

Beispieldatei für ein längeres, einseitig gedrucktes Dokument mit KOMA-Script

TeXLaTeX-Usergroup Berlin

rmfberlin

E-Mail: listen@rmf.berlin

Version: 0.2
im Februar 2018

Inhaltsverzeichnis

1	Aufbau der Gliederung	3
1.1	Kapitel und Abschnitte	3
1.2	Aufzählungen	4
1.2.1	Verschachtelte Aufzählungen	4
1.2.2	Nummerierte Aufzählungen	4
2	Werkzeuge	5
2.1	Empfehlenswerte Literatur	5
2.2	Arbeitsorganisation – Workflow	5
2.3	Editor	5
2.3.1	TeXstudio/Textmaker	6
2.3.2	Visual Studio Code	6
2.4	texdoc	7
2.5	Git – Backups	7
2.6	Literaturdatenbank – JabRef	7
2.7	Literaturverzeichnis – Tipps	7
	Literaturverzeichnis und Web-Links	9

1 Aufbau der Gliederung

Dieses Dokument basiert auf der Report-Klasse von KOMA-Script.¹

1.1 Kapitel und Abschnitte

Der Bericht lässt sich in Kapitel `\chapter`, Abschnitte `\section`, Unterabschnitte `\subsection`,² Unterunterabschnitte `\subsubsection` und auf Gliederungsstufe fünf und sechs noch mit `\paragraph` und `\subparagraph` gliedern.³

Hiermit wird gleich die zweite Fußnote mit dem Befehl `\footnote` gezeigt. Die erste Fußnote wurde durch die Referenz zur Literatur erzeugt.

Es wird empfohlen jeder Gliederungsstufe, ein `\label` und eine Zeile `% Gliederungsstufe (end)` hinzuzufügen. Der Befehl `\label` wird für die Querverweise innerhalb des Dokuments benötigt.⁴

Für den besseren Überblick gibt man dem Argument für `\label` ein Kürzel – eine Art Etikett – voranzustellen, z. B. `cha:` für `chapter` oder `sec:` für `Abschnitt`.⁵

Beispiel:

```
1 \chapter{Erstes Kapitel}
2 \label{cha:erstesKapitel}
3 \section{Erster Abschnitt}
4 \label{sec:ersterAbschnitt}
5 \subsection{Erster Unterabschnitt}
6 \label{sub:ersterUnterabschnitt}
7 % subsection erster Unterabschnitt (end)
8 % section erster abschnitt (end)
9 % chapter erstes kapitel(end)
```

Es wird ebenfalls empfohlen, gleich eine Kommentarzeile einzufügen, die das Ende des jeweiligen Abschnitts anzeigt.

1 Markus Kohm: Die Anleitung KOMA-Script. Stand 20.08.2017, 2018, URL: <https://komascript.de/~mkohm/scrguide.pdf> (besucht am 29. 01. 2018).

2 **schlosser.**

3 \LaTeX -Befehle sind in `in graue Boxen` gesetzt.

4 **schlosser.**

5 **schlosser.**

1.2 Aufzählungen

In diesem Abschnitt geht es um Aufzählungen:

- dies
- das und
- jenes

1.2.1 Verschachtelte Aufzählungen

Und in diesem Unterabschnitt um verschachtelte Aufzählungen:

- erste Ebene
 - zweite Ebene
 - * dritte Ebene
 - bis zu insgesamt vier Verschachtelungen

1.2.2 Nummerierte Aufzählungen

Interessant sind noch verschachtelte Listen wie z. B. mit führenden Ziffern, führenden Buchstaben usw. Dazu muss das Paket »enumerate« geladen sein. Eine ausführliche Beschreibung der verschiedenen Listen und Aufzählungen ist unter⁶ zu finden.

Beispiel:

1. Eins
 - a) zweite Ebene
 - i. dritte Ebene
 - A. bis zu insgesamt vier Verschachtelungen
2. Zwei
3. Drei

Auflistungen lassen sich vielfältig anpassen und gestalten.⁷

⁶ Sascha Frank: Auflistungen & Aufzählungen, Jan. 2018, URL: <https://www.namsu.de/Extra/befehle/Auflistungen.html> (besucht am 29.01.2018).

⁷ schlosser.

2 Werkzeuge

2.1 Empfehlenswerte Literatur

Für den schnellen und erfolgversprechenden Einstieg in \LaTeX lohnt sich das Buch von Schlosser¹ und für die normgerechte wissenschaftliche Arbeit das Buch und die Website von Scholz² (mit Beispieldatei für \LaTeX ³).

2.2 Arbeitsorganisation – Workflow

Es gibt viele Wege zum Erfolg, um eine wissenschaftliche Arbeit zu verfassen oder ein Buch zu schreiben.⁴ Wichtig ist es, sich schon vorher ein paar Gedanken zur »Ethik, Inhalt und Form wissenschaftlicher Arbeiten, wissenschaftliches Handwerkszeug, Quellenarbeit, Methoden, Projektmanagement [und der] Präsentation zu machen«. ⁵ Auch lohnt es sich einen Blick auf die Vorgehensweise von Programmiererinnen zu machen, gerade zum Thema Strukturierung des Quelltextes und zum Projektmanagement.⁶

2.3 Editor

Es gibt kaum ein emotionaleres Thema als »der richtige Editor« für \LaTeX . Die einen schwören auf Emacs oder Vim oder auf die speziellen \LaTeX -Editoren wie Kile, \TeX maker, \TeX shop usw. Die letzteren sind Editoren, die spezielle Aufgaben erfüllen und dafür sicherlich auch optimiert sind.

Universelle Editoren, die mehrere Computer-Sprachen beherrschen, halte ich jedoch für günstiger, als sich in verschiedene Editor für verschiedene Aufgaben/Sprachen einzuarbeiten. Und universelle Editoren haben meines Erachtens noch einen wichtigen Vorteil: Da diese von vielen unterschiedlichen Programmieren genutzt werden, sind diese auch in ständiger Entwicklung und häufig den eigenen Bedürfnissen und Vorlieben sehr gut anpassbar.

1 Joachim Schlosser: Wissenschaftliche Arbeiten schreiben mit LaTeX, 6. Auflage, 2016.

2 Dieter Scholz: Diplomarbeiten normgerecht verfassen. Schreibtipps zur Gestaltung von Studien-, Diplom- und Doktorarbeiten, 2006, URL: <http://buch.profscholz.de/> (besucht am 01. 02. 2018).

3 Ders.: Schreibtipps zur Gestaltung von Studien-, Diplom- und Doktorarbeiten – LaTeX-Vorlage. Musterdateien einer Diplomarbeit für LaTeX, URL: http://www.fzt.haw-hamburg.de/pers/Scholz/schreibtipps/Schreibtipps_LaTeX_Vorlage.zip (besucht am 01. 02. 2018).

4 Rainer-Maria Fritsch: Ein Workflow für ein Sachbuch, URL: <https://texlatex.rmfb.berlin/journal/articles/ein-workflow-f%C3%BCr-ein-sachbuch.html> (besucht am 14. 03. 2018).

5 W3L GmbH: Wissenschaftliches Arbeiten, URL: https://www.springer-campus-it-onlinestudium.de/w3lmedia/W3L/Medium136569/WissenschaftlichesArbeiten_Auflage_2_Praesentation2011.pptx (besucht am 14. 03. 2018).

6 Kathrin Passig/Johannes Jander: Weniger schlecht programmieren, Heidelberg 2013.

Ein Editor ist das ganz persönliche Werkzeug, das sich in die eigene Hand schmiegen muss. Im Folgenden sind Editoren aufgelistet, die frei und Open Source Software sind und für alle drei Betriebssysteme Windows, macOS und Linux zur Verfügung stehen.⁷ Alle Editoren können zwischen dem erzeugten PDF und dem \LaTeX -Quelltext schnell und einfach hin- und herspringen. Rechtschreibprüfung für viele Sprachen sind zuschaltbar.

Zunächst folgen auf \LaTeX spezialisierte Editoren, dann Editoren, die mehr als eine (Computer-)Sprache verstehen.

2.3.1 TeXstudio/Textmaker

TeXstudio⁸ und Textmaker⁹ sind auf das Erstellen von \LaTeX -Dokumenten hoch spezialisiert. Sie bieten eine große Funktionsvielfalt. Das kann auch zugleich ein Problem sein. Ein geisteswissenschaftliche oder eine juristische Arbeit nutzt einen anderen Befehlsvorrat aus dem großen \LaTeX -Kosmos als eine Arbeit aus den Bereichen Mathematik, Informatik, Technik oder Naturwissenschaften.

Die integrierten PDF-Betrachter bieten den großen Vorteil, dass schnell und einfach zwischen dem Text im PDF und dem \LaTeX -Quelltext hin- und hergesprungen werden kann.

2.3.2 Visual Studio Code

Die Firma Microsoft stellt ebenfalls als freien und Open Source Editor »Visual Studio Code« zur Verfügung.¹⁰ Mit der Erweiterung »LaTeX-Workshop« entsteht eine vollfunktionsfähige \LaTeX -Entwicklungsumgebung. »Visual Studio Code« hat eine große Entwicklergemeinschaft und eine sehr große Anzahl von Nutzer*innen.

Die Erweiterung »LaTeX-Workshop« erkennt automatisch, welches Dokument das Masterdokument ist, welche Dateien geändert wurden und neu kompiliert werden müssen und prüft auch, ob sich an der Literaturdatenbank etwas geändert hat.¹¹

Wichtigstes Argument für VS Code ist die sehr gute Integration der Versionsverwaltung »Git«.

Viele Erweiterungen wie z. B. Todo-Listen lassen diesen Editor schnell zu einem universellen Werkzeug für größere Projekte wie eine Master- oder Doktorarbeit werden.

Wer neben \LaTeX auch in HTML, CSS, markdown usw. schreibt, könnte hier einen sehr modernen Editor finden.

7 Wer nur auf dem Mac \LaTeX -Dokumente erstellen will, dem sei Textmate von <https://macromates.com/> ausdrücklich empfohlen.

8 TeXstudio – integrated writing environment for creating LaTeX documents, URL: <https://www.texstudio.org/> (besucht am 04. 02. 2018).

9 TEXMAKER Free cross-platform LaTeX editor, URL: <http://www.xm1math.net/texmaker/index.html> (besucht am 04. 02. 2018).

10 Visual Studio Code, URL: <https://code.visualstudio.com/> (besucht am 04. 02. 2018).

11 James Yu: LaTeX Workshop, URL: <https://marketplace.visualstudio.com/items?itemName=James-Yu.latex-workshop> (besucht am 04. 02. 2018).

2.4 texdoc

Vielen Anfänger*innen unbekannt ist der Befehl `texdoc`. `texdoc` ist ein Kommandozeilenprogramm, das die Dokumentation zu einem \LaTeX -Paket zeigt. Durch die Dokumentationen kann in vielen Fällen die beste Hilfe zu den jeweiligen Paketen erhalten werden. Allerdings ist die Sprache der Paketdokumentationen oftmals Englisch.

Um `texdoc` aufzurufen, muss man ein Terminal-Fenster öffnen und den Befehl `texdoc name_des_pakets` eingeben. Zum Beispiel öffnet `texdoc url` die Dokumentation zum Paket »url.sty« in der Version 3.4 von Donald Arseneau, zuletzt aktualisiert am 2013-09-16. Dieses Paket wird in diesem Dokument benutzt, um Webadressen anzuzeigen.

In Visual Studio Code kann mit der Tastenkombination `ctrl` + `↑` + ``` auch ein integriertes Terminal aufgerufen werden. Auch von hier kann mit dem `texdoc`-Befehl die Dokumentation zu Klassen und Paketen aufgerufen werden.

2.5 Git – Backups

Versionskontrollsysteme (VCS) protokollieren Änderungen an einer Datei oder einer Anzahl von Dateien über die Zeit hinweg, so dass man zu jedem Zeitpunkt auf Versionen und Änderungen zugreifen kann. Git als Versionskontrollsystem ermöglicht ebenso verschiedene Versionen einer Datei vergleichend anzuzeigen.

»Git« ist ein freies und Open Source Versionskontrollsystem, das schnelle und sehr effizient arbeitet.¹² Egal ob »nur« Hausarbeit oder große Projekte wie wissenschaftliche Abschlussarbeiten: regelmäßige Backups und ein Versionskontrollsystem helfen die eigene Arbeit zu sichern.

Wer ohne Backups und ohne ein Versionskontrollsystem arbeitet, handelt leichtsinnig und gefährdet seinen Erfolg(zum Thema Backups siehe auch¹³).

2.6 Literaturdatenbank – JabRef

JabRef ist ein freier und open source Literatur-Referenz Manager. JabRef basiert auf Java VM und ist gleichermaßen unter Windows, macOS und Linux einsetzbar. Die Oberfläche erschließt sich schnell, und es wird das für \LaTeX notwendige `*.bib` erzeugt. Die o. g. Editoren können auf die Literaturreferenzen der JabRef-Datenbank zugreifen.¹⁴

2.7 Literaturverzeichnis – Tipps

Das Literaturverzeichnis wird mit dem Befehl `\printbibliography` ausgegeben. Diesem Befehl kann am Ende mit eckigen Klammern ein neuer Name für das Literaturverzeichnis mit-

¹² Ben Straub Scott Chacon: Pro Git, 2014, URL: <https://git-scm.com/book/en/v2> (besucht am 04. 02. 2018).

¹³ Rainer-Maria Fritsch: Ein Workflow für ein Sachbuch, URL: <https://texlatex.rmfb.berlin/journal/articles/ein-workflow-f%C3%BCr-ein-sachbuch.html> (besucht am 14. 03. 2018).

¹⁴ JabRef – open source bibliography reference manager, URL: <http://www.jabref.org/#jabref> (besucht am 04. 02. 2018).

gegeben werden. Soll auch die gesamte Literatur ausgedruckt werden, selbst wenn diese in dem Text nicht zitiert wurde, dann den Befehl `\nocite*` voran stellen.

```
1 \nocite{*} % Das gesamte Literaturverzeichnis ausgeben, auch die
   Einträge, die nicht im Text zitiert wurden.
2 \printbibliography[title={Literaturverzeichnis und Web-Links}] %
   Literaturverzeichnis umbenennen
```


Literaturverzeichnis und Web-Links

- Frank, Sascha: Auflistungen & Aufzählungen, Jan. 2018, URL: <https://www.namsu.de/Extra/befehle/Auflistungen.html> (besucht am 29. 01. 2018).
- Fritsch, Rainer-Maria: Ein Workflow für ein Sachbuch, URL: <https://texlatex.rmf.berlin/journal/articles/ein-workflow-f%C3%BCr-ein-sachbuch.html> (besucht am 14. 03. 2018).
- GmbH, W3L: Wissenschaftliches Arbeiten, URL: https://www.springer-campus-it-onlinestudium.de/w3lmedia/W3L/Medium136569/WissenschaftlichesArbeiten_Auflage_2_Praesentation2011.pptx (besucht am 14. 03. 2018).
- JabRef – open source bibliography reference manager, URL: <http://www.jabref.org/#jabref> (besucht am 04. 02. 2018).
- Kohm, Markus: Die Anleitung KOMA-Script. Stand 20.08.2017, 2018, URL: <https://komascript.de/~mkohm/scrguide.pdf> (besucht am 29. 01. 2018).
- Kottwitz, Stefan: LaTeX Cookbook, 2015.
- Lingnau, Anselm: LaTeX Hacks. Tipps und Techniken für den professionellen Textsatz, 2007.
- Passig, Kathrin und Johannes Jander: Weniger schlecht programmieren, Heidelberg 2013.
- Schlosser, Joachim: Wissenschaftliche Arbeiten schreiben mit LaTeX, 6. Auflage, 2016.
- Scholz, Dieter: Diplomarbeiten normgerecht verfassen. Schreibtipps zur Gestaltung von Studien-, Diplom- und Doktorarbeiten, 2006, URL: <http://buch.profscholz.de/> (besucht am 01. 02. 2018).
- Ders.: Schreibtipps zur Gestaltung von Studien-, Diplom- und Doktorarbeiten – LaTeX-Vorlage. Musterdateien einer Diplomarbeit für LaTeX, URL: http://www.fzt.haw-hamburg.de/pers/Scholz/schreibtipps/Schreibtipps_LaTeX_Vorlage.zip (besucht am 01. 02. 2018).
- Scott Chacon, Ben Straub: Pro Git, 2014, URL: <https://git-scm.com/book/en/v2> (besucht am 04. 02. 2018).
- Storm, Ingo T.: 3, 2, 1 – ewig deins! So finden Sie die richtige Backup-Strategie, in: c't – Magazin für Computertechnik 13 (2013), S. 112.
- TEXMAKER Free cross-platform LaTeX editor, URL: <http://www.xmlmath.net/texmaker/index.html> (besucht am 04. 02. 2018).
- TeXstudio – integrated writing environment for creating LaTeX documents, URL: <https://www.texstudio.org/> (besucht am 04. 02. 2018).
- unbekannt: Techniken wissenschaftlichen Arbeitens, URL: <http://www.techniken-wissenschaftlichen-arbeitens.de/organisation/arbeitszeit.htm> (besucht am 14. 03. 2018).
- Visual Studio Code, URL: <https://code.visualstudio.com/> (besucht am 04. 02. 2018).
- Yu, James: LaTeX Workshop, URL: <https://marketplace.visualstudio.com/items?itemName=James-Yu.latex-workshop> (besucht am 04. 02. 2018).