

LaTeX-Praxis

# Dokumente gestalten

Jörn Clausen  
joernc@gmail.com

# Übersicht

- Schriften
- Umgebungen
- Makros und eigene Umgebungen

## Schriften

- zu T<sub>E</sub>X gehörende Schriftfamilie: Computer Modern
- mehrere Schnitte: CM Roman, CM Sans, CM Typewriter, ...
- Variationen: **bold**, **bold extended**, *italic*, *slanted*, ...
- ebenfalls von Donald Knuth entwickelt: METAFONT
- weitere Schriften: Linotype Syntax, Times, Helvetica, ...
- aber: viele mathematische Symbole nur in Computer Modern

## Schriften, cont.

- *New Font Selection Scheme* (NFSS)
- vier Eigenschaften bestimmen Schrift:
  - Familie: Roman, Sans Serif, Typewriter
  - Serie: medium, **bold**
  - Form: upright, *italic*, *slanted*, SMALL CAPS
  - Größe: tiny, . . . , small, . . . , large, . . . , Huge
- Eigenschaften unabhängig voneinander
- aber nicht zu jeder Kombination Font vorhanden

## Schriften auswählen

- Familie:

```
\textrm{Roman}           \textsf{Sans Serif}  
\texttt{Typewriter}
```

- Serie:

```
\textmd{medium}         \textbf{bold}
```

- Form:

```
\textup{upright}        \textit{italic}  
\textsl{slanted}        \textsc{small caps}
```

- Kombinationen:

```
\textbf{das ist fett \textit{und kursiv}}  
\textsl{geneigt, \textbf{fett und \textsf{serifenlos}}}
```

# Aufgaben

- Probiere die NFSS-Anweisungen aus. Versuche vor allem, mehrere Schrifteigenschaften gleichzeitig zu ändern.
- Nicht zu jeder Kombinationen von NFSS-Befehlen gibt es auch tatsächlich eine Schrift. Versuche, eine solche Kombination zu finden. Was passiert in diesem Fall? Wie weist  $\LaTeX$  auf das Problem hin?
- Wozu gibt es die Befehle `\textrm`, `\textmd` und `\textup`, die doch offensichtlich die „normale“ Schrift auswählen?

Falls man kursive statt geneigter Schrift haben möchte, muss man darauf achten, in der zweiten Variante beide Schrifteigenschaften konsistent zu ändern.

```
\textsl{Das ist eine \textsl{M"oglichkei}}
ist deutlich besser als
\textsl{Das ist \textup{eine} M"oglichkei}}
```

- Mit Hilfe der Befehle `\textrm`, `\textmd` und `\textup` kann man kurzzeitig zur „normalen“ Schrift wechseln. Die Variante

```
cmt/bx/sc <- cmt/bx/n
cmt/bx/sl <- cmt/bx/n
cms/bx/sc <- cms/bx/n
cms/bx/sl <- cms/bx/n
cms/bx/it <- cms/bx/n
cms/m/sc <- cms/m/sc
cms/m/it <- cms/m/sl
cmt/bx/sc <- cmt/bx/n
```

- Ersetzungen:

Typewriter	bold	<i>upright</i>	<i>italic</i>	<i>slanted</i>	small caps
	medium	<i>upright</i>	<i>italic</i>	<i>slanted</i>	SMALL CAPS
Sans Serif	bold	<i>upright</i>	<i>italic</i>	<i>slanted</i>	<b>small caps</b>
	medium	<i>upright</i>	<i>italic</i>	<i>slanted</i>	SMALL CAPS
Roman	bold	<i>upright</i>	<i>italic</i>	<i>slanted</i>	<i>small caps</i>
	medium	<i>upright</i>	<i>italic</i>	<i>slanted</i>	SMALL CAPS

- Die Computer Modern Familie:

## Schriften auswählen, cont.

- Schriftgröße:

```
\tiny    \scriptsize  \footnotesize  \small    \normalsize  
\large   \Large           \LARGE         \huge     \Huge
```

- Achtung: Diese Befehle funktionieren anders!

```
{\tiny das ist ganz klein}
```

```
\begin{quote}  
\Large das ist sehr gro"s  
\end{quote}
```

- alternative Befehle für Familie, Serie und Form
- siehe NFSS Reference Card

# Umgebungen

- Listen, Aufzählungen, Tabellen, Grafiken, ...
- Umgebungen „klammern“ Bereiche des Dokuments
- `\begin{umgebung} ... \end{umgebung}`
- bereits kennengelernt: `document`
- Umgebungen können geschachtelt werden
- Schachtelung muss passen:

<code>\begin{ausser}</code>	<code>\begin{ausser}</code>
<code>\begin{innen}</code>	<code>\begin{innen}</code>
<code>...</code>	<code>...</code>
<code>\end{innen}</code>	<code>\end{ausser}</code>
<code>\end{ausser}</code>	<code>\end{innen}</code>



# Aufgaben

- Probiere die Umgebung `itemize` aus. Mit ihr kann man Listen erstellen:

```
\begin{itemize}
\item Huey
\item Dewey
\item Louie
\end{itemize}
```

- Versuche, mehrere `itemize`-Umgebungen zu schachteln, d.h. beginne innerhalb eines `\items` eine neue Liste. Was ist zu beobachten? Wie tief lassen sich Listen auf diese Weise schachteln?

- Das Symbol vor jedem Listeneintrag variiert mit der Schachtelungstiefe:
    - zweite Ebene
    - auch zweite Ebene
    - \* dritte Ebene
    - \* immer noch dritte Ebene
    - vierte Ebene
    - ebenfalls vierte Ebene
    - \* hier wieder dritte Ebene
- Es sind maximal vier Ebenen möglich, sonst bricht  $\LaTeX$  die Übersetzung mit einer Fehlermeldung ab.

# Aufgaben

- Ersetze in einer Listenumgebung das Wort `itemize` durch `enumerate`:

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

Was passiert?

- Probiere, `enumerate`-Umgebungen ineinander zu schachteln.
- Mische `itemize`- und `enumerate`-Umgebungen miteinander.

- Die Einträge in der Aufzählung werden fortlaufend nummeriert, je nach Schachtelungstiefe mit unterschiedlichen Zahlen oder Symbolen:
  1. erste Ebene
  2. auch hier erste Ebene
    - a) zweite Ebene
    - b) auch zweite Ebene
  - i. dritte Ebene
  - ii. immer noch dritte Ebene
    - A. vierte Ebene
    - B. ebenfalls vierte Ebene
  - iii. hier wieder dritte Ebene
- Listen und Aufzählungen können beliebig gemischt werden. Das Limit von maximal vier ineinandergeschachtelten Umgebungen gilt für die beiden Arten von Listen unabhängig.

## Zitate

Von Pythagoras, der im alten Griechenland lebte, ist uns folgender Satz "überliefert:

```
\begin{quote}
Die Fläche des Quadrats "über
der Hypotenuse ist gleich der
Summe der Flächen der Quadrate
"über den Katheten.
\end{quote}
```

Von Pythagoras, der im alten Griechenland lebte, ist uns folgender Satz überliefert:

Die Fläche des Quadrats über der Hypotenuse ist gleich der Summe der Flächen der Quadrate über den Katheten.

## Aufgabe

- Setze das folgende Gedicht:

```
\begin{verse}
  Wer reitet so sp"at durch Wind und Nacht? \\
  Es ist der Vater. Es ist gleich acht. \\
  Im Arm den Knaben er wohl h"alt, \\
  er h"alt ihn warm, denn er ist erk"alt'.

  Halb drei, halb f"unf. Es wird schon hell. \\
  Noch immer reitet der Vater schnell. \\
  Erreicht den Hof mit M"uh und Not -- \\
  der Knabe lebt, das Pferd ist tot!
\end{verse}
```

Welche Aufgabe haben die \\ am Zeilenende, was bewirkt die Leerzeile?

- Die doppelten backslashes am Zeilenende erzeugen einen Zeilenumbruch. Man kann sie auch in "normalem" Text verwenden, aber dort sollte man besser  $\TeX$  den Zeilenumbruch überlassen. In einem Gedicht muss man die Formatierung natürlich vorgeben, damit es sich hinten reimt. . . Die Leerzeile trennt einzelne Strophen voneinander, verhält sich also ähnlich wie die Leerzeile zum Trennen von Absätzen.

## Makros

- direkte Verwendung von Schriftbefehlen vermeiden
- „hervorheben“ statt „kursiv“
- `\emph{sehr} gut`: *sehr* gut
- `\emph{auch \emph{das} noch}`: *auch das noch*
- Makros und Umgebungen selber definieren

## einfache Makros definieren

- Floskeln, Abkürzungen, exakte Schreibweisen:

```
\newcommand{\gmail}{googlemail.com}  
\newcommand{\MFG}{Mit freundlichen Gr\"u\ss{}}en}
```

```
Meine Adresse lautet \texttt{juser@}\gmail}.
```

```
\MFG, Joe User
```

- Pakete (z.B. babel) erst nach `\begin{document}` aktiv
- Makros flexibel definieren

## Aufgaben

- Definiere und verwende ein Makro, das den Ausdruck „Hypertext Markup Language“ erzeugt. Schreibe den Satz

Die Hypertext Markup Language wurde von Tim Berners-Lee erfunden.

mit Hilfe dieses Makros.

- Was passiert, wenn Du versuchst, dieses Makro zu definieren?

```
\newcommand{\dots}{...}
```

- einfache Makrodefinition:  

```
\newcommand{\HTML}{Hypertext Markup Language}
```

  

```
Die \HTML{} wurde von Tim Berners-Lee erfunden.
```
- Das Makro `\dots` ist bereits definiert, daher bricht L<sup>A</sup>T<sub>E</sub>X die Übersetzung mit einer Fehlermeldung ab. Man kann Makros mit dem Befehl `\renewcommand` umdefinieren, allerdings ist das in diesem Fall nicht ratsam.

## Makros mit Argumenten

- aus einem Reiseführer:

```
\newcommand{\City}[1]{\textsc{#1}}  
\newcommand{\CityAt}[2]{\textsc{#1 (#2)}}
```

Die Reise f"uhrt von \City{K"oln} und \City{Bonn} den Rhein entlang "uber \City{Koblenz} nach \City{Wiesbaden} und von dort weiter nach \CityAt{Frankfurt}{am Main}.

- [ 1 ], [ 2 ]: Anzahl Parameter
- #1 erster Parameter, #2 zweiter Parameter, ...
- maximal 9 Parameter



## Aufgaben

- Überzeuge Dich davon, dass die gezeigten Makros funktionieren:

```
\newcommand{\City}[1]{\textsc{#1}}  
\newcommand{\CityAt}[2]{\textsc{#1 (#2)}}
```

... von `\City{K"oln}` nach `\CityAt{Frankfurt}{am Main}`.

- Was passiert, wenn Du die Makros mit der falschen Anzahl Parameter verwendest:

... von `\CityAt{K"oln}` nach `\City{Frankfurt}{am Main}`.

- Bei der "falschen" Verwendung gibt es keine Warnungen oder Fehlermeldungen. Das Makro `\CityAt` nimmt sich den nächsten Buchstaben (das "n" von "nach") als zweites Argument, das Makro `\City` ignoriert das zweite Paar Klammern einfach, der Text "am Main" wird als ganz normaler Text gesetzt.

## Aufgaben

- Definiere ein Makro `\email`, um EMail-Adressen zu setzen.

```
\email{webmaster}{uni-bielefeld.de}
```

soll die Ausgabe `webmaster@uni-bielefeld.de` erzeugen.

- Definiere ein Makro `\fhemail`, um EMail-Adressen innerhalb der Fachhochschule Bielefeld zu setzen.

```
\fhemail{juser}
```

soll die Ausgabe `juser@fh-bielefeld.de` erzeugen.

Verwende das Makro `\email` bei der Definition von `\fhemail`.

```
\newcommand{\fhemail}[1]{\email{#1}{fh-bielefeld.de}}  
\newcommand{\email}[2]{\textt{#1@#2}}
```

## Makros, cont.

- sinnvolle, sprechende Namen verwenden
- Verwendung beschreiben, nicht Formatierung
- `\City{...}` statt `\Smallcaps{...}`
- Namenskollisionen vermeiden
- z.B. durch gemischte Groß-/Kleinschreibung

## Umgebungen selber definieren

- ähnlich zu Makros
- gleiche Regeln für Namenswahl
- Was passiert bei `\begin{...}`, was bei `\end{...}`?
- % Kommentarzeichen von TeX
- hier: Unterdrückung unbeabsichtigter Zeilenumbrüche

## einfache Umgebung

```
\newenvironment{Merksatz}
{\begin{quote}%
 \large\bfseries}
{\end{quote}}
```

```
Chemiker wissen:
\begin{Merksatz}
  Nie das Wasser in die S"auere,
  sonst geschieht das Ungeheure.
\end{Merksatz}
```

Chemiker wissen:

**Nie das Wasser in  
die Säure, sonst  
geschieht das Un-  
geheure.**

## Hausaufgabe

- Sieh Dir (z.B. in der „[L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>-Kurzbeschreibung](#)“) die folgenden Umgebungen an:
  - `description`
  - `center`, `flushleft`, `flushright`
  - `tabbing`