

Anleitung zur L^AT_EX-Vorlage

Eine L^AT_EX-Vorlage mit vielen vordefinierten Befehlen

Version: 1.557, 23. Oktober 2023

Philipp Winniewski
Mirco Lukas

L^AT_EX

Inhaltsverzeichnis

1	Einführung	1
1.1	Vorwort	1
1.2	Voraussetzungen	2
1.3	Einrichtung	2
2	Funktionen	5
2.1	Allgemeine Einstellungen	5
2.1.1	Titelseite	5
2.1.2	Indices	6
2.1.3	Literaturverzeichnis	7
2.2	Auswahl der Dokumentklasse	7
2.2.1	Skript	8
2.2.2	Mitschrift	9
2.2.3	Buch	10
2.2.4	Praesentation	11
2.3	Neue Befehle und Abkürzungen	11
2.3.1	Befehle für Fließtexte	11
2.3.2	Befehle für mathematische Umgebungen	14
2.4	Begriffsdefinitionen	14
2.5	Beispiele und Beweise	15
2.6	Aufzählungen	18
2.7	Zitate	20
2.8	Boxen	21
2.8.1	Farbige Kästchen	21
2.8.2	Warnungskästchen	24

1 Einführung

1.1 Vorwort

Herzlich willkommen zur Anleitung der L^AT_EX-Vorlage von MIRCO LUKAS! Auf den folgenden Seiten geht es darum, die Möglichkeiten des Designs kennenzulernen und für sich nutzbar zu machen. Hierbei wird davon ausgegangen, dass die meisten Grundlagen und das Arbeitsprinzip der Textsatzsprache L^AT_EX bekannt sind. Dennoch richtet sich die Anleitung eher an fortgeschrittene Einsteiger, die die Vorlage möglichst schnell und unkompliziert verwenden möchten. Weit fortgeschrittene Anwender und Experten wird die Dokumentation zur Vorlage interessieren, in welcher Befehle samt ihrer Parameter exakt notiert sind. Die Nutzung der Vorlage selbst wird erfahrene Personen vor keine großen Schwierigkeiten stellen und sie können somit die vorliegende Anleitung einfach überfliegen.

L^AT_EX stellt selbstverständlich diverse Dokumentklassen bereit, die schon viele (auch veränderbare) Standardeinstellungen umsetzen. Außerdem gibt es die Möglichkeit, an den vielen feinen Drehreglern der Software mit einzelnen Befehlen zu schrauben. Und nicht zuletzt existieren noch Unmengen von Paketen, die einfach per Befehl heruntergeladen und eingebunden werden können. Wozu also eine spezielle Vorlage? Nun, es gibt eine gewisse Menge an Dingen, die man immer wieder braucht, z. B. die korrekte Formatierung von „z. B.“ oder farbige Infoboxen. Auch bestimmte Einstellungen für das Schreiben in deutscher Sprache müssen getätigt werden. Daher ist es sinnvoll, all diese Standardsachen in eine eigene Vorlage zu kombinieren. Insbesondere sind hier alle Pakete und Einstellungen in jeder Hinsicht aufeinander abgestimmt und mit deutschen Vorgaben und Format-Richtlinien kompatibel. Somit hat man stets den perfekten Ausgangspunkt, um mit einem neuen Werk zu starten, ganz ohne Vorarbeit.

Damit man aber wirklich alle Vorteile der Vorlage nutzen kann, muss man sie zuerst kennenlernen. Daher sind in dieser Anleitung sämtliche (nicht im Hintergrund arbeitenden) Funktionen aufgelistet. Zu jeder Funktion gibt es eine Beschreibung als Text, den Code und mindestens ein Beispiel.

Eine Anmerkung noch: Auch die vorliegende Anleitung wurde mit eben dieser Formatvorlage erstellt!

1.2 Voraussetzungen

Die Anleitung und die Verwendung der Vorlage setzt einen bestimmten Wissensstand voraus. Der Nutzer muss grundsätzlich imstande sein, mit \LaTeX zu arbeiten. Selbstverständlich muss bekannt sein, was \LaTeX eigentlich ist und was es macht. Außerdem ist Vorwissen über das Prinzip der Pakete und Befehle notwendig. Es schadet demnach nicht, sich schon in den Grundzügen mit \LaTeX auseinander gesetzt zu haben. Auf der anderen Seite nimmt die Vorlage einem viel Einarbeitung ab, da auch komplexere Funktionen leicht zugänglich gemacht werden und Feineinstellungen für Format und Sprache bereits integriert sind. Gerade deshalb ist das Design gut für das erste „richtige“ eigene Werk mit \LaTeX , nachdem man sich zuvor etwas in die Software eingearbeitet und selber ein wenig probiert hat. Jedoch werden auch fortgeschrittene Nutzer die eleganten Lösungen schätzen, die Zeit und Nerven sparen.

Zudem bestehen technische Voraussetzungen. Es muss eine \LaTeX -Software eingerichtet und funktionsfähig sein¹. Weil die Vorlage selber verschiedene Pakete benutzt, muss der Zugang zu diesen gewährleistet sein. Das bedeutet, dass entweder eine große Bibliothek mit allen Paketen auf dem jeweiligen System installiert ist oder die benötigten Pakete über eine bestehende Internetverbindung zu jedem Zeitpunkt nachgeladen werden können. Für den Download der Vorlage an sich braucht man ebenso eine Internetverbindung und einen Browser (oder ein Programm zum Download eines *GitHub-Repositorys*). Ansonsten braucht man nichts.

1.3 Einrichtung

Zuallererst muss die Vorlage heruntergeladen werden. Die Dateien findet man in einem *GitHub-Repository*². Zu erreichen ist das Projekt unter der URL <https://github.com/MircoL/LatexTemplateDE>. Auf der rechten Seite befindet sich eine Schaltfläche „Clone or download“³. Wählt man diesen an, kann „Download ZIP“ angeklickt werden und ein Download startet. Die Datei, das ZIP-Archiv, enthält die Vorlage in größenkomprimierter Form. Um die einzelnen Dateien zu erhalten, muss man das Archiv entpacken⁴.

Hat man das Archiv entpackt, erhält man einen Ordner mit folgender Dateistruktur:

¹Wir empfehlen TeXstudio (<http://www.texstudio.org/>)

²Ein *Repository* ist eine Art digitales Archiv zur Verwaltung von Dateien. Diese werden sortiert abgelegt, beschrieben und wenn man etwas ändert, wird der Änderungsverlauf detailliert dargestellt.

³Das Seitendesign kann sich natürlich immer mal wieder ändern. Die Buttons werden jedoch in etwa wie hier beschrieben zu finden sein.

⁴Zum Beispiel mit dem Programm 7-ZIP (<http://www.7-zip.de/download.html>)

1 Einführung

```
Ausführliche Dokumentation
--Grafiken
--AusführlicheDoku.pdf
--AusführlicheDoku.tex
--index.ist
--quellen.bib
--titel.jpg
--version.dat
doc
--docu_v6.pdf
--docu_v6.tex
--index.ist
--muster-buch.pdf
--muster-buch.tex
--muster-mitschrift.pdf
--muster-mitschrift.tex
--muster-praesentation.pdf
--muster-praesentation.tex
--muster-skript.pdf
--muster-skript.tex
--muster-thesis.pdf
--muster-thesis.tex
--titel.jpg
--version.dat
Vorlage
--Vorlage.tex
--index.ist
--quellen.bib
--titel.jpg
LICENSE
README.md
mircol-v6.tex
```

Im Unterordner „doc“ findet man diverse Beispieldokumente als PDF- und TEX-Datei. Zudem befindet sich hier die Dokumentation `docu_v6.pdf`. Die Muster-Dokumente eignen sich gut, um damit ein neues Werk zu beginnen oder zu lernen, wie ein bestimmtes Ergebnis erzielt werden kann. Dazu schaut man in die entsprechende TEX-Datei.

Der nächste Unterordner enthält eine blanke Vorlage, nämlich `Vorlage.tex`. In die Datei „quellen.bib“ fügt man die verwendeten Quellen in BibTeX-Formatierung ein.

1 Einführung

Das eigentliche Kernstück der Vorlage ist die Datei `mircol-v6.tex`, die die Einstellungen enthält. Im Grunde wäre also diese Datei mit dem Ordner „Vorlage“ zusammen die Minimalausstattung, um zu starten. Ein möglicher Projektordner „Meine_Abschlussarbeit“ für ein neues Schriftstück könnte dann so aussehen:

```
Meine_Abschlussarbeit
--Vorlage
----Vorlage.tex
----index.ist
----quellen.bib
----titel.jpg
--mircol-v6.tex
```

Man kann den Ordner `Vorlage` und die Datei `Vorlage.tex` auch umbenennen, z. B. in `Thesis` und `Thesis.tex`.

2 Funktionen

2.1 Allgemeine Einstellungen

Es gibt eine Reihe von Einstellungen, die man immer vornimmt, egal für welchen Dokumenttypen man sich entscheidet. Die Konfigurationen geschehen in dem Dokument, das als Vorlage dient und in welches man sein Werk hineinschreibt. Im Kopfbereich findet man den dafür wichtigen Code-Abschnitt:

```
all/Autoren      = {Philipp Winniewski&Mirco Lukas},
all/Titel        = {Anleitung zur \LaTeX{}-Vorlage},
all/Untertitel   = {Eine \LaTeX{}-Vorlage mit vielen vordefinierten Befehlen},
all/VersionPraefix = {1},
all/Version      = {true},
all/Icon/Breite  = {.7},
all/Icon/URL     = {},
all/Index/Boxen/blau/titel = {Liste der verf\ugbaren Befehle},
all/Index/Boxen/blau/zeigen = {false},
all/Index/Boxen/gelb/titel = {Liste der Hinweise},
all/Index/Boxen/gelb/zeigen = {false},
all/Index/Boxen/gruen/titel = {Liste der Tipps und Tricks},
all/Index/Boxen/gruen/zeigen = {false},
all/Index/Begriffe/titel = {Index},
all/Index/Begriffe/zeigen = {true},
all/Index/Literatur/titel = {Literaturverzeichnis},
all/Index/Literatur/zeigen = {false},
```

2.1.1 Titelseite

Man legt den Autor fest, indem man ihn in die erste Klammer schreibt. Bei mehreren Autoren trägt man ihre Namen mit einem Kaufmanns-Und „&“ getrennt hintereinander ein. Den Titel der Arbeit legt man in der nächsten Klammer fest. Dazu gibt es die Möglichkeit eines weiteren Untertitels in der dritten Klammer. Möchte man einen der Punkte weglassen, so lässt man die entsprechende Klammer leer.

2 Funktionen

Die Vorlage bietet eine automatische Versionsverwaltung. Sobald man speichert, wird eine neue Versionsnummer erstellt (auf dem Deckblatt). Damit hat man eine gute Orientierung darüber, welche Datei die aktuellste ist, falls man viele verschiedene Versionen hat. Durch einen Vergleich der Nummer kann man so entscheiden, welche Datei die neuste ist. Um die Versionsverwaltung zu aktivieren, muss man in die Klammer hinter `all/Version` „true“ schreiben; ansonsten „false“ zur Deaktivierung. Die große Versionsnummer eines Dokuments legt man noch manuell fest, nämlich mit `all/VersionPraefix`. Das, was in der Klammer dahinter steht, ist dann die Hauptnummer. Steht dort eine „42“, so erhält man mit steigender Anzahl an Speichervorgängen die Versionen „42.1“, „42.2“, „42.3“ usw.

Das Titelbild auf dem Deckblatt kann leicht festgelegt werden. Verwendet wird das Bild „titel.jpg“ im jeweiligen Ordner der Vorlage. Möchte man sein eigenes Bild benutzen, so muss dieses im JPG-Format vorliegen und ebenfalls „titel.jpg“ heißen. Nun ersetzt man die Bilddatei im Ordner durch sein eigenes Bild. Ab sofort wird das neue Bild als Titelbild eingebunden. Falls gar kein Bild erscheinen soll, löscht man einfach „titel.jpg“ aus dem Dokumentordner. Die Größe des Bildes legt man in der Zeile `all/Icon/Breite` fest. In der Klammer dahinter steht der Vergrößerungsfaktor. „1“ bedeutet, dass das Bild in seiner Breite die volle Seitenbreite einnimmt, „.5“ steht für „0,5“ und bedeutet halbe Seitenbreite. „1.5“ hieße eineinhalbfache Seitenbreite (was aber nicht so sinnvoll ist, weil das Bild ja dann aus der Seite herausragt).

2.1.2 Indices

Das Konzept der Boxen soll an anderer Stelle erklärt werden. Wichtig ist in diesem Zusammenhang zunächst, dass es die Möglichkeit gibt, farbige Textboxen zu benutzen. Es stehen drei Farben zur Verfügung, sodass Abschnitte gleicher Bedeutung immer die gleichen Box-Farbe nutzen. Beispielsweise kann man einen argumentativen Beweis in einem blauen, eine selbst entwickelte Aussage immer in einem grünen Kasten anzeigen lassen. Die Beweise und Aussagen erhalten alle einen Titel bzw. eine Überschrift.

Im Kopfbereich der Vorlage kann man nun einstellen, ob für die jeweilige Box-Farbe an Ende des PDFs ein Index mit Auflistung der Box-Titel dieser Box-Art angezeigt wird oder nicht. Den Index-Titel legt man in den Zeilen `all/Index/Boxen/.../titel` fest und die Entscheidung zur Darstellung des Index in der darauf folgenden Zeile `all/Index/Boxen/blau/zeigen`. Ein Beispiel soll dies verdeutlichen:

2 Funktionen

```
all/Index/Boxen/blau/titel      = {Liste der Beispiele},
all/Index/Boxen/blau/zeigen    = {false},
all/Index/Boxen/gelb/titel     = {Liste der mathematischen Formeln},
all/Index/Boxen/gelb/zeigen    = {false},
all/Index/Boxen/gruen/titel    = {Liste der Aussagen},
all/Index/Boxen/gruen/zeigen   = {true},
```

Es werden alle im Dokument vorkommenden Boxen angezeigt und in die PDF eingebaut. Aber nur die grünen Boxen werden am Schluss des Dokuments mit einem Index namens „Liste der Aussagen“ aufgelistet.

Auf die gleiche Weise funktioniert der Index der Begriffe, der ebenfalls im Kopfbereich des Dokuments aktiviert und deaktiviert wird.

2.1.3 Literaturverzeichnis

Am Beginn des Dokuments legt man fest, ob am Ende des PDFs ein Literaturverzeichnis angezeigt wird oder nicht. Zur Aktivierung setzt man einfach die Klammer der Zeile `all/Index/Literatur/zeigen` auf „true“ statt auf „false“. Die Überschrift des Verzeichnisses kann in der Zeile `all/Index/Literatur/titel` festgelegt werden. Ansonsten funktioniert dieses Literaturverzeichnis genauso, wie man es von klassischen L^AT_EX-Dokumenten gewohnt ist.

2.2 Auswahl der Dokumentklasse

Man kann zwischen fünf verschiedenen Dokumenttypen wählen. Je nachdem, was für ein Schriftstück erstellt werden soll, entscheidet man sich zwischen *Buch*, *Mitschrift*, *Skript*, *Thesis* und *Praesentation*. Im Kopfbereich des Dokuments befindet sich folgender Abschnitt:

```
\def\dokumentTyp{Skript} % oder
%\def\dokumentTyp{Thesis} % oder
%\def\dokumentTyp{Mitschrift} % oder
%\def\dokumentTyp{Praesentation}
```

Zur Wahl eines Dokumenttyps entfernt man das Prozentzeichen vor der entsprechenden Zeile und setzt die Zeichen vor alle anderen Zeilen. Um eine Thesis zu verfassen, müsste man den Code also wie folgt verändern:

```
%\def\dokumentTyp{Skript} % oder  
\def\dokumentTyp{Thesis} % oder  
\def\dokumentTyp{Mitschrift} % oder  
\def\dokumentTyp{Praesentation}
```

Achtung!

Hinweis zum Typ *Buch*: Bitte hier an die Beispieldatei `/doc/muster-buch.pdf` halten, da dieser Dokumenttyp 2022 komplett neu überarbeitet wurde.

2.2.1 Skript

Ein *Skript* hat ein Deckblatt, ein Inhaltsverzeichnis und die Ränder sind auf allen Seiten stets links und rechts gleich. Das Format ist DIN-A4. Das Deckblatt zeigt das Titelbild, Titel und Untertitel, Autor(en), ggf. Anmerkungen, Versionsnummer und Datum.

Dies eignet sich für längere, sachliche Texte. Beispiele für Skripte sind: Abschlussarbeiten, Anleitungen, Dokumentationen, Beschreibungen und alle Arten von stark strukturierten, langen Texten. Auch die hier vorliegende Anleitung nutzt genau diesen Typ.

Die Menge der notwendigen Einstellungen ist recht kurz. Im Kopfbereich gibt es nur folgende Zeile:

```
Skript/AnmerkungenTitelseite = {},
```

Diese Einstellung ist optional. Wenn man etwas in die Klammer schreibt (und diesen Dokumenttyp nutzt), so wird der Inhalt der Klammer über Datum und Version eingefügt. Das sieht dann so aus:



Das Ergebnis wird mit Hilfe der folgenden Einstellung erreicht:
`Skript/AnmerkungenTitelseite = {blabla},`

2.2.2 Mitschrift

Eine *Mitschrift* ist ein ein- oder mehrseitiges Dokument ohne Deckblatt. Auf jeder Seite sind Kopf- und Fußbereich vorhanden, durch Linien vom Seiteninhalt getrennt. Im Kopfbereich jeder Seite stehen Titel, Autor/-en, Datum, Zugehörigkeit zu einer Veranstaltung oder Sitzungsart und das Titelbild wird klein in der Mitte angezeigt. Der Fußbereich enthält die Seitenzahl. Es gibt kein Inhaltsverzeichnis.

Diese Art von Dokument eignet sich für alle Mitschriften, seien es Notizen einer Unterrichtsstunde oder Vorlesung, oder Protokolle einer Versammlung/Sitzung. Auch Hausaufgaben oder kurze gedankliche Skizzen lassen sich gut damit auf Papier bringen.

Für diese Dokumentart existieren mehrere Einstellungsmöglichkeiten:

<code>Mitschriften/Vorlesungsname</code>	<code>= {},</code>
<code>Mitschriften/Typ</code>	<code>= {\ "Ubung"},</code>
<code>Mitschriften/LfdNr</code>	<code>= {},</code>
<code>Mitschriften/Gruppe</code>	<code>= {},</code>
<code>Mitschriften/Headerhoehe</code>	<code>= {42pt},</code>

Die erste Klammer „Vorlesungsname“ legt den Oberbegriff für die Mitschriften fest, also etwa den Namen eines Unterrichtsfachs oder einer Firma. An zweiter Stelle steht der Begriff für die Art der Sitzung. Dieser könnte beispielsweise „Hausaufgabe“, „Konferenz“ oder „Regionalkonvent“ lauten. Als drittes trägt man die

2 Funktionen

Nummer der Mitschrift ein. Zusätzlich gibt es die Option, die Nummer der vorliegenden Arbeitsgruppe zu notieren. Als letztes erhält man die Möglichkeit, die Größe des Kopfbereichs anzupassen, wobei die bereits vorliegende Einstellung empfehlenswert ist. Auch hier gibt es ein Beispiel:



Dafür wurden folgende Einstellungen verwendet, wobei für das Bild eine eigene Grafik „titel.jpg“ verwendet wurde:

```
all/Autoren = {Philipp Winniewski&Mirco Lukas},
all/Titel   = {Aufbau eines Zyklotrons},
all/Untertitel = {Das Synchrotron},
all/Icon/Breite = {.05},
all/Version = {true},
Mitschriften/Vorlesungsname = {Physik},
Mitschriften/Typ = {Hausaufgabe},
Mitschriften/LfdNr = {23},
Mitschriften/Gruppe = {13},
```

Hinweis 1: Keine Chapter!

Eine Mitschrift hat keine Kapitel. Daher wird `\chapter` automatisch zu `\section` übersetzt.

2.2.3 Buch

Die Vorlage für ein *Buch* hat das deutliche Merkmal, dass die Seitengröße mit DIN-A5 kleiner ist als bei den zwei vorigen Typen. Das Deckblatt erinnert an jenes eines Skripts und man findet ebenfalls ein Inhaltsverzeichnis. Kapitel beginnen jeweils auf Seiten mit ungerader Seitenzahl und der Kapitelname wird über das gesamte Kapitel hinweg im Kopfbereich angezeigt. Darüber hinaus kann der Seitenrand für gerade und ungerade Seitenzahlen, welche in den Ecken oben am Außenrand dargestellt werden, unterschiedlich sein.

2 Funktionen

Ein solches Buch hat in etwa Taschenbuch-Format. Es eignet sich somit für längere Fließtexte wie z.B. Romane.

Achtung!

Bitte hier an die Beispieldatei `/doc/muster-buch.pdf` halten, da dieser Dokumenttyp 2022 komplett neu überarbeitet wurde.

2.2.4 Praesentation

Die *Praesentation* unterscheidet sich sehr stark von den anderen Typen. Genutzt wird hier die sogenannte *Beamer*-Klasse von L^AT_EX. Dabei handelt es sich um eine Formatierung, die auf Beamer- bzw. Projektor-Präsentationen ausgelegt ist. Die erzeugte PDF wird dann als Vollbild auf die Wand geworfen und Seite für Seite durchgeblättert.

Die hier genutzte Vorlage bietet dafür eine schlichte, jedoch ansehnliche Gestaltung. Es gibt eine Deck-Folie mit Datum, Autor, Titel und Untertitel. Zudem wird auf jeder Seite der Autor und die aktuelle Foliennummer unten angezeigt. Nach der zweiten Folie, dem Inhaltsverzeichnis, erhalten alle folgenden Folien eine Kopfleiste mit dem jeweiligen Folientitel.

Diese Einstellungen können mit der Vorlage gemacht werden:

<code>Praesentationen/TitelKurz</code>	<code>= {}</code> ,
<code>Praesentationen/Institut</code>	<code>= {}</code> ,
<code>Praesentationen/InstitutKurz</code>	<code>= {}</code>

2.3 Neue Befehle und Abkürzungen

Die L^AT_EX-Vorlage bietet eine Vielzahl neuer Befehle, die es eigentlich so in L^AT_EX nicht gibt bzw. anders umschrieben werden müssen. Teilweise müsste man sehr langen Code nutzen, um zu bestimmten Ergebnissen zu kommen. Dies zu vermeiden ist das Anliegen der folgenden neuen Befehle, die in jeder Dokumentart verwendet werden können.

2.3.1 Befehle für Fließtexte

Sehr oft werden Zeichenabstände falsch gewählt. Die meisten Menschen wissen gar nicht, dass es etwa unterschiedliche Längen von Leerzeichen gibt. Da sich aber auch die meisten Menschen nicht damit auseinandersetzen möchten, bietet die Vorlage ein automatisches Setzen von Abkürzungen mit Hilfe leicht zu merkender Befehle,

2 Funktionen

sodass etwa die Buchstaben bei „z. B.“ näher rücken und nicht umgebrochen werden. Außerdem gibt es weitere nützliche Befehle.

Befehl	Ergebnis	Beschreibung
<code>\blitz</code>	⚡	Blitzsymbol, z. B. Widersprüche
<code>\colorref{1}</code>	(i)	Nummerierung für Formeln zur Referenzierung (1 bis 4)
<code>\ellipse</code>	...	Auslassungszeichen
<code>\fallunterscheidung[x=]{2 \fuer 3,}{4 \sonst}</code>	$x = \begin{cases} 1 & \text{für 3,} \\ 4 & \text{sonst} \end{cases}$	Geschweifte Klammer mit zwei Feldern rechts (<code>\fuer</code> und <code>\sonst</code> sind optional)
<code>\length{text}</code>	$ text $	Setzt Text in gerade Linien
<code>\name{Mirco Lukas}</code>	MIRCO LUKAS	Kapitälchen für Namensnennungen
<code>\teilaufgabe{1}</code>	<u>TA 1.1</u>	Überschrift für Teilaufgaben mit Aufgabennummer in Klammer

2 Funktionen

Befehl	Ergebnis	Beschreibung
<code>\textmarker{green}{text}</code>	text	Farbiges Hinterlegen eines Textes, wobei die Farbe auf englisch angegeben wird (sehr viele Farben funktionieren!)
<code>\qed</code>	\square	Beweis-Schlusszeichen bzw. <i>quod erat demonstrandum</i> am Zeilenende
<code>\wiki{https://de.wikipedia.org/wiki/LaTeX}</code>	Wikipedia	Link zu einer Wikipedia-Seite, wobei als Text nur „Wikipedia“ angezeigt wird
<code>\dhe</code>	d. h.	„das heißt“
<code>\Dhe</code>	D. h.	„Das heißt“
<code>\idr</code>	i. d. R.	„in der Regel“
<code>\Idr</code>	I. d. R.	„In der Regel“
<code>\obda</code>	o. B. d. A.	„ohne Beschränkung der Allgemeinheit“
<code>\Obda</code>	O. B. d. A.	„Ohne Beschränkung der Allgemeinheit“
<code>\sio</code>	s. o.	„siehe oben“
<code>\Sio</code>	S. o.	„Siehe oben“
<code>\siu</code>	s. u.	„siehe unten“
<code>\Siu</code>	S. u.	„Siehe unten“
<code>\ua</code>	u. a.	„unter anderem“
<code>\Ua</code>	U. a.	„Unter anderem“
<code>\va</code>	v. a.	„vor allem“
<code>\Va</code>	V. a.	„Vor allem“
<code>\zb</code>	z. B.	„zum Beispiel“
<code>\Zb</code>	Z. B.	„Zum Beispiel“
<code>\zt</code>	z. T.	„zum Teil“
<code>\Zt</code>	Z. T.	„Zum Teil“
<code>\href{http://google.de}{Google}</code>	Google	Link mit selbst definiertem, angezeigtem Text in der zweiten Klammer

Die Abkürzungen existieren für Positionen mitten im Satz und für Satzanfänge. Die Versionen für Satzanfänge besitzt als ersten Buchstaben des Befehls auch einen Großbuchstaben, um sich dies leichter merken zu können.

2.3.2 Befehle für mathematische Umgebungen

In L^AT_EX gibt es sogenannte *Mathematische Umgebungen*. Diese erlauben andere Befehle als sonst und ändern auch die Formatierungen. Eine Mathematische Umgebung beginnt und endet mit einem Dollarzeichen „\$“. Innerhalb dieser Zeichen werden mathematische Formeln und Sonderzeichen eingefügt. Folgende Befehle stellt die Vorlage zusätzlich zu denen von L^AT_EX für solche Textabschnitte zur Verfügung:

Befehl	Ergebnis	Beschreibung
<code>\andd</code>	\wedge	Konjunktionszeichen
<code>\bigandd</code>	\bigwedge	Großes Konjunktionszeichen
<code>\bigorr</code>	\bigvee	Großes Disjunktionszeichen
<code>\orr</code>	\vee	Disjunktionszeichen
<code>\epsilon</code>	ε	Alternatives Epsilon
<code>\epsilonOld</code>	ϵ	Standard-Version für Epsilon
<code>\phi</code>	φ	Alternatives Phi
<code>\phiOld</code>	ϕ	Standard-Version für Phi

2.4 Begriffsdefinitionen

In wissenschaftlichen Arbeiten kommt es häufig vor, dass eine Reihe fachlicher Begriffe genutzt, eingeführt, erklärt oder definiert wird. In vielen Fällen möchte man diese Wörter oder Wortkombinationen hervorheben und am Ende des Werks alphabetisch auflisten. Hierfür bietet die L^AT_EX-Vorlage unterstützende Funktionen.

Es gibt folgende drei Befehle für neu im Werk erscheinende Begriffe:

```
\neuerbegriff{}
\nueurbegriffIdx{}
\nueurbegriffIdx[]{}

```

Alle drei Befehle setzen den Inhalt der jeweils hintersten Klammer in ein rotes Kästchen und hinterlegt es leicht grau. Die letzten beiden Befehle ermöglichen zudem das Eintragen des Begriffs in den dazugehörigen Index. Mit dem letzten Befehl kann man die Auflistung des Begriffs im Index mit einer alternativen Bezeichnung durchführen, wobei die Index-Bezeichnung in die eckige Klammer geschrieben wird.

Den Index bzw. die Auflistung der Begriffe im Anhang des Dokuments aktiviert und deaktiviert man im Kopfbereich der L^AT_EX-Datei mit der Einstellung

2 Funktionen

```
all/Index/Begriffe/titel      = {Index der Begriffe},  
all/Index/Begriffe/zeigen    = {true},
```

Dabei gibt man den Titel des Anhangs für Begriffe in der ersten Klammer an. Das Ein- und Ausschalten funktioniert mit der zweiten Klammer, indem man „true“ oder „false“ hineinschreibt. Es folgt ein Beispiel:

```
Ein \neuerbegriff{Autodidakt} ist ein Mensch, der selbstständig lernt. \\  
Ein \neuerbegriffIdx{Lehrer} ist ein Mensch, der anderen Menschen oder  
Tieren etwas lehrt. \\  
Ein \neuerbegriffIdx[Auto, Automobil]{Auto} ist ein aus eigenem Antrieb  
heraus fahrendes Transportmittel mit vier Rädern.
```

Das Ergebnis sieht so aus:

Ein Autodidakt ist ein Mensch, der selbstständig lernt.
Ein Lehrer ist ein Mensch, der anderen Menschen oder Tieren etwas lehrt.
Ein Auto ist ein aus eigenem Antrieb heraus fahrendes Transportmittel mit vier Rädern.

Und der Index dazu am Ende der Arbeit hat diese Gestalt (Seitenausschnitt):



2.5 Beispiele und Beweise

Wissenschaftliche Arbeiten erfordern meist die Nennung von Beispielen und Beweisen. Für diese Fälle gibt es spezielle Umgebungen, die diese Vorlage realisiert. Somit werden diese Text-Ausschnitte angemessen dargestellt. Auch hier existieren drei Befehle:

2 Funktionen

```
\begin{beispiel}  
\end{beispiel}
```

```
\begin{beispiele}  
\end{beispiele}
```

```
\begin{beweis}  
\end{beweis}
```

Jeweils **begin** und **end** rahmen einen solchen Abschnitt ein. Er beginnt immer mit einer fett geschriebenen Überschrift, also „Beispiel.“ oder „Beweis.“. Möchte man diese Überschrift präzisieren, so muss man hinter den Beginn-Befehl eine eckige Klammer mit der entsprechenden Erweiterung der Überschrift einsetzen. Hier einige Beispiele:

```
\begin{beispiel}  
Das hier ist ein Beispiel.  
\end{beispiel}
```

```
\begin{beispiele}  
Das hier sind mehrere Beispiele.  
\end{beispiele}
```

```
\begin{beispiel}[zur Nutzung von Lautsprechern]  
Mit Lautsprechern kann man seine Nachbarn ärgern.  
\end{beispiel}
```

```
\begin{beweis}  
Ist halt so.  
\end{beweis}
```

```
\begin{beweis}[für die Notwendigkeit von Speiseeis]  
Dazu muss man nichts sagen!  
\end{beweis}
```

Dieser Code erzeugt folgende Abschnitte:

Beispiel.

Das hier ist ein Beispiel.

Beispiele.

Das hier sind mehrere Beispiele.

Beispiel zur Nutzung von Lautsprechern.

Mit Lautsprechern kann man seine Nachbarn ärgern.

Beweis.

Ist halt so.

□

Beweis für die Notwendigkeit von Speiseeis.

Dazu muss man nichts sagen!

□

Es kann sich hier leider ein Problem ergeben. Verwendet man eine andere Umgebung innerhalb der gerade vorgestellten Umgebungen, so entstehen große Abstände zwischen Überschrift und Inhalt. Bei Beweisen entsteht zudem ein Abstand zwischen Inhalt und Beweis-Schlusszeichen. Für diese Fälle kann man eine zweite eckige Klammer setzen mit dem Inhalt „o“ für „oben enger“, „u“ für „unten enger“ und „b“ für „beides enger“. Auch das soll am Beispiel gezeigt werden:

```
\begin{beweis}[für meine Theorie]
\begin{itemize}
\item Ich habe Recht.
\item Du hast Unrecht.
\end{itemize}
\end{beweis}
```

```
\begin{beweis}[für meine Theorie][b]
\begin{itemize}
\item Ich habe Recht.
\item Du hast Unrecht.
\end{itemize}
\end{beweis}
```

Dieser Code erzeugt folgende Abschnitte:

Beweis für meine Theorie.

- Ich habe Recht.
- Du hast Unrecht.

□

Beweis für meine Theorie.

- Ich habe Recht.
- Du hast Unrecht.

□

2.6 Aufzählungen

Aufzählungen existieren auch ohne die Vorlage in \LaTeX und können z. B. über den Befehl `\begin{enumerate}` usw. eingefügt werden. Allerdings sind Optionen für alternative Aufzählungsformate kompliziert zu realisieren und man kann sie sich nicht so leicht merken. Genau für diesen Fall bietet die \LaTeX -Vorlage vier neue Aufzählungsumgebungen; eine für Listen mit Buchstaben und eine für Römische Zahlen (jeweils eine Version für große und kleine Zeichen). Die Umgebungen heißen „`enumeratealpha`“, „`enumerateAlpha`“, „`enumerateroman`“ und „`enumerateRoman`“.

Die normale Aufzählung soll mit den vier neuen Formaten im folgenden Beispiel verglichen werden:

```
\begin{enumerate}
\item Eins
\item Zwei
\item Drei
\end{enumerate}
```

```
\begin{enumeratealpha}
\item Eins
\item Zwei
\item Drei
\item Vier
\item Fünf
\end{enumeratealpha}
```

2 Funktionen

```
\begin{enumerateAlpha}  
\item Eins  
\item Zwei  
\end{enumerateAlpha}
```

```
\begin{enumerateroman}  
\item Eins  
\item Zwei  
\item Drei  
\item Vier  
\item Fünf  
\end{enumerateroman}
```

```
\begin{enumerateRoman}  
\item Eins  
\item Zwei  
\end{enumerateRoman}
```

Dieses Ergebnis erzeugt der obige Code:

- 1. Eins
- 2. Zwei
- 3. Drei
- a) Eins
- b) Zwei
- c) Drei
- d) Vier
- e) Fünf
- A) Eins
- B) Zwei

- i) Eins
- ii) Zwei
- iii) Drei
- iv) Vier
- v) Fünf
- I) Eins
- II) Zwei

2.7 Zitate

Auch für Zitate hat L^AT_EX von Haus aus zwei Umgebungen, nämlich „quote“ (für kurze Zitate oder Phrasen) und „quotation“ (für längere Zitate mit mehr als einem Absatz). Die Vorlage erweitert die Zitat-Funktion mit sinnvollen Mechanismen. Die Umgebungen heißen dann „quotex“ und „quotationx“, also mit einem „x“ hinter dem regulären Namen. Dann gibt es zwei neue Funktionen: zum einen kann man leicht die Quelle bzw. den Autor des Zitats nennen, zum anderen kann man zwischen vier verschiedenen Schriftarten wählen. Als Fonts stehen Standard-, Schreibmaschinen-, Hand- und Frakturschrift zur Auswahl. Auf diese Weise kann man bei jedem Zitat für die passende Gestalt sorgen.

Die Codes lauten:

```
\begin{quotex}[] []
\end{quotex}
```

```
\begin{quotationx}[] []
\end{quotationx}
```

Die erste eckige Klammer enthält den Ursprung, die Quelle oder den Autor des Zitats.

Man schreibt in die zweite eckige Klammer die Abkürzung der Schriftart. „tt“ steht hier für „typewriter“ (Schreibmaschine) und „fraktur“ für Frakturschrift. Möchte man das Zitat jedoch in normaler Schrift belassen, lässt man diese Klammer einfach leer.

Für alle vier Schriftbilder folgen nun Beispiele mit „quotex“. Die Umgebung „quotationx“ funktioniert natürlich analog dazu.

```
\begin{quotex}[Helmut Kohl] []  
Entscheidend ist, was hinten rauskommt.  
\end{quotex}
```

```
\begin{quotex}[Helmut Kohl] [tt]  
Entscheidend ist, was hinten rauskommt.  
\end{quotex}
```

```
\begin{quotex}[Helmut Kohl] [fraktur]  
Entscheidend ist, was: hinten raus:kommt.  
\end{quotex}
```

Und dazu das Ergebnis:

Entscheidend ist, was hinten rauskommt.	— <i>Helmut Kohl</i>
Helmut Kohl Entscheidend ist, was hinten rauskommt.	— <i>Helmut Kohl</i>
Entscheidend ist, was hinten rauskommt.	— <i>Helmut Kohl</i>

2.8 Boxen

Eine besondere Fähigkeit der Vorlage ist das Bereitstellen farbiger Kästchen zur Hervorhebung bestimmter Teile des Werks und zur Auflistung der jeweiligen Titel im Anhang. Man unterscheidet hier zwischen normalen Boxen und Warnungsboxen.

2.8.1 Farbige Kästchen

Für Beispiele, Hinweise, Definitionen usw. kann man farbige Kästchen einsetzen. Die jeweilige Farbe ordnet man für sich einer inhaltlichen Kategorie zu, damit gleiche Farben auch das Gleiche bedeuten. Zur Auswahl stehen Blau, Gelb und Grün. Ähnlich wie bei Begriffshervorhebungen können hier jeweils Kästchen einer Farbe zusammengefasst und als Auflistung im Anhang gezeigt werden. Die Einstellung erfolgt im Kopfbereich des Dokuments:

```
all/Index/Boxen/blau/titel = {Index 1},
```

2 Funktionen

```
all/Index/Boxen/blau/zeigen      = {false},
all/Index/Boxen/gelb/titel       = {Index 2},
all/Index/Boxen/gelb/zeigen      = {true},
all/Index/Boxen/gruen/titel      = {Index 3},
all/Index/Boxen/gruen/zeigen     = {false},
```

In die Klammern hinter `all/Index/Boxen/.../titel` schreibt man den Titel für den entsprechenden Index im Anhang für die genannte Farbe. In der Zeile darunter hinter `all/Index/Boxen/.../zeigen` schreibt man in die Klammer entweder „true“, wenn der Index für diese Farbe erstellt werden soll, oder „false“ falls nicht. Im Code oben würde demnach nur ein Anhang für gelbe Kästchen erstellt werden, und zwar mit der Überschrift „Index 2“ und der Auflistung der Titel aller gelben Kästchen.

Der Code für die Boxen selber sieht wie folgt aus:

```
\begin{blueboxIdx}[]{}
\end{blueboxIdx}
```

Der Umgebungsname lautet für die blaue Box „blueboxIdx“ und für Gelb und Grün entsprechend „yellowboxIdx“ bzw. „greenboxIdx“. Die eckige Klammer trägt optional einen L^AT_EX-Label zur Referenzierung, auf welchen über `\ref` oder `\vref` zugegriffen werden kann. Die geschweifte Klammer beinhaltet den Titel der Box. In dieser Klammer kann per Befehl `\boxnummer` auf die fortlaufende Nummerierung der Kästchen zugegriffen werden. Somit passt sich die Nummer automatisch an, falls Boxen dazukommen oder entfernt werden.

Ein Beispiel soll dies alles zeigen:

```
\begin{blueboxIdx}{Definition \boxnummer: Innenwinkel eines Polygons}
Die Innenwinkel eines Polygons sind in der Geometrie die Winkel,
die durch zwei benachbarte Polygonseiten eingeschlossen werden
und in Inneren des Polygons liegen. Die Ecken des Polygons
bilden dabei die Scheitelpunkte der Innenwinkel. Jedes $n$-Eck
besitzt genau $n$ Innenwinkel.
\end{blueboxIdx}
```

```
\begin{yellowboxIdx}{Satz \boxnummer: Innenwinkelsumme im Dreieck}
Die Summe der Innenwinkel eines Dreiecks ist stets 180 Grad.
\end{yellowboxIdx}
```



```
\begin{greenboxIdx}{Beispiel \boxnummer: Innenwinkel eines  
gleichseitigen Dreiecks}  
Da alle Seiten gleich lang sind, sind auch alle Winkel  
gleich groß. Damit ergibt sich folgende Rechnung:  
\begin{equation}  
180 / 3 = 60  
\end{equation}  
\end{greenboxIdx}
```

Das Resultat:

Definition 2: Innenwinkel eines Polygons

Die Innenwinkel eines Polygons sind in der Geometrie die Winkel, die durch zwei benachbarte Polygonseiten eingeschlossen werden und in Inneren des Polygons liegen. Die Ecken des Polygons bilden dabei die Scheitelpunkte der Innenwinkel. Jedes n -Eck besitzt genau n Innenwinkel.

Satz 3: Innenwinkelsumme im Dreieck

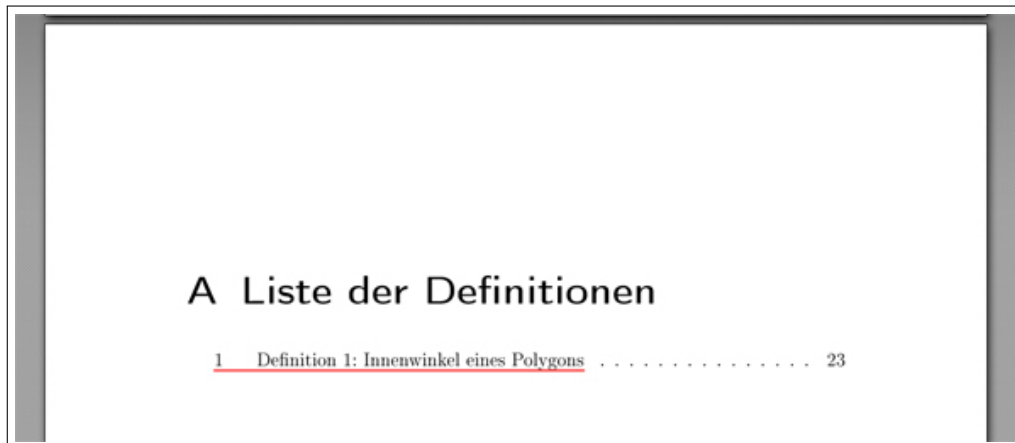
Die Summe der Innenwinkel eines Dreiecks ist stets 180 Grad.

Beispiel 4: Innenwinkel eines gleichseitigen Dreiecks

Da alle Seiten gleich lang sind, sind auch alle Winkel gleich groß. Damit ergibt sich folgende Rechnung:

$$180/3 = 60 \quad (2.1)$$

Aktiviert man nun den Index mit dem Titel „Liste der Definitionen“ (im Kopfbereich des Dokuments) für gelbe Boxen, gäbe es folgende Seite für das obige Beispiel im Anhang:



A Liste der Definitionen

1 Definition 1: Innenwinkel eines Polygons 23

Hinweis 2: Indices mit TeXstudio

Möchte man einen oder mehrere Indices in sein Dokument einfügen, dann muss direkt vor der Berechnung des PDFs der Menüpunkt Tools/Index in der Menüleiste von TeXstudio angeklickt werden. Auch, wenn man den Index aktualisieren möchte, muss dieser Schritt durchgeführt werden.

2.8.2 Warnungskästchen

Zusätzlich zu diesen Farben gibt es noch rote Kästchen, die für Warnungen gedacht sind. Diese tragen als Titel stets „Achtung!“ und können nicht in einem Index aufgelistet werden. Hier das Beispiel:

```
\begin{redbox}  
Man sollte nie unhöflich sein!  
\end{redbox}
```

