

Textverarbeitung mit \LaTeX

Mag. Otto Dolinsek

Oktober 2007

Inhaltsverzeichnis

1	Dokumentklassen	4
2	Beispiel eines LaTeX Dokumentes	5
3	Grundlagen der Textformatierung und Strukturierung	6
3.1	Absatzgestaltung	6
3.2	Absatz-Ausrichtung	7
3.3	Vertikale Abstände zwischen Absätzen	8
3.4	Umbrüche	9
3.5	Horizontale Wortabstände	10
3.6	Schriftgröße	11
3.7	Farben mit dem Paket xcolor	12
3.8	Tabulatoren	13
3.9	Fußnoten	14
3.10	Farben mischen!	15
3.11	Textmaskierung	16
3.11.1	Textmaskierung mit dem Paket fancyvrb	17
3.11.2	Source Code einfügen	18
3.12	Binde- und Gedankenstriche	20
4	Listen	21
4.1	Listen mit Zwischenabstand	21
4.1.1	Unnummerierte Aufzählungsliste	21
4.1.2	Nummerierte Aufzählungslisten	21
4.1.3	Beschreibungsliste	21
4.1.4	Verschachtelte Listen	22
4.2	Listen ohne Zwischenabstand mit dem Package mdwlist	23
4.2.1	Aufzählungslisten	23
4.2.2	Nummerierte Aufzählungslisten	24
4.3	Flexible Listen mit dem Package paralist	25
4.3.1	Flexible Aufzählungslisten mit dem Package paralist	25
4.3.2	Flexible Listen mit dem Package paralist ohne Zeilenabstand	25
4.4	Inline Listen	26
4.5	Allgemeine Listen mit der list Umgebung	27
4.6	Listenmarken definieren	28
5	Tabellen	29
5.1	Tabellen definieren	29
5.2	Tabelle in der Gleitumgebung	30
5.3	Spaltenabstände	31
5.4	Spaltenzwischeräume	31
5.5	Mehrspaltige Zellen	32

5.6	Mehrzeilige Zellen	33
5.7	Vertikale Ausrichtung von Zellen	34
6	Formeln	35
6.1	Einfache Formeln	35
6.2	Einzeilige Formeln	36
6.3	Mehrzeilige, ausgerichtete Formeln	36
7	Grafik	37
7.1	Grafik einbinden	37
7.2	Einheiten	38
7.3	Grafiken nebeneinander	39
7.4	Text umflossene Grafiken	40
8	Literaturangaben	41
9	Satzspiegel	42
10	Kopf- und Fußzeile	43
11	Kommutative Diagramme	44
11.1	xymatrix	44
11.2	Pfeile	44
11.3	Beschriftungen der Pfeile	45
11.3.1	Erweiterungen der Pfeilbeschriftungen	45
11.4	Pfeile senkrecht zur Pfeilrichtung verschieben	45
11.5	Beschriftung in Pfeilrichtung verschieben	46
11.5.1	Gekrümmte Pfeile	46
11.5.2	Erscheinungsbild des Pfeiles	47
11.6	Rahmenformen der Zellen	47
11.7	Kopierbeispiele	49

1 Dokumentklassen

Am Anfang eines Dokumentes wird immer die Dokumentklasse ausgewählt. Eigenschaften wie Schriftenwahl, Seitenaufbau und viele andere Eigenschaften sind vordefiniert.

```
\documentclass[a4paper]{scrartcl}
```

Folgende Dokumentklassen stehen zur Verfügung:

scrartcl Klasse für kurze Texte und Aufsätze. Es wird keine separate Titelseite gesetzt.

scrreprt Klasse für Berichte und längere Arbeiten. Eine Titelseite wird gesetzt.

scrbook Klasse für Bücher. Doppelseitige Ausgabe, wobei Kapitel immer auf der rechten Seite beginnen.

scrlttr2 Klasse für Briefe.

Der eigentliche Text wird mit dem Kommando `\begin{document}` eingeleitet und mit `\end{document}` abgeschlossen. Innerhalb stehen dann einige Kommandos zur Verfügung, mit denen der Text strukturiert werden kann:

<code>\part{Überschrift}</code>	Teil (nur scrbook und scrreprt)
<code>\chapter{Überschrift}</code>	Kapitel (nur scrbook und scrreprt)
<code>\section{Überschrift}</code>	Abschnitt
<code>\subsection{Überschrift}</code>	Unterabschnitt
<code>\subsubsection{Überschrift}</code>	Unterunterabschnitt
<code>\paragraph{Überschrift}</code>	Paragraph
<code>\subparagraph{Überschrift}</code>	Nomen est omen.

\LaTeX kümmert sich selbstständig um die Nummerierung der Kapitel und Abschnitte. Gleichzeitig wählt \LaTeX automatisch geeignete Schriftarten und Textabstände für die Überschriften. Bei dem Dokumenttyp book werden die Texte des `\chapter`- und `\section` Kommandos auch beim Erstellen der Kopfzeilen berücksichtigt. Wenn die Kommandos in der Form `\section [Kurzfassung] {vollständig}` verwendet werden, wird die Kurzfassung für die Kopfzeile und das Inhaltsverzeichnis, die vollständig Variante dagegen unmittelbar im Text verwendet.

\LaTeX wird in einfachen Textdateien gespeichert, aus welchen die Ausgabe erzeugt wird. Die Textdateien können mit jedem beliebigen Editor bearbeitet werden.

2 Beispiel eines LaTeX Dokumentes

```

\documentclass[a4paper,headsepline]{scrartcl}
\usepackage[a4paper,includehead,includefoot,left=3.5cm,width=15cm,top=2cm,bottom=2cm,
headheight=2cm,headsep=1cm,footskip=1.7cm]{geometry}
%A4 Satzspiegel setzen 21cmx29,7cm
\usepackage[autooneside,automark]{scrpage2}%Kopf- und Fußzeile
\pagestyle{scrheadings}
\clearscrheadfoot
%\automark[section]{subsection}
\ohead[]{\headmark}
\chead[]{\HTML-CSS-PHP}
\ihead[]{\includegraphics[height=2cm]{logo.jpg}}
\ifoot[]{\today}
\cfoot[]{\Mag. Otto Dolinsek}
\ofoot[]{\pagemark}
%\renewcommand{\headfont}{\normalfont\sffamily\bfseries}
%Kopf- und Fußzeile umformatieren
\usepackage[ngerman]{babel}
\usepackage[T1]{fontenc}
\usepackage[ansinew]{inputenc}
\usepackage{makeidx}
\usepackage{xcolor}
\usepackage{mdwlist}%Listenpaket
\usepackage{paralist}%Listenpaket
\usepackage{fancyvrb}%Paket für Textmaskierung Verbatim
\usepackage{listings}%Paket für Sourcecode
\usepackage{amsmath,amssymb}%Mathematikpaket
\usepackage{multirow}%Tabellenpaket für mehrzeilige Zellen
\usepackage{array}%Tabellenpaket für Spaltenzwischenräume
%\pagestyle{plain}%Standard Seitennummerierung
%\pagestyle{headings}%Standardkopfzeile
\usepackage{graphicx}%Grafikpaket
\usepackage{wrapfig}%Grafikpaket
\definecolor{hellgelb}{rgb}{1,1,0.8}
\definecolor{darkred}{rgb}{0.6,0,0}
\definecolor{darkgreen}{rgb}{0,0.6,0}
\definecolor{darkblue}{rgb}{0,0,0.6}
\title{Textverarbeitung mit \LaTeX{}}
\author{Mag. Otto Dolinsek}
\date{Oktober 2007}
\parskip12pt %liefert die Absatzeinstellung für das Dokument
\parindent0pt %liefert die Absatzeinrückung für das Dokument
\lstset{language=HTML,basicstyle=\scriptsize,identifiertstyle=\color{black},
keywordstyle=\color{red},stringstyle=\color{black},commentstyle=\color{darkgreen},
backgroundcolor=\color{white},captionpos=b,framexleftmargin=5mm,frame=shadowbox,
rulesepcolor=\color{lightgray},frameround=ftt,numbers=left,numberstyle=\tiny,
stepnumber=1,numbersep=5pt,morekeywords={\lstset}
}
%Definitionen für die Listenzeichen
\usepackage[pdftex,colorlinks=true,linkcolor=blue,urlcolor=blue]{hyperref}
%\includeonly{textstrukturierung}
\tolerance 1414
\hbadness 1414
\emergencystretch 1.5em
\hfuzz 0.3pt
\widowpenalty=10000
\vfuzz \hfuzz
\raggedbottom
\begin{document}
\maketitle %Kapitelüberschrift
\tableofcontents %Inhaltsverzeichnis
\include{..}
\include{literaturangaben}
\end{document}

```

3 Grundlagen der Textformatierung und Strukturierung

3.1 Absatzgestaltung

`\par` Erzeugt einen Absatz, oder eine oder mehrere Leerzeilen.

`\begin{par} ... \end{par}` Markiert den Absatz.

`\parindent0pt` Legt die Einrücktiefe 0pt der ersten Zeile für alle folgenden Absätze fest.

`\parskip12pt` Legt den Abstand 12pt zwischen den nachfolgenden Absätzen fest

`\indent` Sorgt am Anfang eines Absatzes dafür, dass die erste Zeile eingerückt wird.

`\noindent` Sorgt am Anfang eines Absatzes dafür, dass die erste Zeile nicht eingerückt wird.

3.2 Absatz-Ausrichtung

Normalerweise ist sämtlicher Text als Blocksatz ausgerichtet. Links- bzw. rechts ausgerichteter Text erhält mit den Umgebungen

```
\begin{flushleft}  
Links ausgerichteter Text.  
\end{flushleft}  
\begin{flushright}  
Rechts ausgerichteter Text.  
\end{flushright}  
\begin{center}  
Zentrierter Text.  
\end{center}.
```

Links ausgerichteter Text.

Rechts ausgerichteter Text.

Zentrierter Text.

3.3 Vertikale Abstände zwischen Absätzen

- `\smallskip` kleinster vertikaler Abstand zwischen den Zeilen.
- `\medskip` mittlerer vertikaler Abstand zwischen den Zeilen.
- `\bigskip` größter Abstand zwischen den Zeilen.
- `\vspace{1cm}` setzt den vertikalen Abstand in Zentimetern.
- `\vfill` füllt den zur Verfügung stehenden Raum auf der Seite.

Beispiel: Quellcode

```
 Lorem ipsum dolor sit amet
 \smallskip
 At vero eos et accusam ..
 \medskip
 Lorem ipsum dolor sit amet, consetetur ..
 \bigskip
 Lorem ipsum dolor sit amet, consetetur sadipscing elitr ...
 \vspace{2cm}
 At vero eos et accusam et justo duo dolores et ea rebum.
 \vfill
 Duis autem vel eum iriure dolor in hendrerit in vulputate velit ..
```

Ausgabe:

Lorem ipsum dolor sit amet

At vero eos et accusam .. Lorem ipsum dolor sit amet, consetetur ..

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, ..

At vero eos et accusam et justo duo dolores et ea rebum.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit ..

3.4 Umbrüche

- `\newline` Erzeugt einen Zeilenumbruch.
- `\\` Erzeugt einen Zeilenumbruch.
- `\newpage` Erzeugt einen Seitenumbruch.

3.5 Horizontale Wortabstände

- \ erzeugt eine Leerstelle, die nicht verbreitert werden darf. Achte darauf, dass hinter dem Backslash ein Leerzeichen steht.
- \@ kennzeichnet einen nachfolgenden Punkt als Satzende.
- ~ erzeugt eine nicht umbrechbare Leerstelle.
- \, erzeugt eine verkürzte, nicht umbrechbare Leerstelle.
- \quad erzeugt einen einfach vergrößerten Abstand
- \qqquad erzeugt einen vierfach vergrößerten Abstand
- \hspace{1cm} erzeugt ein Abstand von 1 cm.
- \hfill setzt den maximalen Abstand zwischen zwei Worten.

Beispiele:

Seite~1	Seite 1	
l=4\,cm	l=4 cm	
Ein\quad Abstand	Ein Abstand	
Ein\qqquad Abstand	Ein Abstand	
Ein \hspace{5cm} Abstand	Ein	Abstand
Ein\hfill Abstand	Ein	Abstand

3.6 Schriftgröße

- `\emph{Hervorhebung}` *Ein Text mit Hervorhebung.*
- `\itshape{Kursivschrift}` *Ein Text mit Kursivschrift.*
- `\textbf{Text}` Text in **Fettschrift** ausgeben.
- `{\tiny Text}` Text in winzig kleiner Schrift ausgeben.
- `{\scriptsize Text}` Text in sehr kleiner (wie Indizes) Schrift ausgeben.
- `{\footnotesize Text}` Text in kleiner (wie Fußnoten) Schrift ausgeben.
- `{\small Text}` Text in kleiner Schrift ausgeben.
- `{\normalsize Text}` Text in normaler Schrift ausgeben.
- `{\large Text}` Text in großer Schrift ausgeben.
- `{\Large Text}` Text in größerer Schrift ausgeben.
- `{\LARGE Text}` Text in sehr großer Schrift ausgeben.
- `{\huge Text}` Text in riesig großer Schrift ausgeben.
- `{\Huge Text}` Text in gigantisch großer Schrift ausgeben.

3.7 Farben mit dem Paket xcolor

Die Textfarben black, darkgray, lightgray, gray, red, green, blue, cyan, magenta, pink, purple, brown sind definiert!

`\color{Farbe} Text` Bestimmt die aktuelle Vordergrundfarbe.
 Beispiel: `\color{blue} Blauer Text!` **Blauer Text!**

`\textcolor{Farbe}{Text}` Schreibt den Text in der angegebenen Farbe.
 Beispiel: `\textcolor{red}{Roter Text}` **Roter Text**

`\colorbox{Hintergrundfarbe}{Text}` Text mit der angegebenen Hintergrundfarbe.
 Beispiel: `\colorbox{yellow}{Text}` **Text**

`\fcolorbox{Rahmenfarbe}{Hintergrundfarbe}{Text}` Text mit Rahmen- und Rahmenhintergrundfarbe.
 Beispiel: `\fcolorbox{black}{red}{Text}` **Text**

`\fcolorbox{Rahmenfarbe}{Hintergrundfarbe}{\textcolor{Textfarbe}{Text}}` Text im Rahmen
 Beispiel: `\fcolorbox{black}{yellow}{\textcolor{blue}{Text}}` **Text**

`\pagecolor{Farbe}` setzt die Hintergrundfarbe des Dokumentes auf Farbe

Besondere Anmerkung:

`\colorbox` führt keinen automatischen Umbruch durch. Dies kann man beheben, in dem man eine `\parbox` einbettet. Das ganze sieht dann wie folgt aus:

`\colorbox{Hintergrundfarbe}{\parbox{Breite}{.. langer Text ..}}`

Beispiel:

```
\colorbox{lightgray}{\parbox{0.5\textwidth}{Hier der Text\\, der in der Box erscheinen soll!}}
```

Hier der Text
, der in der Box erscheinen soll!

Der Vorteil von `parbox` ist, dass man gleichzeitig die Breite der farbigen Box einstellen kann. Man kann das in den üblichen Angaben (px,pt,cm,...) angeben oder aber auch in einer x-fachen Textweite. `0.5*textwidth` bedeutet hier: Die Box hat die Breite einer halben Seite.

3.8 Tabulatoren

Mit dem Befehl `\=` setzt man die Tabulatorposition, mit `\>` springt man zur nächsten Tabulatorposition und mit `\\` wird die Zeile manuell umgebrochen. Mit dem Befehl `\kill` legt man die Tabulatorenpositionen fest. Diese Zeile wird nicht ausgegeben.

Ein Beispiel:

```
\begin{tabbing}
Optionen \= Platz für Skalierung des Bildes \= Beispiel \kill
Option \> Wirkung \> Beispiel\\
angle \> Drehwinkel des Bildes \> angle=45\\
width \> Breite vorgeben \> width=3cm \\
height \> Höhe vorgeben \> height=35pt\\
scale \> Skalierung des Bildes \> scale=.5\\
\end{tabbing}
```

Option	Wirkung	Beispiel
angle	Drehwinkel des Bildes	angle=45
width	Breite vorgeben	width=3cm
height	Höhe vorgeben	height=35pt
scale	Skalierung des Bildes	scale=.5

3.9 Fußnoten

Fußnoten sind Verweise auf weitere Informationen, die man für wichtig hält.

In \LaTeX gibt es den Befehl `\footnote{Text der Fußnote}`. Der Fußnotenbefehl sollte immer direkt nach dem vorangehenden Wort ohne Leerzeichen stehen. Im normalen Fließtext setzt \LaTeX kleine hochgestellte Ziffern als Fußnotenmarkierung. Allgemein sollte man sparsam mit Fußnoten umgehen.

Mit dem Paket `footmisc` können viele andere Einstellungen angepasst werden.

Beispiel:

```
Mater Papirii pueri, qui cum parente\footnote{parens,-ntis m: = pater,-tris m} suo
in curia fuerat, percontata est filium, quidnam in senatu
patres\footnote{patres, -um m. = senatores,-um m.} egissent.
Puer respondit (sibi) tacendum esse neque id dici licere.\
Mulier fit audiendi cupidior; secretum rei et silentium pueri animum eius ad
inquirendum everbera(n)t: quaerit igitur
compressius\footnote{compressius: drängender} violentiusque.
```

Mater Papirii pueri, qui cum parente¹ suo in curia fuerat, percontata est filium, quidnam in senatu patres² egissent. Puer respondit (sibi) tacendum esse neque id dici licere.

Mulier fit audiendi cupidior; secretum rei et silentium pueri animum eius ad inquirendum everbera(n)t: quaerit igitur compressius³ violentiusque.

¹parens,-ntis m: = pater,-tris m

²patres, -um m. = senatores,-um m.

³compressius: drängender

3.10 Farben mischen!

`\definecolor{'Name'}{'Modell'}{'Spezifikation '}` Farben werden über die Werte zwischen 0 und 1 gesetzt.

Beispiel 1:

```
\definecolor{zufall}{rgb}{1,0.5,0.5}
\textcolor{zufall}{Text mit Farbe zufall}
Text mit der Farbe zufall
```

Beispiel 2:

```
\definecolor{olive}{cmyk}{0.64,0,0.95,0.4}
\textcolor{olive}{Text mit Farbe olive}
Text mit der Farbe olive
```

Beispiel 3:

```
\textcolor{-olive}{Text mit der Komplementärfarbe olive}
Text mit der Komplementärfarbe olive
```

Farbmischung mit Mischoperator ! Farbe mit 10% Rot, 30% Grün, 60% Blau

Beispiel 1:

```
{\color{red!10!green!30!blue} Textfarbe zu 10\%Rot, 30\%Grün und 60\%Blau}
Textfarbe zu 10\%Rot, 30\%Grün und 60\%Blau
```

Beispiel 2:

```
{\color{-red!10!green!30!blue} Komplementärfarbe}
Komplementärfarbe
```

3.11 Textmaskierung

```
\begin{verbatim} ... \end{verbatim}
```

Beschreibung:

Fügt den umschlossenen Text in das formatierte Dokument ein, ohne eventuell enthaltenen LaTeX-Befehle und Befehlszeichen zu interpretieren.

```
\verb+Text+
```

Beschreibung:

Fügt den angegebenen Text in das formatierte Dokument ein, ohne eventuell enthaltenen LaTeX-Befehle und Befehlszeichen zu interpretieren. Anstelle der +-Zeichen kann jedes andere Zeichen verwendet werden (aber sinnvollerweise nicht *); das zur Abgrenzung verwendete Zeichen darf dabei natürlich nicht im Text erscheinen. Beispiel:

Der `\LaTeX`-Befehl erzeugt das `LATEX`-Logo.

Spezielle Zeichen müssen maskiert werden, wenn sie im Text als Zeichen erscheinen sollen.

Beispiele:

```
\ $\backslash$ maskiert das Backslashzeichen.  
& \& maskiert das kaufmännische Und (Et-Zeichen).  
} \} maskiert die geöffnete geschwungene Klammer.  
{ \{ maskiert die geschlossene geschwungene Klammer.
```

3.11.1 Textmaskierung mit dem Paket fancyvrb

```
\begin{Verbatim} ... \end{Verbatim}
```

Beschreibung:

Fügt den umschlossenen Text in den vordefinierten Rahmen ein, ohne eventuell enthaltenen LaTeX-Befehle und Befehlszeichen zu interpretieren.

Folgende Parameter können verwendet werden:

frame(none, leftline, topline, bottomline, lines, single): type of frame around the verbatim environment (Default: none|no frame). With leftline and single modes, a space of a length given by the LATEX `\fboxsep` macro is added between the left vertical line and the text.

framerule (dimension): width of the rule of the frame (Default: 0.4pt if framing specified).

rulecolor (color command): color of the frame rule, expressed in the standard LATEX way (Default: black).

fillcolor (color command): color used to fill the space between the frame and the text (its thickness is given by `framesep`) (Default: none|no color).

```
\begin{Verbatim}
[frame=single, framerule=1mm, framesep=3mm, rulecolor=\color{red}, fillcolor=\color{yellow}]
Hier können Latex Steuerzeichen stehen wie \ \ % usw!
Der Zeienumbruch wird erkannt.
\end{Verbatim}
```



```
Hier können Latex Steuerzeichen stehen wie \ \ % usw!
Der Zeienumbruch wird erkannt.
```

Anmerkung: Die der Verbatim-Umgebung kann nicht geändert werden.

3.11.2 Source Code einfügen

Das Einbetten von Sourcecode-Elementen in LaTeX mittels `\verb` bzw. durch die `verbatim`-Umgebungen sind Lösungen. Leider kann es zu Problemen mit den Zeilen- und Seitenumbrüchen kommen. Abhilfe schafft das Paket `listings`. Dabei können spezielle Parameter gesetzt werden (Syntaxhervorhebung, Zeilennummerierung, Schriftgröße, Rahmenfarbe, Hintergrundfarbe, ..). Die gängigsten Programmiersprachen werden unterstützt (C, C++, HTML, Ant, Basic, Delphi, Java, PHP, ..). Im Kopf muss zunächst das Paket `listings` geladen und konfiguriert werden.

```
\definecolor{hellgelb}{rgb}{1,1,0.8}
\lstset{language=[LaTeX]TeX, basicstyle=\scriptsize, identifierstyle=\color{black},%
keywordstyle=\color{blue}, stringstyle=\color{red}, commentstyle=\color{green},%
frame=single, backgroundcolor=\color{hellgelb}, captionpos=b}%
```

Listing 1: Einbinden des Paketes `listings`

Es gibt drei Möglichkeiten Source-Code in das Dokument einzufügen:

- Im Fließtext.
Beispiel: `\lstinline|echo "hello world"| echo "hello world"`
- Als eigenständiger Sourcecode.

```
\begin{lstlisting}[title=Ausgabe als Sourcecode]
\begin{flushright}
Rechts ausgerichteteter Text.
\end{flushright}
\end{lstlisting}
```

```
\begin{flushright}
Rechts ausgerichteteter Text.
\end{flushright}
```

Ausgabe als Sourcecode

- Als eigenständige externe Datei.

```
\lstset{frameround=fttt, rulesepcolor=\color{white}}
\lstinputlisting[language=PHP, frame=trBL, label=lst:beispielcode, title=PHP]{switch.php}
```

```
switch ($a) {
case 1: echo "Der_Wert_ist_1"; break;
case 2: echo "Der_Wert_ist_2"; break;
default: echo "Der_Wert_ist_weder_1_noch_2"; break;
}
```

Listing 2: PHP

Quellcodebeispiel Dieses Beispiel soll zeigen, wie man die `lstlisting` -Box mit `lstset{..}` konfigurieren kann. Nähere Informationen gibt es in der Package Beschreibung der Listings Box.

```
\lstset {language=PHP, basicstyle=\scriptsize, identifizierstyle=\color{black},
keywordstyle=\color{red}, stringstyle=\color{black}, commentstyle=\color{darkgreen},
backgroundcolor=\color{white}, captionpos=b, framexleftmargin=5mm, frame=shadowbox,
rulesepcolor=\color{lightgray}, frameround=fttt,
numbers=left, numberstyle=\tiny, stepnumber=1, numbersep=5pt,
morekeywords={\$_POST,<?php,?>}
}
\begin {lstlisting} [title=PHP Script: formular.php, frame=trBL, firstnumber=1]
..
\begin {lstlisting}
```

Die obigen `\lstset`-Einstellungen führen zu dieser Ausgabe!

```
1 <?php
2 #####Einlesen der Formulardaten#####
3 $vorname=$_POST[ 'vorname' ];
4 $nachname=$_POST[ 'nachname' ];
5 $password=$_POST[ 'password' ];
6 $schultyp=$_POST[ 'schultyp' ]; //Select
7 $arbeitstage=$_POST[ 'arbeitstage' ]; //Multiselect
8 $geschlecht=$_POST[ 'geschlecht' ]; //Optionsschalter
9 $wahl=$_POST[ 'wahl' ]; // Kontrollkästchen
10 $text=$_POST[ 'text' ];
11 $geheim=$_POST[ 'geheim' ];
12 #####Formulardaten zu einem String zusammenbauen#####
13 $message="Vorname: _$vorname<br/>\n";
14 $message=$message."Nachname: _$nachname<br/>\n";
15 $message=$message."Passwort: _$password<br/>\n";
16 $message=$message."Schultyp: _$schultyp<br/>\n";
17 foreach ( $arbeitstage as $value ) {
18     $message=$message."Gewählter_Arbeitstag: _$value<br/>\n";
19 }#end foreach
20 $message=$message."Geschlecht: _$geschlecht<br/>\n";
21 foreach ( $wahl as $value ) {
22     $message=$message."Gewähltes_Kontrollkästchen: _$value<br/>\n";
23 }#end foreach
24 $message=$message."Nachricht_im_Textfeld: _$text<br/>\n";
25 $message=$message."Geheime_Nachricht: _$geheim<br/>\n";
26 echo $message;
27 echo "<hr/>";
28 ?>
```

PHP Script: formular.php

3.12 Binde- und Gedankenstriche

Es gibt vier Arten von horizontalen Strichen im Text:

- den *Divis* oder *Bindestrich*, der durch ein einfaches Minus (-) erzeugt wird,
- den *Halbgeviertstrich* oder *Gedankenstrich*, der durch ein doppeltes Minus (--)erfordert,
- den *Geviertstrich* oder *langen Gedankenstrich* der ein dreifaches Minus (---) erfordert und
- das *Minuszeichen*, welches mit $\$-\$$ erzeugt wird. Das Zeichen \$ schaltet den Mathematikmodus ein und aus.

Der halbgeviertstrich dient im Deutschen nicht nur als Gedankenstrich, sondern auch als Bereichsstrich zwischen zwei Zahlen oder Datenwerten. Der lange Gedankenstrich wird im Deutschen nicht benutzt, im Englischen ohne Leerzeichen!

Beispiele:

11--19 Uhr	11-19 Uhr
Klagenfurt--Graz	11-19 Uhr
wahr -- oder falsch?	wahr – oder falsch?
true---or false?	true—or false?

4 Listen

4.1 Listen mit Zwischenabstand

4.1.1 Unnummerierte Aufzählungsliste

```
\begin{itemize}  
\item Unnummerierte Aufzählungsliste.  
\item Aufzählungsliste mit einer Freizeile zwischen den Aufzählungen.  
\end{itemize}
```

- Unnummerierte Aufzählungsliste
- Aufzählungsliste mit einer Freizeile zwischen den Aufzählungen

4.1.2 Nummerierte Aufzählungslisten

```
\begin{enumerate}  
\item Nummerierte Aufzählungsliste mit einer Freizeile zwischen den Aufzählungen.  
\item Als Aufzählungszeichen fungieren Zahlen.  
\end{enumerate}
```

1. Nummerierte Aufzählungsliste mit einer Freizeile zwischen den Aufzählungen
2. Als Aufzählungszeichen fungieren Zahlen

4.1.3 Beschreibungsliste

```
\begin{description}  
\item[Punkt 1] Aufzählungsliste mit einer Freizeile zwischen den Aufzählungen  
\item[Punkt2] Als Aufzählungszeichen fungiert der angegebene Text.  
\end{description}
```

Punkt 1 Aufzählungsliste mit einer Freizeile zwischen den Aufzählungen

Punkt2 Als Aufzählungszeichen fungiert der angegebene Text.

4.1.4 Verschachtelte Listen

Listen können beliebig geschachtelt werden.

Ein Beispiel:

```
\begin{enumerate}
\item Die äußere Schachtel
  \begin{itemize}
  \item Der Inhalte der inneren Schachtel.
  \item Das Zeichen hängt von der Verschachtelungstiefe ab.
    \begin{itemize}
    \item Der Inhalte der inneren Schachtel.
    \item Das Zeichen hängt von der Verschachtelungstiefe ab.
    \end{itemize}
  \end{itemize}
\end{enumerate}
\item Der nächste Punkt!
\end{enumerate}
```

1. Die äußere Schachtel

- Der Inhalte der inneren Schachtel.
- Das Zeichen hängt von der Verschachtelungstiefe ab.
 - Der Inhalte der inneren Schachtel.
 - Das Zeichen hängt von der Verschachtelungstiefe ab.

2. Der nächste Punkt!

4.2 Listen ohne Zwischenabstand mit dem Package mdwlist

4.2.1 Aufzählungslisten

```
\begin{itemize*}  
\item Aufzählungsliste ohne Freizeile zwischen den Aufzählungen  
\item Als Aufzählungszeichen fungiert der Punkt  
\end{itemize*}
```

- Aufzählungsliste ohne Freizeile zwischen den Aufzählungen
- Als Aufzählungszeichen fungiert der Punkt

4.2.2 Nummerierte Aufzählungslisten

```
\begin{enumerate*}  
\item Nummerierte Aufzählungsliste ohne Freizeile zwischen den Aufzählungen  
\item Als Aufzählungszeichen fungieren Zahlen  
\end{enumerate*}
```

1. Nummerierte Aufzählungsliste ohne Freizeile zwischen den Aufzählungen
2. Als Aufzählungszeichen fungieren Zahlen

4.3 Flexible Listen mit dem Package paralist

4.3.1 Flexible Aufzählungslisten mit dem Package paralist

```
\begin{itemize}[$\triangleright$]  
\item Aufzählungsliste ohne Freizeile zwischen den Aufzählungen.  
\item Als Aufzählungszeichen fungiert das gewählte Zeichen.  
\end{itemize}
```

- ▷ Aufzählungsliste mit Freizeile zwischen den Aufzählungen.
- ▷ Als Aufzählungszeichen fungiert das gewählte Zeichen.

4.3.2 Flexible Listen mit dem Package paralist ohne Zeilenabstand

```
\begin{compactitem}[$\triangleright$]  
\item Nummerierte Aufzählungsliste ohne Freizeile zwischen den Aufzählungen  
\item Als Aufzählungszeichen fungiert das gewählte Zeichen.  
\end{compactitem}
```

- ▷ Nummerierte Aufzählungsliste mit einer Freizeile zwischen den Aufzählungen
- ▷ Als Aufzählungszeichen fungieren Zahlen

4.4 Inline Listen

```
Listen mit der inpara-Umgebung für  
begin{inparaenum}[(1)]  
  \item \emph{inparaitem} für Aufzählungen  
  \item \emph{inparaenum} für Nummerierungen und  
  \item \emph{inparadesc} für Beschreibungslisten.  
end{inparaenum}
```

Listen mit der inpara-Umgebung für (1) *inparaitem* für Aufzählungen (2) *inparaenum* für Nummerierungen und (3) *inparadesc* für Beschreibungslisten.

4.5 Allgemeine Listen mit der list Umgebung

Es folgt eine Auflistung der Parameter, die das Listenformat beeinflussen. Das Setzen dieser Werte ist relativ aufwendig. Am komfortabelsten ist es, wenn statt der itemize-Umgebung die list-Umgebung verwendet wird. Hier ist es möglich mit Listenparametern die Gestalt der Liste zu verändern.

```
\begin{list}{marke}{definitionen}
\item text
\item text
\end{list}
```

Wichtige Parameter:

<code>\topsepAbstand</code>	Listenabstand vor und nach der Liste.
<code>\parsepAbstand</code>	Itemabstand innerhalb der Liste.
<code>\itemsepAbstand</code>	Zusätzlicher Abstand zwischen den Items.
<code>\leftmarginAbstand</code>	Einrückung des linken Textrandes.
<code>\rightmarginAbstand</code>	Einrückung des rechten Textrandes.
<code>\labelsepAbstand</code>	Abstand rechte Seite des Labels zum linken Textrand.
<code>\labelwidthAbstand</code>	Breite der Labels.
<code>\listparindentAbstand</code>	Einrückung der ersten Zeile eines Absatzes.
<code>\makelabel</code>	Definiert das Listenzeichen.

Beispiel:

```
\newcounter{line} Text vor der allgemeinen Liste ..
\begin{list}{\bf Zeile \arabic{line}:}{\usecounter{line}\it \parsep0pt \itemsep0pt}
\item Text in der allgemeinen Liste
\item Text in der allgemeinen Liste
\item Text in der allgemeinen Liste
\end{list}
Text nach der Liste ..
```

Text vor der allgemeinen Liste ..

Zeile 1: *Text in der allgemeinen Liste*

Zeile 2: *Text in der allgemeinen Liste*

Zeile 3: *Text in der allgemeinen Liste*

Text nach der Liste ..

4.6 Listenmarken definieren

Am besten definiert man neue Listenzeichen in der Präambel, also vor `\begin{document}`. Das geht mit dem Befehl `\newcommand`.

```

\renewcommand{\labelitemi}{\blacksquare}
\renewcommand{\labelitemii}{\bullet}
\renewcommand{\labelitemiii}{\blacktriangleright}
\renewcommand{\labelitemiv}{\gg}

\begin{itemize*}
\item Text in der Liste, erste Ebene
  \begin{itemize}
    \item Text in der Liste, zweite Ebene
      \begin{itemize}
        \item Text der dritten Ebene
          \begin{itemize}
            \item Text der vierten Ebene
          \end{itemize}
        \end{itemize}
      \end{itemize}
    \end{itemize}
  \end{itemize}
\item Text in der Liste, zweite Ebene
\end{itemize}
\item Text in der Liste, erste Ebene
\begin{itemize}
\item Text in der Liste, zweite Ebene
\end{itemize}
\item Text in der Liste, erste Ebene
\end{itemize*}

```

- Text in der Liste, erste Ebene
 - Text in der Liste, zweite Ebene
 - ▶ Text in der Liste, dritte Ebene
 - » Text in der Liste, vierte Ebene
 - ▶ Text in der Liste, dritte Ebene
 - Text in der Liste, zweite Ebene
- Text in der Liste, erste Ebene
 - Text in der Liste, zweite Ebene
- Text in der Liste, erste Ebene

5 Tabellen

Tabellen ordnen Daten in Zeilen und Spalten. Tabelleninhalte werden von Latex nicht sortiert.

Tabellen in wissenschaftlichen arbeiten haben einige Grundregeln!

- Einfache Strukturen sind zu bevorzugen.
- Es gibt keine vertikale Linien.
- Es gibt keine doppelten Linien.
- Zwischen den Datenzeilen gibt es keine Linien.
- Am Beginn und am Ende der Tabelle wird eine etwas dickere Linie gezogen.
- Einheiten für Werte gehören in den Tabellenkopf.
- Keine Wiederholungszeichen verwenden.

5.1 Tabellen definieren

```
\begin{tabular}{|l|c|r|}
\hline
Spalte1,1 ist die erste Spalte&
Spalte1,2 die mittlere&Spalte1,3 die äußere\\
Spalte2,1 & Spalte2,2 & Spalte2,3\\
Spalte3,1 & & Spalte3,3\\
\hline
\end{tabular}
```

Spalte 1,1 ist die erste Spalte	Spalte1,2 die mittlere	Spalte1,3 die äußere
Spalte2,1	Spalte2,2	Spalte2,3
Spalte3,1		Spalte3,3

Leider bricht der Text in den einzelnen Spalten nicht um. Die Spalten sind so breit wie der breiteste Text.

p<width> erzeugt mehrzeilige, umbrechende Spalten. Der Text bricht nach der Breite <width> um.

```
\begin{tabular}{|p{3cm}|p{4cm}|p{2cm}|}
\hline
Das ist ein Beispiel für einen dreispaltigen Text.&
Jede Spalte hat die vordefinierte Breite.& Der Umbruch erfolgt automatisch \\
\hline
\end{tabular}
```

Das ist ein Beispiel für einen dreispaltigen Text.	Jede Spalte hat die vordefinierte Breite.	Der Umbruch erfolgt automatisch
--	---	---------------------------------

5.2 Tabelle in der Gleitumgebung

Die Gleitumgebung ist ein Container, der eine Nummerierung hat und eine Beschriftung ausgibt.

Bei Tabellen kommt die Beschriftung vor dem Inhalt, weil das Lesen der Tabellen eher sequentiell von oben nach unten erfolgt.

```

\begin{table}[htbp]
\centering
\caption{Tabellenüberschrift}
\label{tab:Beispieltabelle}
\begin{tabular}{|p{3cm}|p{4cm}|p{2cm}|}
\hline
Das ist ein Beispiel für einen dreispaltigen Text.&
Jede Spalte hat die vordefinierte Breite.& Der Umbruch erfolgt automatisch \\
\hline
\end{tabular}
\end{table}

```

Tabelle 1: Tabellenüberschrift

Das ist ein Beispiel für einen dreispaltigen Text.	Jede Spalte hat die vordefinierte Breite.	Der Umbruch erfolgt automatisch
--	---	---------------------------------

Das ist ein Beispiel für eine Tabelle mit Überschrift. h steht für *here*, t für *top*, b für *bottom* und p für *page*. Der erste Buchstabe in der Tabellendefinition hat die höchste Priorität. Hätte die Tabelle nicht auf dieser Seite Platz, würde sie ganz oben auf der nächsten Seite stehen usw.

Mittels des Befehls `\label{lab}` kann im Text mit dem Befehl `\ref{lab}` auf diese Tabelle verwiesen werden. `\centering` macht keinen Abstand zwischen der Tabellenüberschrift und der Tabelle.

5.3 Spaltenabstände

Mit dem Befehl `@{\hspace{10mm}}` wird der rechte Abstand zur nächsten Spalte auf 10mm gesetzt.

```
\begin{tabular}[htbp]{|l@{}|l@{\hspace{15mm}}|l@{\hspace{20mm}}|}
\hline
Abstände zwischen & den Zellen mit dem & Befehl \verb+@{Abstand}+.\\
\hline
\end{tabular}
```

Abstände zwischen	den Zellen mit dem	Befehl <code>@{Abstand}</code> .
-------------------	--------------------	----------------------------------

5.4 Spaltenzwischenräume

Mit dem Befehl `@{!zwischenraum}` kann man Zeichen in in den Spaltenzwischenraum schreiben. Der Befehl erfordert das Paket **array**.

```
\begin{tabular}[htbp]{|l!{\$ \leftrightarrows }|l!{\$ \Leftarrow }|l!{\$ \Rightarrow }|}
\hline
Spaltenzwischenraum mit & Symbolen odet& Pfeilen beschreiben.\\
\hline
\end{tabular}
```

Spaltenzwischenraum mit \leftrightarrow	Symbolen odet \Leftarrow	Pfeilen beschreiben. \Rightarrow
---	----------------------------	------------------------------------

5.5 Mehrspaltige Zellen

Für Zellen, die sich über mehrere Spalten erstrecken, stellt \LaTeX den Befehl `\multicolumn` zur Verfügung.

```
\begin{tabular}{|1|1|1|1|}  
\hline  
Ein &nettes &Beispiel \\  
\hline  
\multicolumn{2}{|c|}{einer kleinen} &Tabelle \\  
\hline  
\end{tabular}
```

Ein	nettes	Beispiel
einer kleinen	Tabelle	

5.6 Mehrzeilige Zellen

Für mehrzeilige Zellen gibt es ein eigenes Paket namens **multirow**, welches den Befehl **\multirow** zur Verfügung stellt.

```
\begin{tabular}{|1|1|1|1|}
\hline
\multicolumn{3}{|c|}{Textverarbeitungsprogramme}\\
\hline
\multirow{3}{*}{selbst erlernt} & Word & WinXP\\
& Latex & Linux \\
& QuarkXpress & Apple\\
\hline
\end{tabular}
```

Textverarbeitungsprogramme		
selbst erlernt	Word	WinXP
	Latex	Linux
	QuarkXpress	Apple

5.7 Vertikale Ausrichtung von Zellen

Man benötigt das Paket **array**. Es gibt folgende Spaltentypen

m{<breite>} Erzeugt einen vertikal zentrierten Text der Breite <breite>.

b{<breite>} Erzeugt einen von oben nach unten ausgerichteten Text.

p{<breite>} Erzeugt einen von der Grundlinie nach oben ausgerichteten Text.

```
\begin{tabular}{|l|p{2cm}|m{2cm}|b{2cm}|}
\hline
Der Text wird ausgerichtet.&
Der Text wird ausgerichtet.&
Der Text wird ausgerichtet.&
Der Text wird ausgerichtet.\\
\hline
\end{tabular}
```

Der Text wird ausgerichtet.			
-----------------------------	-----------------------------	-----------------------------	-----------------------------

Tabelle 2: Vertikale Ausrichtung

6 Formeln

Formeln sind mathematische Ausdrücke, die nichts mit der automatischen Berechnung von Werten zu tun haben. Das Paket *amsmath* (American Mathematical Society, 2002) beherrscht genügend mathematische Konstrukte.

Formeln werden im Mathematikmodus gesetzt. Bei der Texteingabe kann man zwischen Mathematik- und Textmodus wechseln.

Es gibt zwei Mathematikmodi:

- abgesetzte Formeln und
- Formeln im Text.

Eine abgesetzte Formel ist zum Beispiel:

$$U_{ind} = \frac{d\phi}{dt} = \dot{\phi} \tag{1}$$

Formeln im Fließtext $a = \frac{dv}{dt} = \dot{v} = \ddot{s}$ müssen mit der vertikalen Platzverteilung besser umgehen als normaler Text. Formeln im Text stehen zwischen zwei Dollarzeichen $\$. . . \$$ (z.B.: $\$a=\frac{dv}{dt}=\dot{v}=\ddot{s}\$$).

Man sollte nur kurze mathematische Formeln innerhalb eines Absatzes schreiben. Für längere und vertikal anspruchsvolle Terme sollte man abgesetzte Formeln verwenden.

6.1 Einfache Formeln

Als einfaches Beispiel erstellen wir die Lösungsformel für quadratische Gleichungen.

Seien a , b , und c mit $a, b, c \in \mathbb{R}$ die Koeffizienten der quadratischen Gleichung $a \cdot x^2 + b \cdot x + c = 0$. Diese Gleichung hat für $a \neq 0$ folgende Lösungen:

```
\begin{equation}
\label{eq:quadratischegleichung}
x_{1,2}=\frac{-b \pm \sqrt{b^2-4 \cdot a \cdot c}}{2 \cdot a}
\end{equation}
```

Seien a , b , und c mit $a, b, c \in \mathbb{R}$ die Koeffizienten der quadratischen Gleichung $a \cdot x^2 + b \cdot x + c = 0$. Diese Gleichung hat für $a \neq 0$ folgende Lösungen:

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} \tag{2}$$

6.2 Einzeilige Formeln

Neben der `equation`-Umgebung gibt es noch weitere Umgebungen für Formeln, die ihre Stärken ausspielen, wenn Formeln mehrzeilig sind oder mehrere Formeln unterzubringen sind. Alle Umgebungen gibt es mit und ohne Stern, wie etwa `equation*` und `equation`. Mit Stern erhalten die Formeln keine Nummerierung und können nicht als Ziel für Querverweise dienen.

Die einfachste Umgebung ist `equation`. Sie dient für normale einzeilige Formeln. Ein Beispiel:

```
\begin{equation}
t=t_1+t_2=\frac{\sqrt{a^2+x^2}}{c_1}+\frac{\sqrt{b^2+(1-x)^2}}{c_2}
\end{equation}
\begin{equation*}
t=t_1+t_2=\frac{\sqrt{a^2+x^2}}{c_1}+\frac{\sqrt{b^2+(1-x)^2}}{c_2}
\end{equation*}
```

$$t = t_1 + t_2 = \frac{\sqrt{a^2 + x^2}}{c_1} + \frac{\sqrt{b^2 + (l - x)^2}}{c_2} \tag{3}$$

$$t = t_1 + t_2 = \frac{\sqrt{a^2 + x^2}}{c_1} + \frac{\sqrt{b^2 + (l - x)^2}}{c_2}$$

6.3 Mehrzeilige, ausgerichtete Formeln

mit der integrierten `split`-Umgebung. Wie alle der folgenden Umgebungen wird am `&`-Zeichen vor Relationszeichen ausgerichtet.

Beispiel: Ausrichtung am = Zeichen einer mehrzeiligen Formel

```
\begin{equation}%* mit Nummerierung
\begin{split}
x_{1,2}&=\frac{-b \pm \sqrt{b^2-4 \cdot a \cdot c}}{2 \cdot a} \\
U_{ind}&=\frac{d\phi}{dt}=\dot{\phi} \\
a&=\frac{dv}{dt}=\dot{v}=\ddot{s}
\end{split}
\end{equation}
```

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a}$$

$$U_{ind} = \frac{d\phi}{dt} = \dot{\phi} \tag{4}$$

$$a = \frac{dv}{dt} = \dot{v} = \ddot{s}$$

7 Grafik

Grafiken können in verschiedenen Dateiformaten vorliegen.

7.1 Grafik einbinden

Mit dem Paket **graphicx** werden Grafiken eingebunden. PdfLaTeX kann die Dateiformate PDF, PNG, JPEG verarbeiten.

Mit dem Befehl `\includegraphics[Option]{Dateiname}` werden Grafiken eingebunden. Folgende Optionen stehen zur Verfügung:

Option	Wirkung
<code>angle</code>	Winkel der Drehung des Bildes (Beispiel: <code>angle=45</code>)
<code>width</code>	Breite vorgeben (Beispiel: <code>width=3cm</code> , <code>width=0.5\textwidth</code>)
<code>height</code>	Höhe vorgeben (Beispiel: <code>height=3cm</code>)
	Bei nur einer Angabe (Breite, Höhe) bleibt das Seitenverhältnis des Bildes erhalten!
<code>scale</code>	Skalierung (Beispiel: <code>scale=.2</code>)

```
\begin{figure}[htbp]
  \centering
  \includegraphics[width=0.2\textwidth, angle=45]{schule.jpg}
  \caption{BG/BRG Lerchenfeld}
  \label{fig:schule}
\end{figure}
```



Abbildung 1: BG/BRG Lerchenfeld

7.2 Einheiten

Gültige Einheiten sind:

pt	Punkte, die Standardeinheit von PostScript. (1pt=1/72 Zoll=0,0353cm, 1cm=28,345pt)
cm	Zentimeter
mm	Millimeter
in	Zoll, 1 Zoll=2,54cm.

7.3 Grafiken nebeneinander

Ein Beispiel für zwei Grafiken mit gleicher Bildunterschrift.

```
\begin{figure}[htbp]
  \centering
  \includegraphics[width=0.2\textwidth,angle=0]{schule.jpg}
  \hspace{3cm}
  \includegraphics[width=0.2\textwidth,angle=45]{schule.jpg}
  \caption{BG/BRG Lerchenfeld}
  \label{fig:zweischulen}
\end{figure}
```



Abbildung 2: BG/BRG Lerchenfeld

7.4 Text umflossene Grafiken

Mit dem Paket **wrapfig** lassen sich Bilder mit Text umfließen. Die Einbindung erfolgt mit dem folgendem Befehl:

```
\usepackage{graphicx}
\usepackage{wrapfig}
.
.
\begin{wrapfigure}[lines]{position}{breite}
\includegraphics{Bild}
\caption{Bildbeschreibung}
\end{wrapfigure}
```

Ein Beispiel:

```
\begin{wrapfigure}{r}{5cm}
\centering
{\includegraphics{schule.jpg}}
\caption{Lerchenfeld}
\end{wrapfigure}
```

Zu sehen ist, dass sowohl das Paket `wrapfig`, als auch `graphics` in der Präambel geladen werden müssen. Direkt vor dem Absatz, in dem die Grafik eingefügt werden soll, muss die `wrapfigure`- Umgebung gesetzt werden, die mit `\begin{wrapfigure}` begonnen wird. Dabei können optional die Zeilen angegeben werden, die über der Grafik stehen sollen. Anschließend muss festgelegt werden, ob die Grafik rechte `r` und links `l` im Text stehen soll und wie viel Platz `breite` dafür reserviert werden soll. In der Umgebung muss dann das Einfügen des Bildes erfolgen, wobei Manipulationen des selbigen stattfinden können. In diesem Beispiel wird noch eine Bildunterschrift mittels dem Befehl `\caption` angegeben. Abgeschlossen wird das ganze mit



Abbildung 3: Lerchenfeld

8 Literaturangaben

Mit der **thebibliography**-Umgebung kann man ein Literaturverzeichnis erzeugen. Darin beginnt jede Literaturangabe mit `\bibitem`. Als Parameter wird ein Name vereinbart, unter dem die Literaturstelle im Text zitiert werden kann. Anschließend folgt der Text der Literaturangabe. Die Nummerierung erfolgt automatisch. Der Parameter bei `\begin{thebibliography}` gibt die maximale Breite der Nummerierung an, also z. B. 99 für maximal zweistellige Nummern. Im Text zitiert man die Literaturstelle mit dem Befehl `\cite` und dem vereinbarten Namen als Argument.

Beispiel:

```
Literaturangaben\cite{srs} sind wichtige Punkte wissenschaftlicher Arbeiten.  
  
\begin{thebibliography}{99}  
\bibitem{srs}  
Sextl Raab Streeruwitz: \emph{Physik 1,} Schulbuch-Nr. 2515,  
ISBN 3-2209-00803-5, 2. Auflage 1992  
\end{thebibliography}
```

Literaturangaben[1] sind wichtige Punkte wissenschaftlicher Arbeiten.

Literatur

- [1] Sextl Raab Streeruwitz: *Physik 1*, Schulbuch-Nr. 2515, ISBN 3-2209-00803-5, 2. Auflage 1992

9 Satzspiegel

Zum freien Setzen des Satzspiegels gibt es das `geometry`-Paket. Das A4-Format hat die Größe 21cmx29.7cm. Bei einseitigen Dokumenten beträgt das Verhältnis vom linken zum rechten Rand 1:1.

Die Parameter `top`, `bottom`, `left`, und `right` bilden den Seitenrand, `width` und `height` definieren den Textkörper.

Das Verhältnis vom oberen zum unteren Seitenrand sollte 2:3 betragen.

Das Verhältnis vom gesamten Textkörper zur Papierhöhe ungefähr 70%.

Das folgende Beispiel demonstriert den Einsatz des `geometry`-Pakets.

Für den Textkörper (ohne Kopf- und Fußzeile) verbleiben in diesem Beispiel 20cm.

```
\documentclass[a4paper, headsepline]{scrartcl}
\usepackage[a4paper, includehead, includefoot, left=3cm, width=15cm,
top=2cm, bottom=3cm, headheight=2cm, headsep=1cm, footskip=1.7cm]{geometry}
%A4 Format: 21cmx29,7cm
```

Freie Anpassung des Satzspiegels

10 Kopf- und Fußzeile

Man schreibe in die Präampel folgende 2 Zeilen.

```
\pagestyle{plain}
%Zentrierte Seitennummerierung in der Fußzeile!
\pagestyle{headings}
%Standardkopfzeile – zentrierte Ausgabe der Section!
```

Eigene Kopf- und Fußzeilen

Für weitere spezielle Einstellungen dient das Paket `scrpage2`!

`\ihead` repräsentiert den linken Teil, `\ohead` den rechten Teil und `\chead` den mittleren Teil der Kopfzeile. Analoges gilt für die Fußzeile.

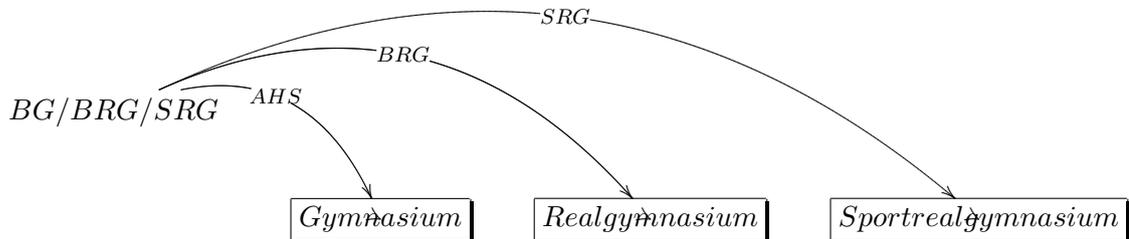
`\headmark` fügt standardgemäß die aktuelle Section des Dokumentes ein, während mit `\pagemark` die Seitennummerierung ausgegeben wird.

Das folgende Beispiel demonstriert die Anwendung des Paketes `scrpage2`.

```
\usepackage[autooneside,automark]{scrpage2}%Kopf-und Fußzeile
\pagestyle{scrheadings}
\clearscrheadfoot
%\automark[section]{subsection}
\ohead[]{\headmark} %rechts
\chead[]{\HTML-CSS-PHP} %mittig
\ihead[]{\includegraphics[height=2cm]{logo.jpg}} %links
\ifoot[]{\today} %links
\cfoot[]{\Mag. Otto Dolinsek} %mittig
\ofoot[]{\pagemark} %rechts
%\renewcommand{\headfont}{\normalfont\sffamily\bfseries}
%Kopf- und Fußzeile umformatieren
```

Eigene Kopf -und Fußzeilen

11 Kommutative Diagramme



Zur Ausführung des Quellcodes muss das Paket `\usepackage[all]{xy}` eingebunden werden.

11.1 xymatrix

Die Objekte des Diagrammes ordnet man in einem Rechteckschema (`xymatrix`) an. Für die Pfeile werden keine eigenen Tabellenzellen reserviert. Die Tabellenzellen werden wie üblich mittels `&` getrennt. Die Tabellenzeilen beendet man mit `\\`. Eine leere Zelle wird durch `\:` dargestellt.

Beispiel:

```
\xymatrix{A & B & C \\ \: & D & E}
```

$$\Rightarrow \begin{array}{ccc} A & B & C \\ & D & E \end{array}$$

11.2 Pfeile

Um die Zellen mit Pfeilen zu verbinden, fügt man den Kontrollcode `\ar[direction]` ein. `direction` kann eine Kombination der Buchstaben `u`, `d`, `l`, `r` sein. Eine Verbindung der Zelle11 mit der Zelle23 erfolgt mit `\ar[dr]` (down, right,right).

Beispiel:

```
\xymatrix{A \ar[dr] & B & C \\ D & E & F}
```

$$\Rightarrow \begin{array}{ccc} A & B & C \\ D & E & F \end{array}$$

11.3 Beschriftungen der Pfeile

Die Beschriftungen der Pfeile werden an `\ar[direction]` mit folgenden Zeichen angehängt:

- `^` - Beschriftung auf dem Pfeil
- `_` - Beschriftung unter dem Pfeil
- `|` - Beschriftung im Pfeil

Beispiel:

```
$$ \xymatrix{A \ar[r]^{\text{oben}} & B \ar[r]_{\text{unten}} & C \ar[r]|{\text{in}} & D} $$
```

$$A \xrightarrow{\text{oben}} B \xrightarrow{\text{unten}} C \xrightarrow{\text{in}} D$$

11.3.1 Erweiterungen der Pfeilbeschriftungen

Beschriftungen können auch am Fuß oder Kopf des Pfeiles mit `\ar[dir]^{<label>}` und `\ar[dir]_{<label>}` vorgenommen werden. Die Zahl der `<`, `>` verschiebt den Text zum Zentrum.

Beispiel:

```
$$ \xymatrix{A \ar[r]^{>>>{\text{Kopf}}} & B \ar[r]^{>>>{\text{Kopf}}} & C \ar[r]_{<{\text{Fuß}}} & D \ar[r]_{<<<{\text{Fuß}}} & E}
```

$$A \xrightarrow{\text{Kopf}} B \xrightarrow{\text{Kopf}} C \xrightarrow{\text{Fuß}} D \xrightarrow{\text{Fuß}} E$$

11.4 Pfeile senkrecht zur Pfeilrichtung verschieben

Will man mehrere Pfeile übereinander zeichnen, so kann man sie um *wert* durch `@<wert>` senkrecht zur Pfeilrichtung verschieben. *wert* muss dabei eine Größe mit Einheit sein, z. B. 0.1cm, .5ex.

Beispiel:

$$A \xleftrightarrow[f]{g} B \quad \xymatrix{A \ar[r]^f & B \ar[r]^g}$$

$$A \xleftrightarrow[f]{g} B \quad \xymatrix{A \ar@<2pt>[r]^f & B \ar@<2pt>[r]^g}$$

11.5 Beschriftung in Pfeilrichtung verschieben

Die Beschriftung kann man in Pfeilrichtung verschieben, indem man direkt vor dem Beschriftungstext `/wert/` einfügt: `\ar[l]^{0.3em/f}`.

$$A \xrightarrow{f} B \quad \text{\code{\xymatrix{A \ar[l]^f & B}}$$

$$A \xrightarrow{f} B \quad \text{\code{\xymatrix{A \ar[l]^{0.3em/f} & B}}$$

11.5.1 Gekrümmte Pfeile

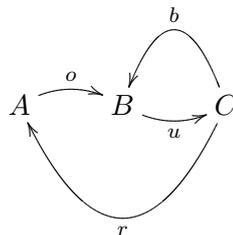
Mit `@/^/` bzw. `@/_/` werden Pfeile nach links (konvex) bzw. nach rechts (konkav) gekrümmt (in Pfeilrichtung betrachtet). Optional kann man noch angeben, wie stark der Pfeil gekrümmt wird, z. B. `@/_1cm/`.

Beispiel:

```

$$
\xymatrix{
A \ar@/^/[r]^o & & \\
B \ar@/_/[r]_u & & \\
C \ar@/^1.5cm/[ll]^r \ar@/_1cm/[l]_b & & \\
}
$$

```



11.5.2 Erscheinungsbild des Pfeiles

Das Aussehen des Pfeiles kann mittels `@{Fuß Schaft Spitze}` bestimmt werden. Zwischen den Zeichen für *Fuß Schaft Spitze* dürfen keine Leerzeichen stehen. Folgende Zeichen werden dabei erkannt:

Schäfte: -, ., ~, =, :, -, - und das Leerzeichen
Füße und Spitzen: >, <, (,), |, /, », «, ||, //, >|, |- , +, x, o

Beispiele:

$A \xleftarrow{e} B$	<code>\xymatrix{A \ar@{<->}[r]^e & B}</code>
$A \xleftarrow{p} B$	<code>\xymatrix{A \ar@_{<->}[r]^p & B}</code>
$A \xleftarrow{\cdot p} B$	<code>\xymatrix{A \ar@{<.>}[r]^p & B}</code>
$A \xleftarrow{\sim g} B$	<code>\xymatrix{A \ar@{x~>}[r]^g & B}</code>
$A \xrightarrow{f} B$	<code>\xymatrix{A \ar@{=>}[r]^f & B}</code>
$A \xrightarrow{:f} B$	<code>\xymatrix{A \ar@{:>}[r]^f & B}</code>
$A \xrightarrow{-f} B$	<code>\xymatrix{A \ar@{->}[r]^f & B}</code>
$A \xrightarrow{f} B$	<code>\xymatrix{A \ar@2{->}[r]^f & B}</code>
$A \xleftrightarrow{f} B$	<code>\xymatrix{A \ar@3{<->}[r]^f & B}</code>

11.6 Rahmenformen der Zellen

Die Zellen der Matrix können umrahmt werden. Folgende Rahmen (Frames) stehen zur Verfügung.

[F];	<code>single</code>
[F.]	<code>dotted</code>
[F=]	<code>double</code>
[F--]	<code>dashed</code>
[F-,]	<code>shades</code>
[F-:<5pt>]	<code>single,rounded</code>

Der Abstand des Textes zum Rahmen kann mit dem Parameter + optimiert werden.

+ [F]	<i>single</i>
+ [F.]	<i>dotted</i>
+ [F=]	<i>double</i>
+ [F--]	<i>dashed</i>
+ [F-,]	<i>shades</i>
+ [F- :<5pt>]	<i>single, rounded</i>

Der Rahmen kann gezielt vergrößert bzw. verkleinert werden. Dazu dienen die Parameter

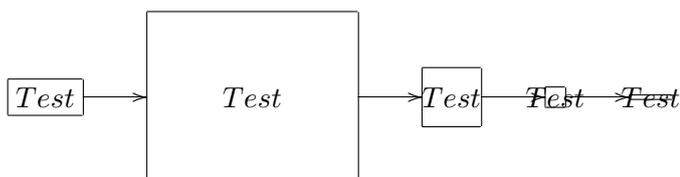
- + Vergrößern des Rahmens.
- +<2cm> Rahmen um 2cm vergrößern.
- + = Rahmen als größtes Quadrat (|x|).
- = Rahmen als kleinstes Quadrat (|x|).
- Verkleinern des Rahmen.

Beispiele:

```

$$
\xymatrix{
*+[F]{Test} \ar[r] & *+<2cm>[F]{Test} \ar[r] & *+[F]{Test} \ar[r] \\
& *-[F]{Test} \ar[r] & *-[F]{Test}
}
$$

```



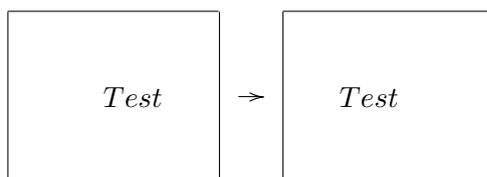
Die Objekte werden in den Zellen zentriert, so dass die Pfeile immer zur Objektmitte weisen. Eine andere Ausrichtung der *xymatrix*-Elemente ist mit [u], [d], [l], [r] möglich.

Beispiel:

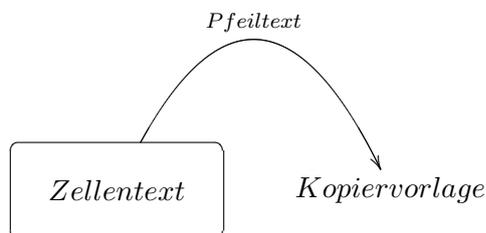
```

$$ \xymatrix{ *+<2cm>[F][r]{Test} \ar[r] & *+<2cm>[F][l]{Test} } $$

```



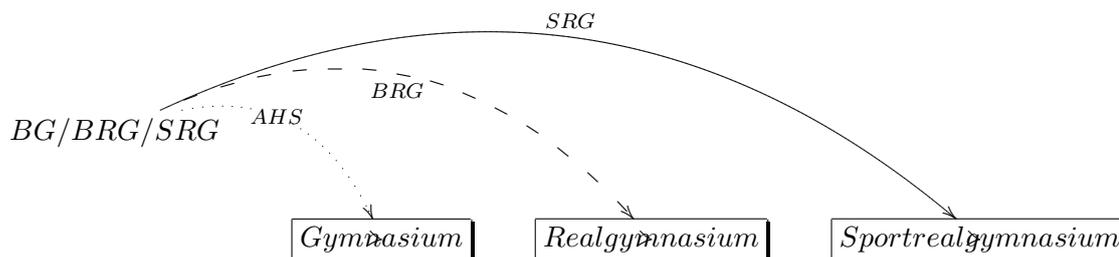
11.7 Kopierbeispiele



```

$$
\matrix{
*+<1cm>[F-:<3pt>]{Zellentext} \ar@/^2cm/{->}[r]^{\text{Pfeiltext}}
& \text{Kopiervorlage} \\
}
$$

```



```

$$
\matrix {
BG/BRG/SRG \ar@/^1cm/{.>}[dr]|{AHS} \ar@/^1.5cm/{-->}[drr]_{BRG}
\ar@/^2cm/[drrr]^{\text{SRG}} \\
\text{:} & *+[F-,]{Gymnasium} \ar@{=>} & *+[F-,]{Realgymnasium} \ar@{=>} \\
& *+[F-,]{Sportrealgymnasium} \ar@{=>} \\
}
$$

```