETH bin ich bei den Informatikdienste (ID) im Bereich Portfolio Management (PM) tätig. Das PM ist eine Gruppe innerhalb der Sektion Procurement and Portfolio Management (PPF).

Am besten sieht man das, wenn im Latex Editor Texmaker die Tex-Datei geöffnet hat und man sich mit **F1** die pdf Vorschau des Dokumentes anzeigen lässt.

7.3. Verwenden von Abkürzungen

Abkürzungen sollten in wissenschaftlichen Arbeiten, egal ob in Semester-, Prüfungs-, Bachelor- oder Masterarbeiten, möglichst spärlich und nur dann verwendet werden, wenn sie Klarheit und Lesbarkeit nicht beeinträchtigen. Dein angestrebter Leserkreis sollte sofort verstehen, was gemeint ist, oder es in einem vorangestellten Abkürzungsverzeichnis erklärt bekommen.

Abkürzungen in Fachpublikationen sind i.d.R weniger Problematisch, sofern die verwendeten Abkürzungen im entsprechenden Fachbereich üblich sind.

Führe keine Abkürzung ein, die du nicht mindestens drei- oder viermal verwendest. Verwende stattdessen einfach den vollen Begriff.

8. Arbeiten mit Fussnoten

Für die Verwendung von Fussnoten gelten folgende Regeln:

- Fussnoten werden durch hochgestellte Ziffern nach dem Satzzeichen eines Absatzes bzw. Satzes oder nach einem zu zitierenden Wort angegeben.
- Mehrere Fussnoten an derselben Stelle sind nicht sinnvoll.
- Fussnoten gelten als ganze Sätze und müssen daher mit Grossbuchstaben begonnen und mit einem Punkt beendet werden.
- Gliederungsüberschriften dürfen nicht mit einer Fussnote versehen werden.
- Fussnoten gehören immer auf dieselbe Seite wie die Fussnotenreferenz im Text³.
- Ein Zitat aus einer anderen als der Originalquelle zu übernehmen (rezitieren) ist zu vermeiden.

Weiter sollte auch beachtet werden, dass die Arbeit auch ohne das Lesen der entsprechenden Fussnote verständlich sein muss. Daher gehören z.B. für die Argumentation wichtige Thesen nicht in eine Fussnote. Der in Fussnoten stehende Text sollte darum kurz und präzise sein. Exkurse sind zu vermeiden.

8.1. Erfassen von Fussnoten

Soll nun eine solche Fuss- (resp. Endnote) gesetzt werden, geht das wie folgt:

Latex-Text

An dieser Stelle `\footnote{Fussnoten-Text}` wird eine Fussnote eingefügt.

In diesem Dokument erscheint die Fussnote dann aber erst am Ende des Dokumentes, vor der eidesstattlichen Erklärung (siehe auch Inhaltsverzeichnis).

Print-Text

An dieser Stelle⁴ wird eine Fussnote eingefügt.

9. Arbeiten mit Stichworten

Es ist im Rahmen einer wissenschaftlichen Arbeit i.d.R. nicht unbedingt erforderlich, ein Stichwortverzeichnis zu erstellen. Als erstes jedoch ein Beispiel mit etwas Text, der lediglich dazu dient einige Einträge im Stichwortverzeichnis zu erzeugen.

Auf dieser Seite sind die Begriffe System, Systemzustand, Systemelement und Emergenz von Interesse. Auf dieser Seite sei nochmals was gesagt zu System und Systemelement sowie zu Transformationsprozess.

Wichtige Hinweise:

- Das Stichwortverzeichnis muss separat erstellt werden. In Texmaker erfolgt dies mit der Funktionstaste `F12`.
- Das Standard-Stichwortverzeichnis ist meines Erachtens nicht schön. Darum habe ich für dieses eine eigene Stil-Datei `index.ist` erstellt. Diese muss in der Konfiguration von Texmaker angegeben werden (siehe auch Abb. 2: Konfiguration Latex: Befehle auf Seite 3).

9.1. Erfassen von Stichworten

Um einen Begriff (oder Namen) in das Stichwortverzeichnis aufzunehmen, wird der entsprechende Begriff (oder Namen) wie folgt markiert:

Latex-Text

Die Begriffe System`\index{System}`, Systemzustand`\index{System!Zustand}` und Systemelement`\index{System!Element}` sollen im Stichwortverzeichnis aufgenommen werden.

Im gedruckten Text sieht das dann wie folgt aus:

Print-Text

Die Begriffe System, Systemzustand und Systemelement sollen im Stichwortverzeichnis aufgenommen werden.

Im gedruckten Text ist nicht zu erkennen, ob ein bestimmter Begriff (oder Name) im Stichwortverzeichnis aufgenommen wird.

10. Arbeiten mit Quellenangaben

Quellenangaben und Zitate erfüllen in wissenschaftlichen Texten bzw. in wissenschaftlichen Arbeiten zwei Hauptziele[3]:

- a) Nachvollziehbarkeit der Argumentation, des (gedanklichen) Experiments Ein wissenschaftlicher Beitrag kann nur dann weiterverwendet werden, wenn sich die Argumentation, das (gedankliche) Experiment für die Lesenden überprüfen lässt.
- b) Unterscheidung zwischen eigenen und fremden Gedanken (geistiges Eigentum) Ideen, Beispiele, ein bestimmtes methodisches Vorgehen, Bilder, Grafiken, Tabellen u.a., die aus anderen Texten stammen, müssen als fremdes geistiges Eigentum ausgewiesen werden.

Wichtige Regel: Quellenangaben gehören zum Satz dazu und stehen daher VOR dem Punkt bzw. Satzzeichen.

10.1. Erfassen von Quellen

Sämtliche (Literatur-) Quellen die im Dokument referenziert werden müssen in einer separaten Datei mit Endung `*.bib` in einem speziellen Format erfasst werden.

```
@BOOK{Kottwitz2011,  
  AUTHOR    = {Kottwitz, Stefan},  
  YEAR      = {2011},  
  TITLE     = {LaTeX Beginner's Guide},  
  EDITION   = {},  
  ISBN      = {978-1-847-19987-4},  
  PUBLISHER = {Packt Publishing Ltd},  
  ADDRESS   = {Birmingham}  
}  
  
@MISC{Lamport2017,  
  AUTHOR    = {Lamport, Leslie},  
  YEAR      = {2017},  
  TITLE     = {{LESLIE LAMPORT'S HOME PAGE}},  
  URL       = {http://lamport.org/},  
  NOTE      = {Online; gesehen 20. September 2017}  
}
```

Abb. 20: Erfassen der Quellen

Auf der Internet-Seite <http://literatur-generator.de/> kann man sich solche Einträge generieren lassen.

10.2. Verwenden von Quellen

Um auf eine verwendete Quelle zu verweisen, wird der Verweis wie folgt eingesetzt:

Latex-Text

Are you ready to leave those "what you see is what you get" word processors behind and to enter the world of real, reliable, and high-quality typesetting `\cite{Kottwitz2011}`?

Im gedruckten Text sieht das dann wie folgt aus:

Print-Text

Are you ready to leave those "what you see is what you get" word processors behind and to enter the world of real, reliable, and high-quality typesetting[1]?

10.3. Quellenverzeichnis erstellen

Die in der `*.bib` Datei erfassten Informationen zu der Quelle sind im Literatur- / Quellenverzeichnis aufgeführt.

Literatur- / Quellenverzeichnis

[1] Stefan Kottwitz. LaTeX Beginner's Guide. Packt Publishing Ltd, Birmingham, 2011. ISBN 978-1-847-19987-4.

Abb. 21: Auflistung der Quellen im Literatur- / Quellenverzeichnis

Erstellt wird das Literatur- / Quellenverzeichnis mit folgendem Befehl:

```
%
% =====
% Literatur- und Quellenverzeichnis der Arbeit
% =====
%
\pagebreak
\clearpage
\markright{Literatur- / Quellenverzeichnis}
\clearpage
\addcontentsline{toc}{section}{Literatur- / Quellenverzeichnis}
\renewcommand{\refname}{Literatur- / Quellenverzeichnis}
\bibliography{./Verzeichnisse/Literatur}
%
```

Abb. 22: Erstellen des Literatur- / Quellenverzeichnis

Ergänzende Materialien / Anhang

An dieser Stelle können ergänzende Materialien und/oder Anhänge eingefügt werden.

Literatur- / Quellenverzeichnis

- [1] Stefan Kottwitz. LaTeX Beginner's Guide. Birmingham: Packt Publishing Ltd, 2011. ISBN: 978-1-847-19987-4.
- [2] Leslie Lamport. LESLIE LAMPORT'S HOME PAGE. Online; gesehen 20. September 2017. 2017. URL: <http://lamport.org/>.
- [3] Fachhochschule Nordwestschweiz. Quellenangaben und Zitate. Online; gesehen 20. September 2017. 2017. URL: http://www.schreiben.zentrumlesen.ch/stud_zitieren.cfm.
- [4] FAQ Tex. How to create a subsubsection. Online; gesehen 20. September 2017. 2017. URL: <http://www.tex.ac.uk/FAQ-subsubsub.html>.

Stichwort- / Namensverzeichnis

Emergenz, 19

MikTex, 2

System, 19

 Element, 19

 Zustand, 19

Texmaker, 2

Transformationsprozess, 19

Fussnoten

¹Leslie Lamport auf Wikipedia:

https://de.wikipedia.org/wiki/Leslie_Lamport

²<http://www.tablesgenerator.com/>

³Keine Regel ohne Ausnahme! In diesem Dokument werden alle Fussnoten am Ende des Dokumentes aufgeführt, um den Textfluss und das Seitenlayout nicht zu sehr zu stören.

⁴Fussnoten-Text

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit mit Namen ***Titel der Doku*** ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

(Ort, Datum)

(Peter Kessler, 1234567)