

\LaTeX -Tutorium der Fachschaft WIAI

Valentin Barth, Linus Dietz, Jascha Karp, Michael Träger, Christian Zeck

Fachschaft Wirtschaftsinformatik und Angewandte Informatik
der Otto Friedrich Universität Bamberg

22. September 2014





- Powered by Prof. Dr. Udo Krieger
- Version 9.2
- In \LaTeX -Beamer gesetzt und erweitert von Linus Dietz
- Ursprünglicher Foliensatz von
 - Marcel Grossmann
 - Steffen Illig
 - Martin Sticht
 - Michael Timpelan
- ca. 2500 Codezeilen





Sinnvoll

- Artikel
- Bücher
- wissenschaftliche Arbeiten
- Bewerbungen

Unsinn

- private Briefe
- Geburtstags-einladungen
- Getränkekarten

Wahnsinn

- Einkaufszettel
- Brainstorming
- ...

Vom Code zum Dokument

Kein WYSIWYG

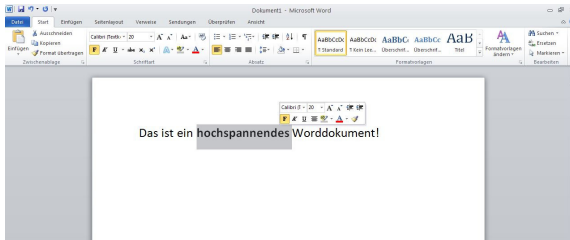


Abbildung 1: What You See Is What You Get

```
\input{./setup/command.tex}
```

```
\begin{document}
\header
%\begin{center}

% INPUTS
% \input{./setup/headline}
\input{./inhalt/freitag}
\input{./inhalt/wiaismamstag}
\input{./inhalt/sonntag}

%\end{center}
\legend
\end{document}
```

Abbildung 2: What Will I Get?

```
\input{./setup/command.tex}

\begin{document}
\header
%\begin{center}

% INPUTS
% \input{./setup/headline}
\input{./inhalt/freitag}
\input{./inhalt/wiaismstag}
\input{./inhalt/sonntag}

% \end{center}
\legend
\end{document}
```

Abbildung 3: Textdatei mit
 \LaTeX -Code



Abbildung 4:
Compiler (z.B.
MikTeX)



Abbildung 5:
schönes, lesbares und
druckbares
Dokument

Vorteile

- dynamische Verzeichnisse und Referenzen
- automatische Layouts
- einfaches verteiltes Arbeiten möglich

Nachteile

- Was kommt später raus?
- viele, zum Teil komplexe Befehle

Software unter Windows:

- MikTeX
(<http://www.miktex.org>)
2 Varianten: Basic oder Complete
- ProTeXt
(<http://www.tug.org/protext>)
enthält MikTeX,
TeXnicCenter und
Ghostscript – einfache
Installation

Software unter *nix:

- TeXLive
Pakete unter Ubuntu: `texlive-full`
ist das Meta-Paket mit allen benötigten
Paketen. Enthält auch Folgende:
 - `texlive-base`
 - `texlive-lang-german`Installation: `sudo apt-get install texlive-full`
- MacOS: MacTeX
(<http://www.tug.org/mactex/2009>)



- Kile¹
KDE-Programm, auch unter Gnome/Unity etc. verwendbar.
Installation auf Debiansystemen mit `sudo apt-get install kile`.
- Vim \LaTeX -suite (Plugin)²
Ein Traum für Vim-User.
- TexShop (MacOS)³

¹<http://kile.sourceforge.net/>

²<http://vim-latex.sourceforge.net/>

³<http://pages.uoregon.edu/koch/texshop/>



■ TeXnicCenter⁴

⁴<http://www.texniccenter.org/>



- TeXMaker⁵
Sehr solide, verwenden wir hier im Tutorium.
- TeXstudio⁶
Ähnlich wie der TeXMaker, allerdings etwas mächtiger.
- TeXlipse⁷
Für fortgeschrittene User, Plugin für Eclipse. Gute IDE-Unterstützung, Code-Completion, Autobuilds, Versionsverwaltung etc.

⁵<http://www.xmlmath.net/texmaker>

⁶<http://sourceforge.net/projects/texstudio/?source=dlp>

⁷<http://texlipse.sourceforge.net/>

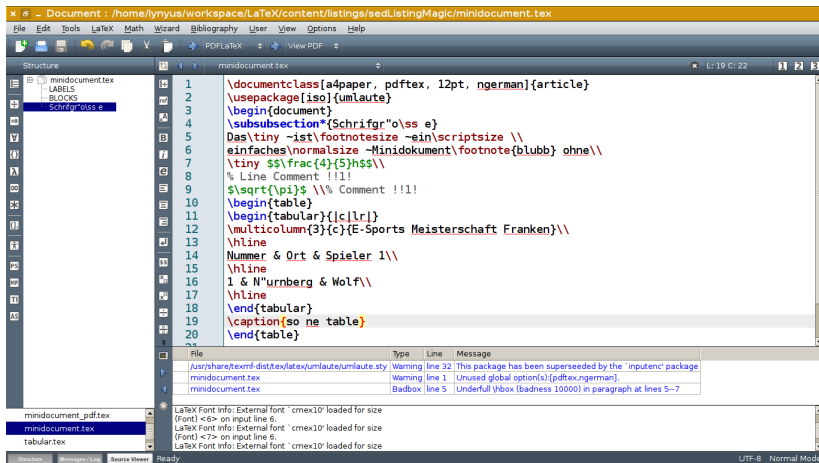


Abbildung 6: Das Standardfenster des Texmaker

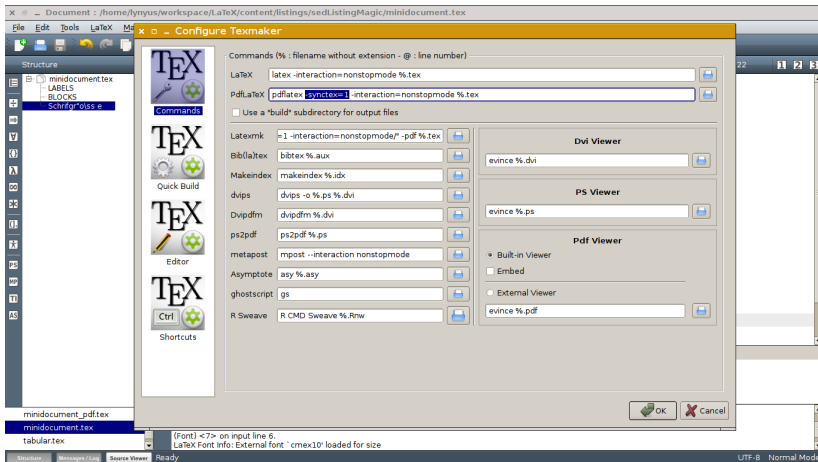


Abbildung 7: Syntex



Neue Befehle:

- `\documentclass{article}`
- `\begin{document}`
- `\end{document}`

Das ist alles was man für ein \LaTeX -Dokument braucht. Und das probieren wir jetzt aus!



Hello World!

Aufgabe 1

- Normale \LaTeX -Dateien haben `.tex` als Dateiendung

Achtung:

Bitte zu Beginn einen neuen Ordner erstellen und alle Dateien darin speichern!



Neue Befehle in diesem Abschnitt

- `\usepackage{Paket}`
- `\befehl[evtl_optionen]{
Formatierter_Text}`
- `\begin{Umgebung}`
- `\end{Umgebung}`
- `\\`
- `\newpage`
- `\sub*section{Titel}`
- `\textbf{Text}`
- `\textit{Text}`
- `\underline{Text}`
- `\tiny`
- `\scriptsize`
- `\footnotesize`
- `\normalsize`
- `\large`
- `\Large`
- `\LARGE`
- `\huge`

```
\documentclass[a4paper, pdftex, ngerman]{article}  
\usepackage[utf8]{inputenc}  
\usepackage[T1]{fontenc}  
\begin{document}
```

Das ist ein einfaches Minidokument
ohne Besonderheiten. Zeilenumbrüche
funktionieren immer automatisch!

Mehrere

Leerzeichen hintereinander werden
zu einem zusammengefasst.

Getrennt wird auch automatisch. //

Mit zwei Backslashes beginnt eine neue
Zeile. //

Ein neuer Absatz entsteht durch eine
leere Zeile.

```
\end{document}
```

Es gibt verschiedene Arten
von Dokumenten.
Hier wird die Dokumentenart
article verwendet (weiter
möglich: book und report)
In [] steht die Papiergröße
und die Schriftgröße des
Standardtextes.

```
\documentclass[a4paper, pdftex, ngerman]{article}  
\usepackage[utf8]{inputenc}  
\usepackage[T1]{fontenc}  
\begin{document}
```

Das ist ein einfaches Minidokument
ohne Besonderheiten. Zeilenumbrüche
funktionieren immer automatisch!

Mehrere

Leerzeichen hintereinander werden
zu einem zusammengefasst.

Getrennt wird auch automatisch. //

Mit zwei Backslashes beginnt eine neue
Zeile. //

Ein neuer Absatz entsteht durch eine
leere Zeile.

```
\end{document}
```

`\begin{Umgebung}`

Es beginnt eine neue
Umgebung, hier das
eigentliche Dokument.

`\end{Umgebung}`

Die mit `\begin{}`
eingeleitete Umgebung ist
hier zu Ende.

// Zeilenumbruch

```
\documentclass[a4paper, pdftex, ngerman]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\begin{document}
```

Das ist ein einfaches Minidokument
ohne Besonderheiten. Zeilenumbrüche
funktionieren immer automatisch!

Mehrere

Leerzeichen hintereinander werden
zu einem zusammengefasst.

Getrennt wird auch automatisch. //

Mit zwei Backslashes beginnt eine neue
Zeile. //

Ein neuer Absatz entsteht durch eine
leere Zeile.

```
\end{document}
```

ngerman

Für Deutschland typische
Formatierungen und
(Trenn)-Regeln werden
verwendet.

inputenc

Definiert den Zeichen-
satz, der verwendet werden
soll. Es sollte immer UTF-8
verwendet werden, weil er
universal auf allen
Betriebssystemen läuft.

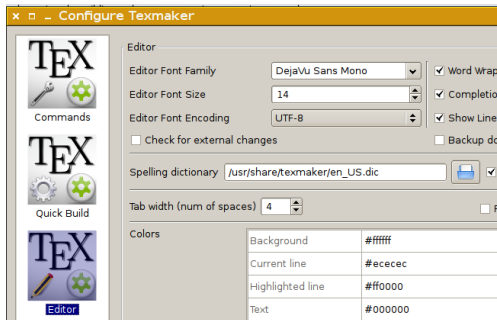


Abbildung 8: UTF-8 im Texmaker

Bei Gruppenarbeiten muss jedes Mitglied zwingend UTF-8 im Editor einstellen, sonst ist Ärger so gut wie vorprogrammiert! (Kaputte Umlaute, Kompilierungsfehler uvm., wenn es nicht nur „Windows“-User gibt.)

Wird ein Dokument geöffnet, wird automatisch der richtige Zeichensatz benutzt. Beim Erstellen neuer Dokumente wird die Datei in dem Format gespeichert, die im Editor voreingestellt ist. In den Texmaker-Einstellungen muss derselbe Zeichensatz verwendet werden, der auch im erstellten LaTeX-Dokument verwendet wird.

```
\documentclass[a4paper, pdftex,
ngerman]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\begin{document}
```

Das ist ein einfaches Minidokument
ohne Besonderheiten. Zeilenumbrüche
funktionieren immer automatisch!
Mehrere
Leerzeichen hintereinander werden
zu einem zusammengefasst.
Getrennt wird auch automatisch.\\
Mit zwei Backslashes beginnt eine neue
Zeile.\\
Ein neuer Absatz entsteht durch eine
leere Zeile.
\end{document}

Das ist ein einfaches Minidokument ohne
Besonderheiten. Zeilenumbrüche funk-
tionieren immer automatisch! Mehrere
Leerzeichen hintereinander werden zu einem
zusammengefasst. Getrennt wird auch au-
tomatisch.

Mit zwei Backslashes beginnt eine neue Zeile.

Ein neuer Absatz entsteht durch eine leere
Zeile.

Abbildung 9: Der Code von der linken
Seite als .pdf.


```
\documentclass[a4paper, pdftex, 12pt,
ngerman]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{babel}
\begin{document}
\tableofcontents
\section{Kapitel 1}
Hier kommt der erste Teil.
\subsection{Unterkapitel 1}
Das erste Unterkapitel.
\subsection{Unterkapitel 2}
Und noch ein Unterkapitel.
\subsubsection{Unterunterkapitel 1}
Das ist ein Unterkapitel von einem
Unterkapitel.
\end{document}
```

`\newpage`

Seitenumbruch

`\section{Titel}`

Ein neuer Abschnitt mit dem in {}
angegebenen Titel beginnt.

`\subsection{Titel}`

Ein Unterabschnitt.

`\subsubsection{Titel}`

Noch eine Ebene darunter.

```
\documentclass[a4paper, pdftex, 12pt,
ngerman]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{babel}
\begin{document}
\tableofcontents
\section{Kapitel 1}
Hier kommt der erste Teil.
\subsection{Unterkapitel 1}
Das erste Unterkapitel.
\subsection{Unterkapitel 2}
Und noch ein Unterkapitel.
\subsubsection{Unterunterkapitel 1}
Das ist ein Unterkapitel von einem
Unterkapitel.
\end{document}
```

1 Kapitel 1

Hier kommt der erste Teil.

1.1 Unterkapitel 1

Das erste Unterkapitel.

1.2 Unterkapitel 2

Und noch ein Unterkapitel.

1.2.1 Unterunterkapitel 1

Das ist ein Unterkapitel von einem Unterkapitel.

Abbildung 10: Die Kapitel werden automatisch mitgezählt

Neben `\section{}`, `\subsection{}`, und `\subsubsection{}` gibt es auch noch den Befehl `\part{}` welcher einen größeren Teil definiert.

`\part{}` füllt eine ganze eigene Seite.

Neben dem Dokumenttypen `article` existieren für Fließtextdokumente noch weitere wie `book` und `report`.

Bei `book` wird in der Regel zwischen linker und rechter Seite unterschieden, wobei die sich z.B. darin unterscheiden, ob die Seitenzahl links oder rechts steht, bzw. was sonst noch in der Kopf- oder Fußzeile stehen kann. In `book` und `report` gibt es noch den Gliederungsbefehl `\chapter{}`.

Unter anderem folgende Möglichkeiten:

fetter

kursiver

unterstrichener

unterstrichen und fett

Textformatierungen

\textbf{Text} fetter Text

\textit{Text} kursiver Text

\underline{Text} unterstrichen

```
\documentclass[a4paper, pdftex, 12pt,
ngerman]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\begin{document}
Unter anderem folgende
Möglichkeiten:\\
\textbf{fetter}\\
\textit{kursiver}\\
\underline{unterstrichener}\\
\underline{\textbf{unterstrichen und
fett}}\\
\end{document}
```

```
\documentclass[a4paper, pdftex,  
12pt,ngerman]{article}  
\usepackage[utf8]{inputenc}  
\usepackage[T1]{fontenc}  
\usepackage{babel}  
\begin{document}  
\tiny unlesbarer Text //  
\scriptsize sehr kleiner Text //  
\footnotesize Fußnotengröße //  
\normalsize Standardgröße //  
\large größer //  
\LARGE noch größer //  
\LARGE sehr Groß //  
\huge riesig //  
\end{document}
```

unlesbarer Text
sehr kleiner Text
Fußnotengröße
Standardgröße
größer
noch größer
sehr Groß
riesig

```
\documentclass[a4paper, pdftex, 12pt,  
ngerman]{article}
```

```
\usepackage[utf8]{inputenc}  
\usepackage[T1]{fontenc}  
\usepackage[iso]{umlaute}  
\usepackage{babel}  
% NEU NEU NEU  
\usepackage{eurosym}  
\begin{document}  
\textit{Einige Sonderzeichen:}  
% \$ & \{ \} - \# § ©/ \€  
\slash ~ \$\backslashash$ \euro  
\textit{Binde-/Gedanken-/Trennstriche:}  
- -- --- - $(letzteres mathematisches  
Minus)  
\textit{Anführungszeichen aus ngerman:}  
\glqq \grqq \flqq \frqq  
\end{document}
```

Sonderzeichen müssen mit dem '`\`' eingeführt werden. Manchmal, z.B. in Überschriften müssen Umlaute des Pakets `ngerman` mit "`a`" "`o`" "`u`" und das β mit `\ss` gebildet werden, ansonsten reicht es das Packet `babel` einzubinden.

Einige Sonderzeichen:

% \$ & { } - # § ©/ \€

Binde-/Gedanken-/Trennstriche:

- -- --- - (letzteres mathematisches
Minus)

Anführungszeichen aus (n)german:

„ “ „>>

Für das €-Zeichen wird das Package `eurosym` benötigt.



Neue Befehle in diesem Abschnitt

- `\footnote`
- `\footnotemark[]`


```
\documentclass[a4paper, pdftex, 12pt,  
ngerman]{article}  
\usepackage[utf8]{inputenc}  
\usepackage[T1]{fontenc}  
\usepackage{babel}  
\begin{document}
```

Die Fußnote `\footnote{hier folgt der Fußnotentext}` zu einem Wort oder Text erscheint immer auf der Seite, wo sie hingehört. Der Fußnotentext steht dabei in Klammern. `\\`

`\\`
Eine manuelle Nummerierung ist ebenfalls möglich. `\footnote[10]{genau so}`, auch ohne Fußnotentext `\footnotemark[2]`.
`\end{document}`

`\footnote{Fußnotentext}` Erstellt eine Fußnote an dieser Stelle mit automatischer Nummerierung.

`\footnote[Zahl]{Fußnotentext}`
Eine manuelle Nummerierung ist ebenfalls möglich.

`\footnotemark[Zahl]` Eine Zahl kann auch ohne Fußnotentext eingetragen werden

Die Nummerierung erfolgt automatisch und ist fortlaufend, unabhängig davon, ob eine neue Seite oder section beginnt.





Übung 2

- Benennt Eure Dateien einheitlich
- Verwendet passende Abschnittsbefehle
- Wenn was schief läuft, schaut in der Konsole nach
- Übung macht den Meister!



Neue Pakete in diesem Abschnitt

- `graphicx`
- `float`

Neue Befehle in diesem Abschnitt

- `\includegraphics{Datei}`
- `\caption{Bildunterschrift}`
- `\label{Label}`
- `\ref{Referenz}`

`\begin{figure}[option]`

Mögliche Optionen für die Positionierung:

`[h]` = Hier an dieser Stelle

`[t]` = Oben auf der Seite

`[b]` = Unten auf der Seite

`[p]` = Platzierung auf einer eigenen Seite

`\includegraphics[option]{datei}`

Mögliche Optionen für das Einfügen:

`[width=300pt]` = Skalieren auf eine Breite

`[height=5cm]` = Skalieren auf eine Höhe

scale, angle und noch viele mehr. . .

Kombinationen möglich:

`[width=\textwidth,height=5cm]`

```
\documentclass{article}  
\usepackage{graphicx}  
\begin{document}  
\begin{figure}[h]  
\begin{center}  
\includegraphics[width=50mm]{tux.png}  
\caption{Der kleine Tux}  
\label{img:kleinertux}  
\end{center}  
\end{figure}  
\end{document}
```

Obacht!

`\label{}` immer nach `\caption{}`

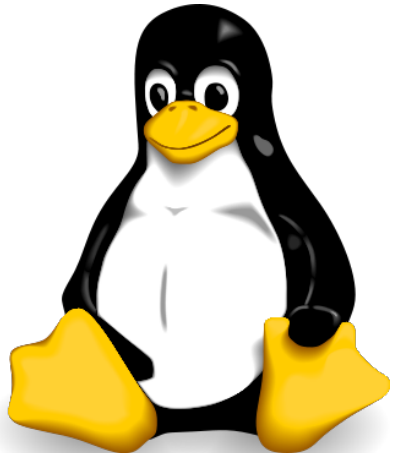


Abbildung 11: Der kleine Tux

Trotz Definition einer Positionsumgebung verrutscht das Bild oft, da es nicht immer möglich ist das Bild an passender Stelle einzufügen.

Lösung: Das Paket `float` liefert in den meisten Fällen bessere Positionierungen.

```
\documentclass{article}
\usepackage{graphicx}
\usepackage{float}
\begin{document}
\begin{figure}[H]
\begin{center}
\includegraphics[width=70mm]{pfad/tux.png}
\caption{Der kleine Tux jetzt in
Float}
\label{img:kleinertux-float}
\end{center}
\end{figure}
\end{document}
```

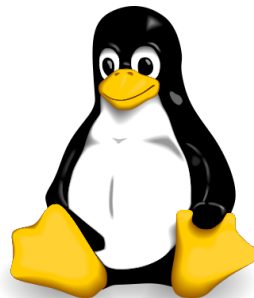


Abbildung 12: Der kleine Tux jetzt in Float

`\label{Labelname}` Mit diesem Befehl setzt man ein Label. Später im Text kann man dann durch eine Referenz auf dieses Label verweisen.
Dies geschieht mit dem Befehl `\ref{Labelname}`.

Der kleine Tux ist ein Allesfresser. Egal ob Gemüse oder Schnittlauch, nichts ist vor ihm sicher. (siehe Bild `\ref{img:tux1}`)

Ergebnis:

Der kleine Tux ist ein Allesfresser. Egal ob Gemüse oder Schnittlauch, nichts ist vor ihm sicher. (siehe Bild 1)

Und warum das Ganze?

Durch solche Referenzen wird immer auf das richtige Bild verwiesen, auch wenn zwischendurch noch weitere Bilder eingefügt wurden.





Aufgabe 3

- Versucht die Autovervollständigungsfunktion des \TeX maker zu lernen und einzusetzen
- Nicht vergessen die neuen Packages einzubinden
- Achtet auf eine intuitive Benennung der Labels



Neue Pakete in diesem Abschnitt

- `amsmath`
- `amsthm`
- `amssymb`
- `mathtools`

Neue Befehle in diesem Abschnitt

- `\sqrt{}`
- `\frac{}{}`
- `\int_X`
- `\sum_{}`
- `\lim_{}`
- `\prod`
- `\limits_{}`
- `\dots`
- `\cdot`
- `-`
- `^`

$$2\sqrt{\frac{\pi^2}{3}} \cdot c_2$$

`$2\sqrt{\frac{\pi^2}{3}}\cdot c_2$`

Die Formel-Umgebung wird durch `$` angefangen und beendet.

`$2\sqrt{\frac{\pi^2}{3}}\cdot c_2$`

So weit reicht die Wurzel.

`$2\sqrt{\frac{\pi^2}{3}}\cdot c_2$`

Ein Bruch hat immer Zähler und Nenner.

$$\int_0^{\infty}$$

`\int_0^{\infty}`

$$\sum_{i=1}^n$$

`\sum_{i=1}^n`

$$\lim_{n \rightarrow \infty}$$

`\lim_{n \rightarrow \infty}`

$$\prod_{i=1}^{n+1} i = 1 \cdot 2 \cdot \dots \cdot n \cdot (n+1)$$

`\prod_{i=1}^{n+1} i = 1`
`\cdot 2 \cdot \dots \cdot n \cdot`
`(n+1)`

Die American Mathematical Society hat einen wunderschönen Guide für das `amsmath`-Package.⁸

⁸<ftp://ftp.ams.org/pub/tex/doc/amsmath/amsl.doc.pdf>



Neue Pakete in diesem Abschnitt

- `longtable`

Neue Befehle in diesem Abschnitt

- `\begin{tabular} ... \end{tabular}`
- `\begin{table} ... \end{table}`
- `\begin{longtable} ... \end{longtable}`
- `\begin{tabbing} ... \end{tabbing}`
- `|`
- `&`
- `\hline`
- `\multicolumn{}{}{}`

Aufbau:

```
\begin{table}[Position]
\begin{tabular}{Spaltendefinitionen}
Tabelleninhalt
\end{tabular}
\caption{Untertitel}
\label{tab:bsptab1}
\end{table}
```

Reminder: Positionsangaben für die meisten \LaTeX - Umgebungen

[h] oder [H] = hier an dieser Stelle

[t] = oben auf der Seite

[b] = unten auf der Seite

[p] = Platzierung auf der einer eigenen Seite

Hier wird bestimmt wie die einzelnen Spalten ausgerichtet sein sollen und wie die senkrechten Tabellenlinien gesetzt werden sollen:

Befehle:

<code>l</code>	= linksbündige Spalte
<code>c</code>	= zentrierte Spalte
<code>r</code>	= rechtsbündige Spalte
<code>p{Spaltenbreite}</code>	= eine linksbündige Spalte mit bestimmter Spaltenbreite
<code> </code>	= setzt eine senkrechte Tabellenlinie an dieser Stelle

```
\begin{tabular}{c|p{40mm}|lr|c}  
\multicolumn{5}{c}{E-Sports Meisterschaft Franken} \\  
\hline  
\hline  
Nummer & Ort & Spieler 1 & Spieler 2 & Ergebnis \\  
\hline  
1 & Nürnberg & Wolf & Lamm & 23:10 \\  
\hline  
2 & Bamberg & Meyer & Beyer & \\  
\hline  
3 & Zirndorf & Brandst. & Brauer & 21:21\\ \\  
\hline  
\end{tabular}
```

Hier werden die definierten Spalten mit Inhalt gefüllt.

Befehle:

- `&` = dient zur horizontalen Trennung von Zellen
- `\\` = neue Zeile
- `\hline` = setzt eine waagerechte Tabellenlinie
- `\multicolumn{Spaltenzahl}{Spaltenausrichtung}{Text}`
= Verbindet beliebig viele Spalten miteinander.

```
\begin{tabular}{c|p{40mm}|lr|c}
\multicolumn{5}{c}{E-Sports Meisterschaft Franken} \\
\hline
\hline
Nummer & Ort & Spieler 1 & Spieler 2 & Ergebnis \\
\hline
1 & Nürnberg & Wolf & Lamm & 23:10 \\
\hline
2 & Bamberg & Meyer & Beyer & \\
\hline
3 & Zirndorf & Brandst. & Brauer & 21:21 \\
\hline
\end{tabular}
```

E-Sports Meisterschaft Franken

Nummer	Ort	Spieler 1	Spieler 2	Ergebnis
1	Nürnberg	Wolf	Lamm	23:10
2	Bamberg	Meyer	Beyer	
3	Zirndorf	Brandst.	Brauer	21:21

Bei „tabular“ wird die Tabelle auf einer Seite angezeigt. Wenn sie nicht draufpasst, wird sie abgeschnitten.

Für Tabellen, die länger als eine Seite sind, wird eine Tabelle benötigt die eine Trennung der Tabelle vornimmt.

Lösung: longtable

longtable ermöglicht den Seitenumbruch in der Tabelle. Ausserdem ist longtable eine eigene Umgebung, braucht deshalb keine table-Umgebung mehr!

```
\begin{longtable}{Spaltendefinitionen}  
Tabelleninhalt  
\caption{Untertitel}  
\label{tab:bsptab2}  
\end{longtable}
```



Steuerung

`\=` eine Tabulatorstelle setzen

`\>` eine Tabulatorstelle ansteuern

```
\documentclass{article}
\begin{document}
\begin{tabbing}
Mitarb\=eiter: \\
A \> Daniel \\
B \> Martin \\
C \> Linus \\
xxx\=xxx\=xxxxxxx\kill
\> Gremien \\
\>\> Klausuren \\
\>\> Emails \\
\end{tabbing}
\end{document}
```

Mitarbeiter:

A Daniel

B Martin

C Linus

Gremien

Klausuren

Emails

Durch den Befehl `\kill` wird der Zeileninhalt nicht angezeigt. Dadurch können Formatierungen vorgenommen werden, ohne den zugehörigen Text anzuzeigen.





Neue Befehle in diesem Abschnitt

- `\begin{itemize} ... \end{itemize}`
- `\begin{enumerate} ... \end{enumerate}`
- `\item`

```
\documentclass{article}
\begin{document}
\begin{itemize}
\itemerster Stichpunkt
\itemzweiter Stichpunkt
\itemdritter Stichpunkt
\itemletzter Stichpunkt
\end{itemize}
\end{document}
```

- erster Stichpunkt
- zweiter Stichpunkt
- dritter Stichpunkt
- letzter Stichpunkt

Die einzelnen Stichpunkte werden innerhalb der „itemize“-Umgebung durch den Befehl `\item` gekennzeichnet.

```
\documentclass{article}
\begin{document}
\begin{itemize}
\itemerster Stichpunkt
\itemzweiter Stichpunkt
\begin{itemize}
\itemerster Unterpunkt
\itemzweiter Unterpunkt
\end{itemize}
\itemdritter Stichpunkt
\itemletzter Stichpunkt
\end{itemize}
\end{document}
```

- erster Stichpunkt
- zweiter Stichpunkt
 - erster Unterpunkt
 - zweiter Unterpunkt
- dritter Stichpunkt
- letzter Stichpunkt

Auf diese Weise kann man Unterpunkte bis auf 4 Ebenen tief schachteln.



```
\documentclass{article}
\begin{document}
\begin{enumerate}
\itemerstens
\begin{enumerate}
\itemerster Unterpunkt
\itemzweiter Unterpunkt
\end{enumerate}
\itemzweitens
\itemusw.
\end{enumerate}
\end{document}
```

1 erstens

1 erster Unterpunkt

2 zweiter Unterpunkt

2 zweitens

3 usw.

Auch hier werden die einzelnen Punkte durch den Befehl `\item` gekennzeichnet. Schachtelungen können wieder bis zu 4 Ebenen tief sein.

Gemischte Aufzählungen?

Geht Alles!



```
\begin{enumerate}
\itemerstens
\item\begin{itemize}
\itemerster Unterpunkt
\itemzweiter Unterpunkt
\end{itemize}
\itemdrittens
\begin{enumerate}
\itemAuch ich zähle!
\end{enumerate}
\itemusw.
\end{enumerate}
```

- 1 erstens
- 2
 - erster Unterpunkt
 - zweiter Unterpunkt
- 3 drittens
 - 1 Auch ich zähle!
- 4 usw.

Die Darstellung der jeweiligen Symbole kann mit `\item[]` angepasst werden.





Aufgabe 4

- Achtet auf die `\\` am Ende einer Tabellenzeile!
- Seid aufmerksam bei den Verschachtelung der `\items`
- Dabei kann es Euch helfen, den Code einzurücken um den Überblick zu behalten

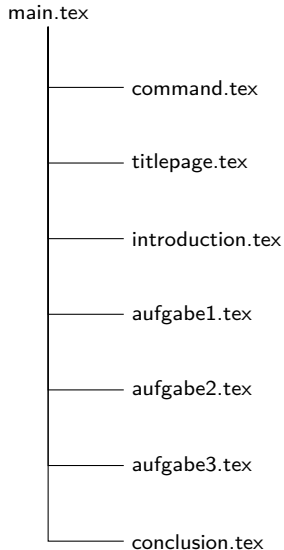


Ein handfestes Dokument aufbauen

... und dabei den volle Charme von \LaTeX erleben!

Neue Befehle in diesem Abschnitt

- `\input{}`
- `\tableofcontents`
- `\listoffigures`
- `\listoftables`
- `\vspace{}`
- `\today`
- `\begin{titlepage} ... \end{titlepage}`



In der main.tex werden alle anderen Dateien zu einem Dokument zusammengefasst. Dazu muss man die einzelnen Dateien dafür anpassen.

Es muss **alles** (einschliesslich) vor und nach `\begin-` und `\end{document}` gelöscht werden:

```
\documentclass[pdftex]{article}
```

```
\usepackage{babel}
```

```
\begin{document}
```

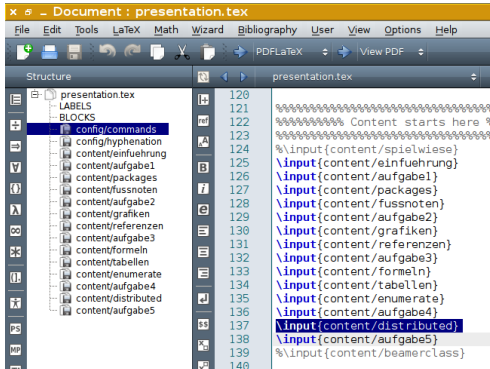
Dieses Dokument kann nun mit dem Befehl `\input\{Dateiname\}` in LaTeX eingebunden werden. `\\`

```
\end{document}-
```

Mit dem Befehl

`\input{pfad/zur/datei}` kann man danach diese in eine andere .tex-Datei einbinden.

2



Achtung:

Die Pfadangabe ist
immer relativ zur
Hauptdatei!

Abbildung 13: Texmaker listet die `\inputs`



L^AT_EX

Eine kurze Einführung

Universität Bamberg

10. Februar 2015
Fachschaft WIAI

```
\begin{titlepage}  
\begin{center}  
\Huge\LaTeX\\  
\vspace{5mm} \LARGE Eine kurze  
Einführung\\  
\vspace{12mm} \Large Universität  
Bamberg\\[5mm]  
\large\today\\  
Fachschaft WIAI\normalsize\\  
\end{center}  
\end{titlepage}
```

Seitennumerierung

- `\thispagestyle{empty}` – Keine Seitennumerierung
- `\setcounter{page}{1}` – Setzt die Seitennumerierung auf einen bestimmten Wert
- `\pagenumbering{Roman|roman|arabic|Alph|alph}` – Definiert die Seitenzählung
- `\newpage` – Erzeugt eine neue Seite



Aufgabe 5

Baut eines der beiden Layouts für Aufgabe 5 in \LaTeX nach!

Universität Bamberg



Aufgabe 5

- Baut euch eine Titelseite!
- Verwendet Aufgabe $\mathcal{A}2$ bis $\mathcal{A}4$ als Inhalt.
- Achtet auf die richtige Seitennummerierung!
- Ihr könnt euch für eines der zwei Layouts aus dem VC-Kurs entscheiden.



Neue Pakete in diesem Abschnitt

- natbib

Neue Befehle in diesem Abschnitt

- `\cite{Author2014}`
- `\bibliographystyle{alpha|abbrv|natdin|apa|etc.}`
- `\bibliography{literature.bib}`



Vorteile:

- Literaturverzeichnis wird in einer vom Dokument unabhängigen .bib-Datei gespeichert.
- Speicherung der Daten im BIB_TE_X-Format. Hierbei wird nach Quellenart unterscheiden, z.B. mit @book, @article usw.
- Große Auswahl an Zitierstilen
- **Automatische, dem Style entsprechende, Generierung des Literaturverzeichnisses (LVZ)**
- Aufnahme der Einträge in das LVZ nur wenn die Quelle zuvor im Text zitiert wurde

Nachteile:

- Das Erstellen des LVZ mit besonderen Anforderungen ist zum Teil nur erschwert möglich

Beispieleintrag:

```
@book{Culik93,  
  title= {Die Welt der Pinguine},  
  author= {B.M. Culik and R. P.  
    Wilson},  
  publisher= {{BLV}München},  
  year= {1993}  
}
```

Erklärungen zum Eintrag:

- **@book** - Angabe der Quellenart, hier also ein Buch
- Culik93 - Definition eines eindeutigen Referenzierungsschlüssels
- **author** - Autor des Buches
- **title** - Titel des Buches
- **publisher** - Verlag
- **year** - Erscheinungsjahr

Achtung: Reihenfolge beim Kompilieren beachten!

(1) pdf_latex (2) bib_tex (3) pdf_latex (4) pdf_latex



- | | |
|--|--|
| ■ <code>\cite{Culik93}</code> | Zitieren eines BIB _T E _X Eintrages |
| ■ <code>\bibliographystyle{alphadin}</code> | Auswahl des LVZ-Stils „alphadin“ |
| ■ <code>\bibliography{bibliography.bib}</code> | Angabe der BIB _T E _X -Datei |
| ■ <code>\usepackage{natbib}</code> | Für den Zitierstil <code>natdin</code> notwendig |

alpha

Pinguine sind laut [CW93, S. 85] allmächtig. [Bau12] stellt das in Frage und [Fal02] benennt seine Datenbank nach ihnen.

Literatur

- [Bau12] Mick Bauer. Paranoid penguin: Eleven years of paranoia, a retrospective. *Linux J.*, 2012(213), January 2012.
- [CW93] B. M. Culik and R. P. Wilson. *Die Welt der Pinguine*. BLV München, 1993.
- [Fal02] Daniel Fallman. The penguin: Using the web as a database for descriptive and dynamic grammar and spell checking. In *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '02, pages 616–617, New York, NY, USA, 2002. ACM.

Abbildung 14: Alpha Zitierstil

IEEEtran

Pinguine sind laut [1, S. 85] allmächtig. [2] stellt das in Frage und [3] benennt seine Datenbank nach ihnen.

Literatur

- [1] B. M. Culik and R. P. Wilson, *Die Welt der Pinguine*. BLV München, 1993.
- [2] M. Bauer, “Paranoid penguin: Eleven years of paranoia, a retrospective,” *Linux J.*, vol. 2012, no. 213, Jan. 2012. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2132860.2132867>
- [3] D. Fallman, “The penguin: Using the web as a database for descriptive and dynamic grammar and spell checking,” in *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '02. New York, NY, USA: ACM, 2002, pp. 616–617. [Online]. Available: <http://doi.acm.org/10.1145/506443.506511>

Abbildung 15: IEEEtran Zitierstil

Natdin

Pinguine sind laut (Culik u. Wilson, 1993, S. 85) allmächtig. Bauer (2012) stellt das in Frage und Fallman (2002) benennt seine Datenbank nach ihnen.

Literatur

- [Bauer 2012] BAUER, Mick: Paranoid Penguin: Eleven Years of Paranoia, a Retrospective. In: *Linux J.* 2012 (2012), Januar, Nr. 213. <http://dl.acm.org/citation.cfm?id=2132860.2132867>. – ISSN 1075–3583
- [Culik u. Wilson 1993] CULIK, B. M. ; WILSON, R. P.: *Die Welt der Pinguine*. BLV München, 1993
- [Fallman 2002] FALLMAN, Daniel: The Penguin: Using the Web As a Database for Descriptive and Dynamic Grammar and Spell Checking. In: *CHI '02 Extended Abstracts on Human Factors in Computing Systems*. New York, NY, USA : ACM, 2002 (CHI EA '02). – ISBN 1–58113–454–1, 616–617

Abbildung 16: Natdin Zitierstil

Apa

Pinguine sind laut [Culik and Wilson1993, S. 85] allmächtig. [Bauer2012] stellt das in Frage und [Fallman2002] benennt seine Datenbank nach ihnen.

Literatur

[Bauer2012] Bauer, M. (2012). Paranoid penguin: Eleven years of paranoia, a retrospective. *Linux J.*, 2012(213).

[Culik and Wilson1993] Culik, B. M. and Wilson, R. P. (1993). *Die Welt der Pinguine*. BLV München.

[Fallman2002] Fallman, D. (2002). The penguin: Using the web as a database for descriptive and dynamic grammar and spell checking. In *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '02, pages 616–617, New York, NY, USA. ACM.

Abbildung 17: Apa Zitierstil

Questions ?

Fachschaft WIAI

fachschaft.wiai@uni-bamberg.de