

L^AT_EX-Einführungskurs

P. Müller, M. Giger

26. März 2018

Inhaltsverzeichnis

1	Voraussetzungen	2
2	T_EXstudio	2
2.1	Benutzeroberfläche	2
3	L^AT_EX	4
3.1	Commands und Enviroments	4
3.2	L ^A T _E X-Dokument einbinden	5
3.3	Überschriften und Abschnitte	5
3.4	Aufzählungen	5
3.5	Mathematikumgebung	6
3.6	Bilder einfügen	7
3.7	Tabellen	7
3.8	Minipages	8
3.9	Multicols	8
3.10	Schriftgrössen und Formate	8
3.11	PDFs und Code einbinden	9
3.12	Weitere Tutorials	10
A	Lösung	11
B	L^AT_EX_{2_ε} Cheat Sheet	12

1 Voraussetzungen

Folgendes wird für diesen Kurs vorausgesetzt:

1. Vorlage-Projekt „VorlageZF“ auf Rechner.
2. Eine Version von T_EXstudio installiert.

Sämtliche Aufgaben und Anpassungen werden innerhalb des Vorlage-Projekts vorgenommen.

2 T_EXstudio

T_EXstudio ist ein plattformunabhängiger Editor, um komfortabel L^AT_EX-Dokumente zu erstellen. Es ist einfach zu bedienen und richtet sich vor allem an die Zielgruppe L^AT_EX-Einsteiger. Im Vergleich zu anderen L^AT_EX-Editoren beweist sich T_EXstudio als eines der Programme mit den umfangreichsten Funktionen. Zudem gibt es sowohl Versionen für Linux, als auch für Windows und Mac OS. Durch die Plattformunabhängigkeit können eine Reihe an externen Programmen integriert werden, darunter Rechtschreibüberprüfungen in verschiedenen Sprachen. Zudem bietet es eine rundum Hilfestellung, das Aufspüren und Beseitigen von Fehlern und eine sofortige Vorschau.

2.1 Benutzeroberfläche

Die Oberfläche besteht aus einer Symbolleiste, einem Editor und einer Konsole, die zusammengefasst in einem großen Hauptfenster erscheinen. Dieses Konzept verleiht der Software ein recht homogenes Aussehen, wirkt aber unter bestimmten Umständen – etwa auf kleinen Bildschirmen – schnell unübersichtlich. Neben Menü und Werkzeugleiste finden sich im Fenster mehrere Ein- und Ausgabebereiche. Die Werkzeugleisten lassen sich mit der Maus verschieben; einzelne Menüs können über den Konfigurationsdialog unter Menüs ein- oder ausgeblendet werden.

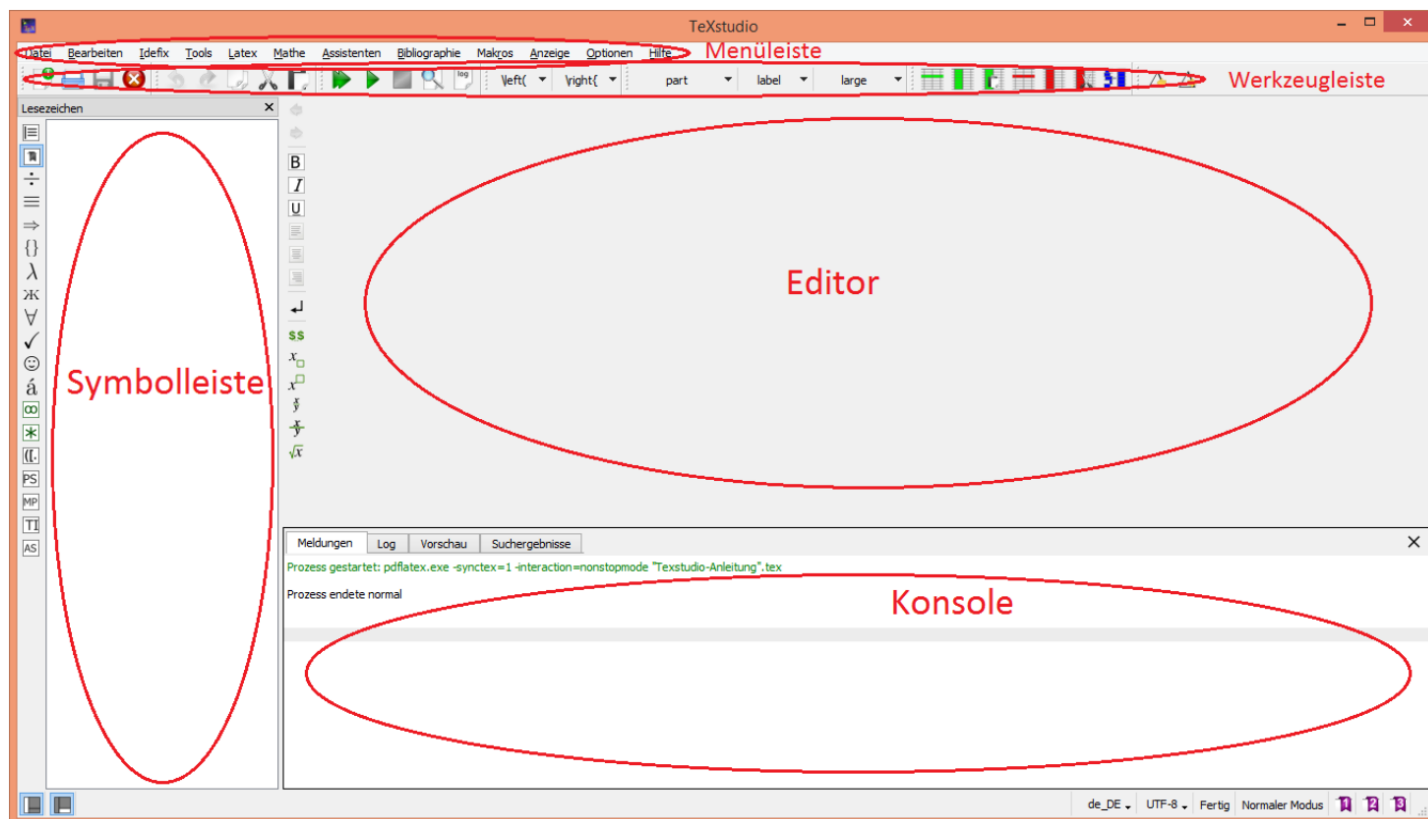


Abbildung 1: Benutzeroberfläche T_EXstudio

- **Editor:** Hier können die L^AT_EX-Dokumente bearbeitet werden.
- **Konsole:** Hier werden etwaige Fehlermeldungen beim Kompilieren ausgegeben.
- **Symbolleiste:** Hier wird die interne Hierarchie des aktuell bearbeiteten Files angezeigt.

Das Aussehen des Editors kann unter *Optionen > T_EXstudio konfigurieren* im Detail eingestellt werden. Unter *Editor* legt man das grundlegende Verhalten fest. Im Dialog *Erweiterter Editor* die Feinheiten, wie Breite des Tabulators, die Suchfunktion oder den automatischen Zeilenumbruch.

► **Aufgabe 1: Neues Dokument erstellen**

Erstelle ein neues Dokument „*Tutorial.tex*“ und speichere es im Unterordner „*sections*“.
(Später werden wir dieses Dokument ins Hauptdokument einfügen.)

Für eine detailliertere Einführung in T_EXstudio : [T_EXstudio -Einführung](#)

3 L^AT_EX

Die Alternative zum Schreiben wissenschaftlicher Dokumente heißt L^AT_EX. Es ist eine Erweiterung zum Textsatzsystem T_EX, welches in den 70er Jahren von Donald E. Knuth entwickelt wurde und seitdem gereift ist.

Funktionsweise von L^AT_EX

Beim Arbeiten mit L^AT_EX werden Text und Formatierungsbefehle klar lesbar in einem reinen Text-File gespeichert. Man kann somit mit einem kleinen, aber schnellen Texteditor arbeiten und braucht kein riesiges Programm wie Word zu bemühen. Um das Dokument betrachten zu können, wird aus dem Textfile mit Hilfe des Compilers ein druckfertiges Dokument erzeugt, z. B. PDF, DVI, auch HTML-Code ist möglich. L^AT_EX ist case sensitive.

Vorteile von L^AT_EX

- Es ist kostenlos.
- Es zwingt den Autor zum strukturierten Schreiben. Das Layout wird von L^AT_EX professionell in Buchdruckqualität gesetzt.
- Es läuft auf nahezu allen Plattformen. Ein Dokument sieht auf jeder Plattform und jedem Rechner gleich aus, egal welcher Drucker angeschlossen ist.
- Es läuft zuverlässig und absturzfrei.
- Komplizierte Strukturen wie: Fußnoten, Querverweise, Literatur-, Tabellen-, Abbildungsverzeichnisse, Indexe, etc. sind sehr einfach zu erzeugen.

3.1 Commands und Enviroments

Commands sind spezielle Angaben in der Eingabedatei, die Signale an L^AT_EX übermitteln. Diese Signale können unterschiedliches bewirken. Sie können beispielsweise:

- Änderungen am Format des Dokuments auslösen
- Inhaltliche Ausgaben erzeugen
- Informationen für später speichern
- LaTeX-Funktionen erweitern

L^AT_EX-Commands zeichnen sich im Regelfall durch einen vorangestellten \ (Backslash) aus (z.B. `\documentclass`). Darauf folgen unter Umständen mehrere optionale oder verpflichtende *Parameter*. Verpflichtende Parameter sind dabei in geschwungenen, optionale in eckigen und Zahlenangaben manchmal in runden Klammern eingeschlossen. Die Anzahl und Anordnung der Parameter ist beliebig und kann für jeden Befehl anders sein. Ein typischer Command sieht etwa wie folgt aus:

```
\befehlname[optionaleParameter]{verpflichtendeParameter} % Fiktiver Befehl
```

Mit % können Kommentare erzeugt werden.

Es gibt aber auch sogenannte *Umgebungen*. Umgebungen werden mit `\begin` eingeleitet und enden mit `\end`. Umgebungen bewirken ein spezielles Verhalten innerhalb ihrer Grenzen.

```
\begin{umgebungsname}
...
\end{umgebungsname}
```

Viele Umgebungen können ineinander verschachtelt werden. Das merkt man schon daran, dass alle Inhalte in die Umgebung document eingepasst werden. Wir sehen das beispielsweise auch später bei den Aufzählungen. Allerdings müssen hier gewisse Regeln eingehalten werden: wird eine Umgebung mit `\begin` gestartet und werden daraufhin weitere Umgebungen mit `\begin` gestartet, so kann die ursprüngliche, äußere Umgebung erst mit `\end` geschlossen werden, wenn auch die inneren Umgebungen bereits geschlossen wurden.

3.2 L^AT_EX-Dokument einbinden

Auf Grund der Übersichtlichkeit rät es sich, ein L^AT_EX-Dokument in mehrere Teildokumente zu unterteilen. Es gibt zwei Befehle um .tex-Dokumente in ein anderes Dokument einzufügen. Nämlich `\include` und `\input`.

```
\include{pfad/dokument1}
\input{pfad/dokument2}
```

`\include` und `\input` haben sehr unterschiedliche Aufgaben.

`\input` lädt die Datei an Ort und Stelle in die Ziel-Datei und ist als ob man den Text im File direkt in die Ziel-Datei geschrieben hätte. Es kann letztlich überall und für jede Art Datei verwendet werden. Es kann auch verschachtelt angewendet werden, d.h. eine eingebundene Datei kann ihrerseits Dateien mit `\input` einbinden.

`\include` hingegen beginnt eine neue Seite, bevor es `\input` ausführt. Includes sind nicht verschachtelbar. Eine so eingebundene Datei kann aber natürlich `\input` enthalten. `\include` erzeugt eine neue aux-Datei für die eingebundene Datei. Das erlaubt es beispielsweise, ein Dokument in mehrere logische Einheiten zu zerlegen (etwa einzelne Kapitel), die jede einer Datei entsprechen, die mit `\include` in die Hauptdatei eingebunden wird.

► Aufgabe 2: Dokument einbinden

Binde das Dokument „*Tutorial.tex*“ in das Hauptdokument ein. (Dokumentpfad ist relativ anzugeben)

3.3 Überschriften und Abschnitte

In L^AT_EX ist das Erzeugen einer Überschrift mit dem Einleiten eines neuen Abschnitts im Ausgabedokument verknüpft. Abschnitte lassen sich hierarchisch unterteilen. Standardmäßig stehen die folgenden Ebenen (hierarchisch absteigend) zur Verfügung:

```
\section{Kapitel}
\subsection{Unterkapitel}
\subsubsection{Unterunterkapitel}
```

1 Kapitel

1.1 Unterkapitel

1.1.1 Unterunterkapitel

In anderen Dokumentklassen gibt es z.B. noch `\chapter` und `\part` als übergeordnete Ebenen. L^AT_EX erzeugt automatisch eine richtig verschachtelte Nummerierung. Ausschlaggebend dafür ist ausschließlich die Reihenfolge und Tiefe, in der die Befehle angegeben werden.

► Aufgabe 3: Kapitel einfügen

Füge im Dokument „*Tutorial.tex*“ eine neue *Section* hinzu.

3.4 Aufzählungen

Für Aufzählungen stehen unter L^AT_EX die Umgebungen `itemize` und `enumerate` zur Verfügung. Selbstverständlich können Aufzählungen auch ineinandergeschachtelt werden. Durch den Befehl `\item` wird ein neues Aufzählungsobjekt erzeugt. Die Aufzählungspunkte sind je nach Tiefe der Aufzählung durch verschiedene Symbole (Punkt, Bindestrich, Sternchen, kleiner Punkt) gekennzeichnet. Mehr als vier Ebenen sind nicht vorgesehen.

```
\begin{itemize}
  \item Punkt Ja
  \item Punkt Nein
\end{itemize}
```

- Punkt Ja
- Punkt Nein

Es sind auch nummerierte Aufzählungen mit `enumerate` möglich.

```
\begin{enumerate}
  \item Element 1
  \item Element 2
\end{enumerate}
```

1. Element 1
2. Element 2

► Aufgabe 4: Aufzählungen erstellen

Erstelle folgende Aufzählung:

- Argument 1
- Argument 2
 1. Subargument 1
 2. Subargument 2
- Argument 3

3.5 Mathematikumgebung

Mathematische Ausdrücke lassen sich mit L^AT_EX sehr gut formulieren und darstellen. L^AT_EX beherrscht von sich aus diverse Mathematik-Operatoren. Die *American Mathematical Society* (AMS) hat allerdings hervorragende Erweiterungspakete entwickelt. Es gibt unterschiedliche Möglichkeiten, um in L^AT_EX eine Gleichung in ein Dokument einzufügen. Einerseits bieten sich die `equation`-Umgebung, `\[ \]` oder für *inline*-Ausdrücke `$ $`.

Mit der `equation`-Umgebung lassen sich Gleichungen in einem neuen Absatz einfügen. Alle Formelzeichen werden im `\displaystyle`, d.h. sie werden mit grossen Symbolen dargestellt. Zusätzlich erhalten die Gleichungen noch eine Nummer, die an der rechten Seite angezeigt wird.

```
\begin{equation}
  \oint \vec{B} \cdot \vec{dA} = 0
\end{equation}
```

$$\oint \vec{B} \cdot d\vec{A} = 0 \quad (1)$$

Mit `\[ \]` lassen sich ebenfalls Gleichungen in einem neuen Absatz einfügen. Allerdings werden diese nicht nummeriert. Auch hier werden Zeichen im `\displaystyle` dargestellt.

```
\[ u_i = \pm \frac{d\Phi}{dt} \]
```

$$u_i = \pm \frac{d\Phi}{dt}$$

Mit `$ $` können Gleichungen *inline* geschrieben werden. Das bedeutet Formeln können zusammen mit Text auf derselben Linie dargestellt werden.

```
Kreis: $A=r^2\cdot\pi$
```

$$\text{Kreis: } A = r^2 \cdot \pi$$

► Aufgabe 5: Physikalisches Gaußsches Gesetz

Versuche folgenden mathematischen Ausdruck zu schreiben:

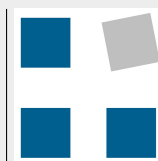
$$\operatorname{div} \vec{D} = \vec{\nabla} \cdot \vec{D} = \rho \Leftrightarrow \oiint_{\partial V} \vec{D} \cdot d\vec{A} = \iiint_V \rho \cdot dV = Q(V)$$

Experimentiere auch mit der *inline*-Mathematikumgebung, um die unterschiedlichen Darstellungen zu sehen.

3.6 Bilder einfügen

In einem Dokument möchte man häufig Grafiken und Bilder einfügen. Hierfür bietet L^AT_EX den Befehl `\includegraphics`. Das einzufügende Bild sollte sich irgendwo im selben Ordner, oder noch besser in einem Subordner des Hauptdokuments befinden.

```
\includegraphics[width=2cm]{images/HSRLogo}
```



3.7 Tabellen

Für Tabellen bieten sich in L^AT_EX unterschiedliche Umgebungen an. Wichtige Umgebungen sind: `tabular`, `tabularx`, `tabulary` und `longtable`. Mit `\hline` können horizontale Linien gezogen werden.

- **tabular**: Tabellenumgebung bei der die Tabellengröße dem Inhalt angepasst wird.
- **tabularx**: Tabellen bei der die Breite der Tabelle vorgegeben werden kann. Zusätzlicher Spaltentyp X.
- **tabulary**: Erweiterung von tabularx. Bietet 4 weitere Spaltentypen L,R,C,J.
- **longtable**: Tabelle die über mehrere Seiten verteilt werden kann.

```
\begin{tabular}{|l|c|r|}
\hline
aaa & bbb & ccc \\
\hline
a & b & c \\
\hline
\end{tabular}
```

aaa	bbb	ccc
a	b	c

In `{|l|c|r|}` werden die Spalten angegeben. Folgende Spaltentypen sind standardmässig definiert.

- |: Trennstrich zwischen Spalten
- l: Linksbündig
- c: Zentriert
- r: Rechtsbündig
- p{breite}: Minipage mit angegebener Breite, Ausrichtung oben
- m{breite}: Minipage mit angegebener Breite, Ausrichtung mittig
- b{breite}: Minipage mit angegebener Breite, Ausrichtung unten
- X: Spalte mit variabler Breite (tabularx)

► Aufgabe 6: Tic Tac Toe

Versuche mittels `tabular` ein Tic Tac Toe-Spiel zu zeichnen.

Bsp:

x	o	o
x	o	x
x	x	o

3.8 Minipages

Die `minipage` wird oft genutzt um Dinge nebeneinander setzen zu können, die sich sonst nicht so einfach nebeneinander setzen lassen. Zum Beispiel zwei Bilder nebeneinander, zwei Tabellen oder ein Bild neben einen Text beziehungsweise umgekehrt. Die Idee hinter der `minipage`-Umgebung ist, dass innerhalb einer bestehenden Seite eine weitere Seite eingebaut wird. Dadurch hat man nun die Möglichkeit diese neue Seite zu verwenden oder eine neue *Minipage* danebenzusetzen (z.B. um zwei Bilder nebeneinander darzustellen).

```
\begin{minipage}[h]{3cm}
  Minipage 1
\end{minipage}
\begin{minipage}[h]{3cm}
  \includegraphics[width=2cm]{images/HSRLogo}
\end{minipage}
```

Minipage 1



3.9 Multicols

Die `multicols`-Umgebung erlaubt, Teile eines Dokuments mehrspaltig zu setzen. Dabei regelt die Umgebung automatisch den Umbruch zur nächsten Spalte.

```
\setlength{\columnsep}{0.5cm} %Abstand Spalten
\setlength{\columnseprule}{0.2pt} %Strichdicke
\begin{multicols}{2}
  \input{sources/text}
\end{multicols}
```

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit	amet, laoreet vitae, arcu. Aenean faucibus pede eu ante.
---	--

3.10 Schriftgrößen und Formate

Die Schriftgröße einzelner Wörter, Abschnitte oder des ganzen Textes können geändert werden. Dafür bieten sich die folgenden Commands (in der Grösse aufsteigend) an :

```
{\tiny A} {\small A} {\large A}
{\Large A} {\LARGE A} {\huge A} {\Huge A}
```

A A A A A A A

Zudem kann Text **fett**, *kursiv*, unterstrichen oder auch **farbig markiert** werden.

```
\textbf{A} \textit{A}
\underline{A} {\color{red}A}
```

A A A A

3.11 PDFs und Code einbinden

Um PDFs in ein Dokument einzubinden wird zusätzlich zum Vorlage-Header noch das Package `pdfpages` benötigt. Mit dem Befehl `\includepdf` wird dann das PDF hinzugefügt. Der Beispielcode bindet alle Seiten von „ELT3.pdf“ ein.

```
\includepdf[pages=-]{sources/ELT3}
```

Es lassen sich aber auch Codefiles unterschiedlicher Programmiersprachen (z.B. C, C++, Java, Python, etc.) einbinden. Dafür kann die `lstlisting`-Umgebung oder `\lstinputlisting` verwendet werden. Mit `\lstdefinestyle` lassen sich unterschiedliche Codeformatierungen erstellen. Hier ein Beispiel für C++:

```
\lstdefinestyle{cpp}{ %Namensdefinition
  backgroundcolor=\color{white}, %Hintergrundfarbe Fenster
  tabsize=2, %Tab = n Spaces
  language=[GNU]C++, %Auswahl Programmiersprache
  basicstyle=\scriptsize, %Grundlegender Schriftstyle
  aboveskip={0.5\baselineskip}, %Abstand zu Rahmen
  columns=fixed, %Fixe Zeichenbreite
  showstringspaces=false, %Anzeigen von Spaces in Strings
  extendedchars=false, %Chars 128-255 enabled
  breaklines=true, %Automatischer Zeilenumbruch
  prebreak = \raisebox{0ex}[0ex][0ex]{\ensuremath{\hookleftarrow}},
  %Anfang von Zeile
  frame=single, %Rahmen um Code
  numbers=left, %Zeilennummern anzeigen
  showtabs=true, %Anzeigen von Tabs
  showspaces=false, %Anzeigen von Spaces
  identifierstyle=\ttfamily, %Format von Identifiern
  keywordstyle=\color{rgb}{0,0.2,0.8}, %Format von Schlüsselwörtern
  commentstyle=\color{rgb}{0.026,0.112,0.095}, %Format von Kommentaren
  stringstyle=\color{rgb}{0.627,0.126,0.941}, %Format von Strings
  numberstyle=\color{rgb}{0.205, 0.142, 0.73}, %Format von Zahlen
}
```

Für genauere Informationen bezüglich dem `listings`-Package empfiehlt sich die [Package-Dokumentation](#).

Hier noch ein Beispiel, wie der formatierte Code schlussendlich aussieht:

```
\lstinputlisting[style=cpp]
{sources/HelloWorld.cpp}
```

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main()
6 {
7     cout << "Hello World" << endl;
8     return 0;
9 }
```

3.12 Weitere Tutorials

Hier noch Links zu weiteren L^AT_EX-Tutorials:

[Youtube-Tutorial](#)

latex.TuGraz.at

latex-tutorial.com (Englisch)

latex.hpfsc.de

fernuni-hagen.de

► **Aufgabe 7: Ausprobieren!**

L^AT_EX lässt sich am besten lernen, indem man ausprobiert und andere Files analysiert. Clone dazu weitere Formelsammlungen von [GitHub](#), schau dir an wie diese erstellt wurden und passe sie nach deinem Geschmack an. Falls Fragen auftauchen, dann versuch dein Problem zu googeln.

Viel Spass beim Lernen!

A Lösung

```
% Vorlage.tex
```

```
...
```

```
\input{sections/Tutorial}
```

```
...
```

```
% Tutorial.tex
```

```
\section{Tutorial (Aufgabe 3)}
```

```
\subsection{Aufzählung (Aufgabe 4)}
```

```
\begin{itemize}
```

```
\item Argument 1
```

```
\item Argument 2
```

```
\begin{enumerate}
```

```
\item Subargument 1
```

```
\item Subargument 2
```

```
\end{enumerate}
```

```
\item Argument 3
```

```
\end{itemize}
```

```
\subsection{Mathematikumgebung (Aufgabe 5)}
```

```
\[ \text{div } \vec{D} = \vec{\nabla} \cdot \vec{D} = \rho \Leftrightarrow
```

```
\oiint_{\partial V} \vec{D} \cdot d\vec{A} = \iiint_V \rho \, dV = Q(V) \]
```

```
$ \text{div } \vec{D} = \vec{\nabla} \cdot \vec{D} = \rho \Leftrightarrow
```

```
\oiint_{\partial V} \vec{D} \cdot d\vec{A} = \iiint_V \rho \, dV = Q(V) $
```

```
\subsection{Tabellen (Aufgabe 6)}
```

```
\begin{tabular}{c|c|c}
```

```
x & o & o \\ \hline
```

```
x & o & x \\ \hline
```

```
x & x & o \\
```

```
\end{tabular}
```

B L^AT_EX 2_ε Cheat Sheet

Document classes

book	Default is two-sided.
report	No \part divisions.
article	No \part or \chapter divisions.
letter	Letter (?).
slides	Large sans-serif font.

Used at the very beginning of a document:
`\documentclass{class}`. Use `\begin{document}` to start contents and `\end{document}` to end the document.

Common documentclass options

10pt/11pt/12pt	Font size.
letterpaper/a4paper	Paper size.
twocolumn	Use two columns.
twoside	Set margins for two-sided.
landscape	Landscape orientation. Must use <code>dvips -t landscape</code> .
draft	Double-space lines.
Usage: <code>\documentclass[opt,opt]{class}</code> .	

Packages

fullpage	Use 1 inch margins.
anysize	Set margins: <code>\marginwidth{l}{r}{t}{b}</code> .
multicol	Use <i>n</i> columns: <code>\begin{multicols}{n}</code> .
latexsym	Use L ^A T _E X symbol font.
graphicx	Show image: <code>\includegraphics[width=x]{file}</code> .
url	Insert URL: <code>\url{http://...}</code> . Use before <code>\begin{document}</code> . Usage: <code>\usepackage{package}</code>

Title

<code>\author{text}</code>	Author of document.
<code>\title{text}</code>	Title of document.
<code>\date{text}</code>	Date.
These commands go before <code>\begin{document}</code> . The declaration <code>\maketitle</code> goes at the top of the document.	

Miscellaneous

<code>\pagestyle{empty}</code>	Empty header, footer and no page numbers.
<code>\tableofcontents</code>	Add a table of contents here.

Document structure

<code>\part{title}</code>	<code>\subsubsection{title}</code>
<code>\chapter{title}</code>	<code>\paragraph{title}</code>
<code>\section{title}</code>	<code>\subparagraph{title}</code>
<code>\subsection{title}</code>	

Use `\setcounter{secnumdepth}{x}` suppresses heading numbers of depth $> x$, where chapter has depth 0. Use a *, as in `\section*{title}`, to not number a particular item—these items will also not appear in the table of contents.

Text environments

<code>\begin{comment}</code>	Comment (not printed). Requires <code>verbatim</code> package.
<code>\begin{quote}</code>	Indented quotation block.
<code>\begin{quotation}</code>	Like quote with indented paragraphs.
<code>\begin{verse}</code>	Quotation block for verse.

Lists

<code>\begin{enumerate}</code>	Numbered list.
<code>\begin{itemize}</code>	Bulleted list.
<code>\begin{description}</code>	Description list.
<code>\item text</code>	Add an item.
<code>\item[x] text</code>	Use <i>x</i> instead of normal bullet or number. Required for descriptions.

References

<code>\label{marker}</code>	Set a marker for cross-reference, often of the form <code>\label{sec:item}</code> .
<code>\ref{marker}</code>	Give section/body number of marker.
<code>\pageref{marker}</code>	Give page number of marker.
<code>\footnote{text}</code>	Print footnote at bottom of page.

Floating bodies

<code>\begin{table}[place]</code>	Add numbered table.
<code>\begin{figure}[place]</code>	Add numbered figure.
<code>\begin{equation}[place]</code>	Add numbered equation.
<code>\caption{text}</code>	Caption for the body.
The <i>place</i> is a list valid placements for the body. t=top, h=here, b=bottom, p=separate page, !=place even if ugly. Captions and label markers should be within the environment.	

Text properties

Font face

Command	Declaration	Effect
<code>\textrm{text}</code>	<code>{\rmfamily text}</code>	Roman family
<code>\textsf{text}</code>	<code>{\sffamily text}</code>	Sans serif family
<code>\texttt{text}</code>	<code>{\ttfamily text}</code>	Typewriter family
<code>\textmd{text}</code>	<code>{\mdseries text}</code>	Medium series
<code>\textbf{text}</code>	<code>{\bfseries text}</code>	Bold series
<code>\textup{text}</code>	<code>{\upshape text}</code>	Upright shape
<code>\textit{text}</code>	<code>{\itshape text}</code>	<i>Italic shape</i>
<code>\textsl{text}</code>	<code>{\slshape text}</code>	<i>Slanted shape</i>
<code>\textsc{text}</code>	<code>{\scshape text}</code>	SMALL CAPS SHAPE
<code>\emph{text}</code>	<code>{\em text}</code>	<i>Emphasized</i>
<code>\textnormal{text}</code>	<code>{\normalfont text}</code>	Document font
<code>\underline{text}</code>		<u>Underline</u>

The command (*tttt*) form handles spacing better than the declaration (*tttt*) form.

Font size

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

These are declarations and should be used in the form `{\small ...}`, or without braces to affect the entire document.

Verbatim text

<code>\begin{verbatim}</code>	Verbatim environment.
<code>\begin{verbatim*}</code>	Spaces are shown as <code>␣</code> .
<code>\verb!text!</code>	Text between the delimiting characters (in this case ‘!’) is verbatim.

Justification

Environment	Declaration
<code>\begin{center}</code>	<code>\centering</code>
<code>\begin{flushleft}</code>	<code>\raggedright</code>
<code>\begin{flushright}</code>	<code>\raggedleft</code>

Miscellaneous

`\linespread{x}` changes the line spacing by the multiplier x .

Text-mode symbols

Symbols

<code>&</code>	<code>\&</code>	<code>_</code>	<code>_</code>	<code>...</code>	<code>\ldots</code>	<code>•</code>	<code>\textbullet</code>
<code>\$</code>	<code>\\$</code>	<code>^</code>	<code>\^{}</code>	<code> </code>	<code>\textbar</code>	<code>\</code>	<code>\textbackslash</code>
<code>%</code>	<code>\%</code>	<code>~</code>	<code>\~{}</code>	<code>#</code>	<code>\#</code>	<code>§</code>	<code>\S</code>

Accents

<code>ò</code>	<code>\'o</code>	<code>ó</code>	<code>\'o</code>	<code>ô</code>	<code>\^o</code>	<code>õ</code>	<code>\~o</code>	<code>ō</code>	<code>\=o</code>
<code>ô</code>	<code>\.o</code>	<code>ö</code>	<code>\"o</code>	<code>q</code>	<code>\c o</code>	<code>ö</code>	<code>\v o</code>	<code>ő</code>	<code>\H o</code>
<code>ç</code>	<code>\c c</code>	<code>q</code>	<code>\d o</code>	<code>q</code>	<code>\b o</code>	<code>ö</code>	<code>\t oo</code>	<code>œ</code>	<code>\oe</code>
<code>Œ</code>	<code>\OE</code>	<code>æ</code>	<code>\ae</code>	<code>Æ</code>	<code>\AE</code>	<code>å</code>	<code>\aa</code>	<code>Å</code>	<code>\AA</code>
<code>ø</code>	<code>\o</code>	<code>Ø</code>	<code>\O</code>	<code>ı</code>	<code>\l</code>	<code>Ł</code>	<code>\L</code>	<code>ı</code>	<code>\i</code>
<code>j</code>	<code>\j</code>	<code>i</code>	<code>\~</code>	<code>ı</code>	<code>\?</code>				

Delimiters

<code>‘ ‘ ‘ ‘</code>	<code>{ { { {</code>	<code>[[[[</code>	<code>((((</code>	<code><</code>	<code>\textless</code>
<code>’ ’ ’ ’</code>	<code>} } } }</code>	<code>]]]]</code>	<code>))))</code>	<code>></code>	<code>\textgreater</code>

Dashes

Name	Source	Example	Usage
hyphen	-	X-ray	In words.
en-dash	--	1–5	Between numbers.
em-dash	---	Yes—or no?	Punctuation.

Line and page breaks

<code>\\</code>	Begin new line without new paragraph.
<code>*</code>	Prohibit pagebreak after linebreak.
<code>\kill</code>	Don’t print current line.
<code>\pagebreak</code>	Start new page.
<code>\noindent</code>	Do not indent current line.

Miscellaneous

<code>\today</code>	26. März 2018.
<code>\sim</code>	Prints $\sim$ instead of $\sim{}$, which makes $\sim$.
<code>~</code>	Space, disallow linebreak (W.J.~Clinton).
<code>\@.</code>	Indicate that the . ends a sentence when following an uppercase letter.
<code>\hspace{l}</code>	Horizontal space of length l (Ex: $l = 20\text{pt}$).
<code>\vspace{l}</code>	Vertical space of length l .

`\rule{w}{h}` Line of width w and height h .

Tabular environments

tabbing environment

`\=` Set tab stop. `\>` Go to tab stop.
Tab stops can be set on “invisible” lines with `\kill` at the end of the line. Normally `\\` is used to separate lines.

tabular environment

`\begin{array}[pos]{cols}`
`\begin{tabular}[pos]{cols}`
`\begin{tabular*}{width}[pos]{cols}`

tabular column specification

<code>l</code>	Left-justified column.
<code>c</code>	Centered column.
<code>r</code>	Right-justified column.
<code>p{width}</code>	Same as <code>\parbox[t]{width}</code> .
<code>@{decl}</code>	Insert <i>decl</i> instead of inter-column space.
<code> </code>	Inserts a vertical line between columns.

tabular elements

<code>\hline</code>	Horizontal line between rows.
<code>\cline{x-y}</code>	Horizontal line across columns x through y .
<code>\multicolumn{n}{cols}{text}</code>	A cell that spans n columns, with <i>cols</i> column specification.

Math mode

For inline math, use `\(...\)` or `\$...\$`. For displayed math, use `\[...\]` or `\begin{equation}`.

Superscript ^x	<code>^x</code>	Subscript _x	<code>_x</code>
$\frac{x}{y}$	<code>\frac{x}{y}</code>	$\sum_{k=1}^n$	<code>\sum_{k=1}^n</code>
$\sqrt[n]{x}$	<code>\sqrt[n]{x}</code>	$\prod_{k=1}^n$	<code>\prod_{k=1}^n</code>
∂	<code>\partial</code>	$\int_a^b$	<code>\int_a^b</code>

Math-mode symbols

$\leq$	<code>\leq</code>	$\geq$	<code>\geq</code>	$\neq$	<code>\neq</code>	$\approx$	<code>\approx</code>
$\times$	<code>\times</code>	$\div$	<code>\div</code>	$\pm$	<code>\pm</code>	$\cdot$	<code>\cdot</code>
$^\circ$	<code>^\circ</code>	$\circ$	<code>\circ</code>	$\prime$	<code>\prime</code>	$\cdots$	<code>\cdots</code>
∞	<code>\infty</code>	$\neg$	<code>\neg</code>	$\wedge$	<code>\wedge</code>	$\vee$	<code>\vee</code>
$\supset$	<code>\supset</code>	$\forall$	<code>\forall</code>	$\in$	<code>\in</code>	$\rightarrow$	<code>\rightarrow</code>
$\subset$	<code>\subset</code>	$\exists$	<code>\exists</code>	$\notin$	<code>\notin</code>	$\Rightarrow$	<code>\Rightarrow</code>
$\cup$	<code>\cup</code>	$\cap$	<code>\cap</code>	$\mid$	<code>\mid</code>	$\Leftrightarrow$	<code>\Leftrightarrow</code>
$\dot{a}$	<code>\dot{a}</code>	$\hat{a}$	<code>\hat{a}</code>	$\bar{a}$	<code>\bar{a}</code>	$\tilde{a}$	<code>\tilde{a}</code>
α	<code>\alpha</code>	β	<code>\beta</code>	γ	<code>\gamma</code>	δ	<code>\delta</code>
ϵ	<code>\epsilon</code>	ζ	<code>\zeta</code>	η	<code>\eta</code>	ε	<code>\varepsilon</code>
θ	<code>\theta</code>	ι	<code>\iota</code>	κ	<code>\kappa</code>	ϑ	<code>\vartheta</code>
λ	<code>\lambda</code>	μ	<code>\mu</code>	ν	<code>\nu</code>	ξ	<code>\xi</code>
π	<code>\pi</code>	ρ	<code>\rho</code>	σ	<code>\sigma</code>	τ	<code>\tau</code>
υ	<code>\upsilon</code>	ϕ	<code>\phi</code>	χ	<code>\chi</code>	ψ	<code>\psi</code>
ω	<code>\omega</code>	Γ	<code>\Gamma</code>	Δ	<code>\Delta</code>	Θ	<code>\Theta</code>
Λ	<code>\Lambda</code>	Ξ	<code>\Xi</code>	Π	<code>\Pi</code>	Σ	<code>\Sigma</code>
Υ	<code>\Upsilon</code>	Φ	<code>\Phi</code>	Ψ	<code>\Psi</code>	Ω	<code>\Omega</code>

Bibliography and citations

When using BibTeX, you need to run `latex`, `bibtex`, and `latex` twice more to resolve dependencies.

Citation types

<code>\cite{key}</code>	Full author list and year. (Watson and Crick 1953)
<code>\citeA{key}</code>	Full author list. (Watson and Crick)
<code>\citeN{key}</code>	Full author list and year. Watson and Crick (1953)
<code>\shortcite{key}</code>	Abbreviated author list and year. ?
<code>\shortciteA{key}</code>	Abbreviated author list. ?
<code>\shortciteN{key}</code>	Abbreviated author list and year. ?
<code>\citeyear{key}</code>	Cite year only. (1953)

All the above have an NP variant without parentheses; Ex. `\citeNP`.

B_BT_EX entry types

<code>@article</code>	Journal or magazine article.
<code>@book</code>	Book with publisher.
<code>@booklet</code>	Book without publisher.
<code>@conference</code>	Article in conference proceedings.
<code>@inbook</code>	A part of a book and/or range of pages.
<code>@incollection</code>	A part of book with its own title.
<code>@misc</code>	If nothing else fits.
<code>@phdthesis</code>	PhD. thesis.
<code>@proceedings</code>	Proceedings of a conference.
<code>@techreport</code>	Tech report, usually numbered in series.
<code>@unpublished</code>	Unpublished.

B_BT_EX fields

<code>address</code>	Address of publisher. Not necessary for major publishers.
<code>author</code>	Names of authors, of format
<code>booktitle</code>	Title of book when part of it is cited.
<code>chapter</code>	Chapter or section number.
<code>edition</code>	Edition of a book.
<code>editor</code>	Names of editors.
<code>institution</code>	Sponsoring institution of tech. report.
<code>journal</code>	Journal name.
<code>key</code>	Used for cross ref. when no author.
<code>month</code>	Month published. Use 3-letter abbreviation.
<code>note</code>	Any additional information.
<code>number</code>	Number of journal or magazine.
<code>organization</code>	Organization that sponsors a conference.
<code>pages</code>	Page range (2,6,9--12).
<code>publisher</code>	Publisher's name.
<code>school</code>	Name of school (for thesis).
<code>series</code>	Name of series of books.
<code>title</code>	Title of work.
<code>type</code>	Type of tech. report, ex. "Research Note".
<code>volume</code>	Volume of a journal or book.
<code>year</code>	Year of publication.

Common B_BT_EX style files

<code>abbrv</code>	Standard	<code>abstract</code>	alpha with abstract
<code>alpha</code>	Standard	<code>apa</code>	APA
<code>plain</code>	Standard	<code>unsrt</code>	Unsorted

The L^AT_EX document should have the following two lines just before `\end{document}`, where `bibfile.bib` is the name of the B_BT_EX file.

```
\bibliographystyle{plain}
\bibliography{bibfile}
```

B_BT_EX example

The B_BT_EX database goes in a file called *file.bib*, which is processed with `bibtex file`.

```
@String{N = {Na\-ture}}
@Article{WC:1953,
  author = {James Watson and Francis Crick},
  title = {A structure for Deoxyribose Nucleic Acid},
  journal = N,
  volume = {171},
  pages = {737},
  year = 1953
}
```

Sample L^AT_EX document

```
\documentclass[11pt]{article}
\usepackage{fullpage}
\title{Template}
\author{Name}
\begin{document}
\maketitle

\section{section}
\subsection*{subsection without number}
text \textbf{bold text} text. Some math:  $2+2=5$ 
\subsection{subsection}
text \emph{emphasized text} text. \cite{WC:1953}
discovered the structure of DNA.
```

A table:

```
\begin{table}[!th]
\begin{tabular}{|l|c|r|}
\hline
first & row & data \\
second & row & data \\
\hline
\end{tabular}
\caption{This is the caption}
\label{ex:table}
\end{table}
```

The table is numbered `\ref{ex:table}`.
`\end{document}`