

# Einführung in das Textsatzsystem $\text{\LaTeX}$

## erster Tag

Moritz Brinkmann  
mail@latexkurs.de

22. Februar 2020

Nach den zwei Workshop-Tagen können Sie ...

- einfache Dokumente in  $\text{\LaTeX}$  setzen
- Hilfestellungen in Klassen- und Paketdokumentationen auffinden
- mehrsprachige Dokumente erstellen
- Abbildungen einbinden und Tabellen anlegen
- Referenzapparate erzeugen
- mathematische Formeln setzen
- größere Projekte strukturieren

## Termine

- zwei Sitzungen:
  - Samstag 22. Februar (heute)
  - Sonntag 23. Februar (morgen)

jeweils 10 bis 15 Uhr

- ca. 45 Minuten Pause

## Termine

- zwei Sitzungen:
  - Samstag 22. Februar (heute)
  - Sonntag 23. Februar (morgen)

jeweils 10 bis 15 Uhr

- ca. 45 Minuten Pause

## Materialien

- Alle Materialien stehen auf der Workshophomepage zum Download:

<https://ma.latexkurs.de/>



## Übungen

- Theorie und Praxisphasen wechseln sich ab.
- Sie dürfen (und sollen) Beispiele *jederzeit* selbst an Ihrem Computer nachmachen.
- Probieren Sie Neues ruhig *sofort* aus ...!
- Fragen Sie gerne nach, wenn Sie etwas nicht hinkriegen.

## $\text{\LaTeX}$ flavour

Die Inhalte dieses Kurses beziehen sich auf die (relativ moderne)  $\text{\LaTeX}$ -Variante  $\text{\XeLaTeX}$ .

- ① Worum geht es überhaupt?
- ② Grundlegende Bedienung
  - Klassen und Pakete
  - Grundbefehle
- ③ Typografische Grundlagen
  - Makrotypografie
  - Mikrotypografie
- ④ Dokumentation & Fehlermeldungen
  - Dokumentation
  - Fehlermeldungen
- ⑤ Sprachen
- ⑥ Gleitobjekte
  - allgemeine Gleitumgebungen
  - fake-Gleitobjekte
  - Grafiken

- 7 Bibliografien
  - biblatex
  - Verwaltung von Referenzen
- 8 Mathematiksatz
- 9 Tabellen
  - schöne Tabellen
  - automatische Spaltenbreite
- 10 Umfangreichere Projekte
- 11 Diagramme

Teil I

# The Name of the Game



# The Name of the Game

- Programm  $\text{\TeX}$  (Seit 1977)  
Geschrieben von Donald E. Knuth für sein Buch  
„The Art of Computer Programming“.  
„ $\text{\TeX}$ “ von griechisch τέχνη

# The Name of the Game

- Programm  $\text{\textcolor{blue}{T}}\text{\textcolor{blue}{E}}\text{X}$  (Seit 1977)
- Makropaket  $\text{\textcolor{blue}{plain}}\text{\textcolor{blue}{T}}\text{\textcolor{blue}{E}}\text{X}$   
Macht  $\text{\textcolor{blue}{T}}\text{\textcolor{blue}{E}}\text{X}$  für normale Nutzer bedienbar.

# The Name of the Game

- Programm  $\text{T}_{\text{E}}\text{X}$  (Seit 1977)
- Makropaket  $\text{plainT}_{\text{E}}\text{X}$
- großes Makropaket  $\text{LaT}_{\text{E}}\text{X}$  (Anfänge 1980er)  
Von Leslie Lamport: „Lamport’s  $\text{T}_{\text{E}}\text{X}$ “.  
Viele Vereinfachungen für den normalen Anwender.

# The Name of the Game

- Programm  $\text{T}_{\text{E}}\text{X}$  (Seit 1977)
- Makropaket  $\text{plainT}_{\text{E}}\text{X}$
- großes Makropaket  $\text{LaT}_{\text{E}}\text{X}$  (Anfänge 1980er)
- aktuelle, stabile Version:  $\text{LaT}_{\text{E}}\text{X } 2_{\epsilon}$  (1994)  
„in einer  $\epsilon$ -Umgebung von 2“

# The Name of the Game

- Programm  $\text{T}_{\text{E}}\text{X}$  (Seit 1977)
- Makropaket  $\text{plainT}_{\text{E}}\text{X}$
- großes Makropaket  $\text{LaT}_{\text{E}}\text{X}$  (Anfänge 1980er)
- aktuelle, stabile Version:  $\text{LaT}_{\text{E}}\text{X } 2_{\epsilon}$  (1994)
- zukünftige Entwicklung:  $\text{\LaTeX}3$   
noch nicht eigenständig verfügbar, aber als Paket `expl3` in  $\text{\LaTeX } 2_{\epsilon}$

# Was ist T<sub>E</sub>X – und was nicht?

## Dafür ist L<sup>A</sup>T<sub>E</sub>X gut geeignet ...

- Alle Schriftstücke mit logischem Aufbau
  - Naturwissenschaftliche Arbeiten (hervorragender Mathesatz)
  - Geisteswissenschaftliche Arbeiten (hervorragende Mehrsprachigkeit, Bibliographieerstellung, Erstellung von Apparaten etc.)
  - Artikel, Bachelorarbeiten, Dissertationen, ...
  - Buchreihen, Briefe
  - Präsentationen
- Viel „Missbrauch“ durch kreative Paketautoren

# Was ist T<sub>E</sub>X – und was nicht?

## Dafür ist L<sup>A</sup>T<sub>E</sub>X weniger gut geeignet ...

- Dokumente ohne logische Struktur
  - Präsentationen (bunt, drehend, blinkend, „durcheinander“)
  - Werbezettel
  - Plakate
- Dokumente mit vielen uneinheitlichen Bildern, die frei bewegt werden

# Wie funktioniert T<sub>E</sub>X?

- WYSIWYM
- reine Textdateien
- keine versteckten Einstellungen
- Textauszeichnung durch besondere Befehle:
  - „Ich will einen Artikel schreiben!“
  - „Setze eine Überschrift!“
  - „Schreibe das folgende fett!“
  - „Setze eine Tabelle, die ...“



# Wie funktioniert T<sub>E</sub>X?

## Vorteile

- Stabilität und Portabilität
- geringe Dateigrößen
- Bearbeitung mit beliebigem Editor
- Textdateien *immer* lesbar
- Ausgabe überall gleich

## Nachteile

- Ergebnis nicht direkt sichtbar
- unintuitive Bedienung
- steile Lernkurve
- Bei Änderungen muss alles neu kompiliert werden
- komplizierte Layout-wünsche schwer realisierbar

Wie lässt sich Text von Befehlen unterscheiden?

Ansatz in *klassischen* Programmiersprachen:

```
print ( " Hallo Welt! " );
```

⇒ für ein Textsatzprogramm ungeeignet

# Ein einfaches T<sub>E</sub>X-Dokument

- T<sub>E</sub>X ist eine Auszeichnungssprache (*markup language*)
- einzelne Zeichen haben besondere Bedeutung
- Backslash (\) dient als *escape character* und markiert den Anfang eines Befehls: \chapter \section \author

Einfachstes T<sub>E</sub>X-Dokument:

```
Hallo Welt! \bye
```

# Ein einfaches T<sub>E</sub>X-Dokument

- T<sub>E</sub>X ist eine Auszeichnungssprache (*markup language*)
- einzelne Zeichen haben besondere Bedeutung
- Backslash (\) dient als *escape character* und markiert den Anfang eines Befehls: \chapter \section \author

Einfachstes T<sub>E</sub>X-Dokument:

```
Hallo Welt! \bye
```

```
$ tex dokument.tex
```

erzeugt ein .dvi-Dokument und eine .log-Datei

# Ein einfaches $\text{\LaTeX}$ -Dokument

```
\documentclass{minimal}  
\begin{document}  
Hallo Welt!  
\end{document}
```

Hallo Welt!

# Ein einfaches $\text{\LaTeX}$ -Dokument

```
\documentclass{minimal}  
\begin{document}  
Hallo Welt!  
\end{document}
```

Hallo Welt!

## Arbeitsauftrag

Erstellen Sie ein erstes  $\text{\LaTeX}$ -Dokument, indem Sie dieses Minimalbeispiel in Ihrem Editor abtippen!

<code>\</code>	<i>escape character</i> , Leitet Befehle ein
<code>{}</code>	<i>grouping character</i> , gruppieren zusammengehörende Zeichen z. B. Argumente <code>\textbf{fett}</code>
<code>\$</code>	<i>math character</i> , startet und beendet Mathemodus
<code>&amp;</code>	<i>tabbing character</i> , trennt Spalten in Tabellen
<code>%</code>	<i>comment character</i> Kommentiert den Rest der Zeile aus
<code>^_~#</code>	weitere Zeichen mit besonderer Bedeutung

Teil II

# Grundlegende Bedienung



Dokumentenklassen legen grundlegende Eigenschaften des Dokuments fest:

- Layout
- Standardschriften
- Satzspiegel
- Gliederungsbefehle
- Aussehen von Verzeichnissen, Tabellen, Aufzählungen, ...

Eigenschaften sind durch Änderung von Optionen oder Laden von Paketen anpassbar.

## Standardklassen

article	(Kurze) Artikel
report	Reporte, Tagungsberichte
book	Bücher
letter	Briefe
minimal	für Minimalbeispiele

## KOMA-Script

scrartcl	Erweiterung von article
scrreprt	Erweiterung von report
scrbook	Erweiterung von book
scrlltr2	sehr mächtige Briefklasse

## Spezialklassen

beamer	für Präsentationen
tikzposter	wissenschaftliche Poster

- Pakete bieten zusätzliche Funktionalität
- Arbeitserleichterungen
- Fehlerkorrekturen
- Einbinden in der Präambel mittels

`\usepackage[<option(en)>]{<paketname>}`:

```
\documentclass{article}
\usepackage{
  amsmath,
  hyperref,
}
\usepackage[left=2cm]{geometry}
```

# Gliederungsbefehle

- Gliederungen strukturieren Dokumente,
- ermöglichen automatische Nummerierung, Eintragung in Verzeichnisse, Kolumnentitel etc.
- Werden von der Dokumentenklasse festgelegt
- Grundstruktur im Kernel definiert

⇒ bestimmte Elemente immer verfügbar

```
\part{Band I}  
\chapter{Kapitel}  
\section{Abschnitt}  
\subsection{Unterabschnitt}  
\subsubsection{Unterunterabschnitt}  
\paragraph{Paragraph}  
\subparagraph{Unterparagraph}
```

# Grundbefehle

allgemein

<code>\textrm{Serifen}</code>	Serifen <i>Abcdxyz</i>
<code>\textit{kursiv}</code>	<i>kursiv</i> <i>Abcdxyz</i>
<code>\textsl{geneigt}</code>	<i>geneigt</i> <i>Abcdxyz</i>
<code>\textsf{serifenlos}</code>	serifenlos <i>Abcdxyz</i>
<code>\textbf{fett}</code>	<b>fett</b> <b>Abcdxyz</b>
<code>\texttt{Schreibmaschine}</code>	Schreibmaschine <i>Abcdxyz</i>
<code>\textsc{Kapitälchen}</code>	KAPITÄLCHEN <i>ABCDXYZ</i>
<code>\emph{Hervorhebung}</code>	<i>Hervorhebung</i> <i>Abcdxyz</i>
<code>\\</code>	Zeilenende
<code>\par</code> oder Leerzeile	Absatzende
<code>\$E = \frac{p^2}{2m}\$</code>	Inline-Mathemodus: $E = \frac{p^2}{2m}$
<code>\[E = \frac{p^2}{2m}\]</code>	Display-Mathemodus: $E = \frac{p^2}{2m}$
<code>\tableofcontents</code>	Produziert Inhaltsverzeichnis
<code>\today</code>	aktuelles Datum

# Grundbefehle

## Schriftgrößen

<code>\tiny</code>	winzig
<code>\small</code>	klein
<code>\normalsize</code>	normal
<code>\large</code>	groß
<code>\Large</code>	größer
<code>\LARGE</code>	noch größer
<code>\huge</code>	riesig
<code>\Huge</code>	noch riesiger

## **Eingabe**

.tex      T<sub>E</sub>X-Datei mit Dokumententext

## **Ausgabe**

.pdf      pdfT<sub>E</sub>X-Ausgabe oder Umwandlung von (x)dvi

## **Eingabe**

.tex      T<sub>E</sub>X-Datei mit Dokumententext

## **Ausgabe**

.pdf      pdfT<sub>E</sub>X-Ausgabe oder Umwandlung von (x)dvi

## **Hilfsdateien (nur schreiben)**

.log      Log-Datei mit Informationen, Warnungen, Fehlern



## **Eingabe**

.tex      T<sub>E</sub>X-Datei mit Dokumententext

## **Ausgabe**

.pdf      pdfT<sub>E</sub>X-Ausgabe oder Umwandlung von (x)dvi

## **Hilfsdateien (nur schreiben)**

.log      Log-Datei mit Informationen, Warnungen, Fehlern

## **Hilfsdateien (schreiben und lesen)**

.aux      Hilfsdatei mit temporären Informationen

.toc      table of contents

.lof      list of figures

.synctex.gz      nötig für die SyncT<sub>E</sub>X-Funktion

⋮      ⋮

Teil III

# Typografische Grundlagen

- Satzspiegel
- Kopf und Fußzeilen
- Wahl der Schriften
- Formatierung von Abständen
- Aussehen von Verzeichnissen, Fußnoten, ...

- Satzspiegel
- Kopf und Fußzeilen
- Wahl der Schriften
- Formatierung von Abständen
- Aussehen von Verzeichnissen, Fußnoten, ...

## Arbeitsauftrag

Laden Sie sich die Datei `uebung_layout.tex` von der Workshopwebseite herunter. Vollziehen Sie daran nach und nach alle typografischen Einstellungen, die besprochen werden. Im Idealfall wählen Sie alle Werte so, dass Sie den Anforderungen an die Ihnen bevorstehende Abschlussarbeit genügen.

## “ Style

Format one-sided DIN A4

Font Size 12 pt

Line Spread 1.5 pt

Alignment justified (“Blocksatz”)

Left and right margin 3 cm

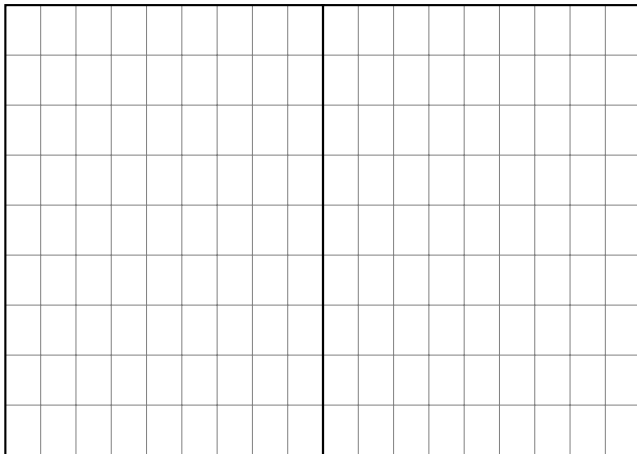
*Guidelines for Bachelor theses*

”

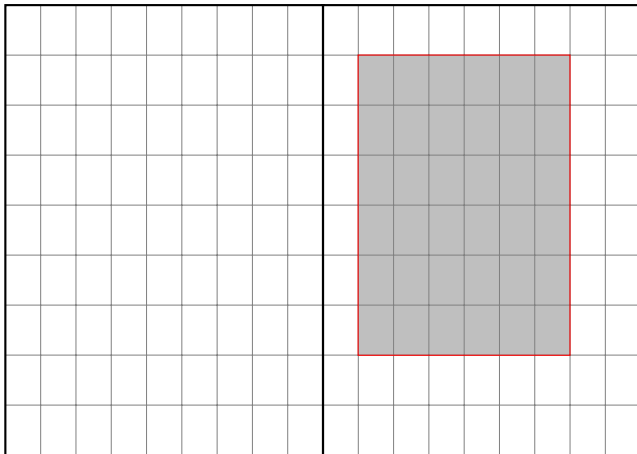
Mit Satzspiegel bezeichnet man die vom Text bedeckte Fläche (im Gegensatz zu den Rändern)

- Ein- oder zweiseitiger Satz?
- Schriftgröße, Laufweite,
- Kopf- und Fußzeilen
- Textspalten

# Moderne Satzspiegelkonstruktion

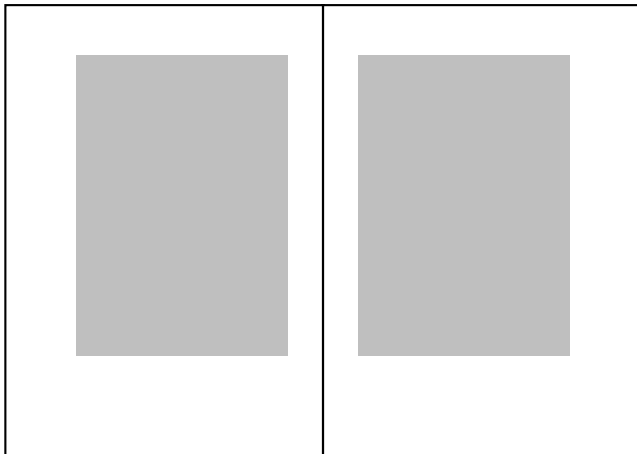


# Moderne Satzspiegelkonstruktion

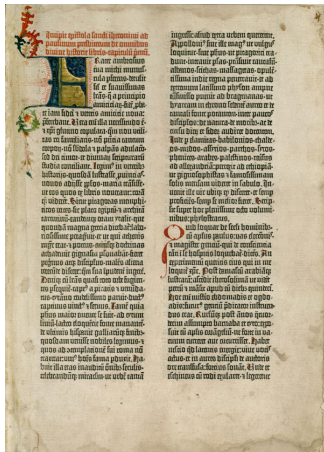




# Moderne Satzspiegelkonstruktion



# Satzspiegel bei Gutenberg



# Satzspiegel mit KOMA-Skript

- KOMA-Skript bietet optimale Satzspiegelkonstruktion mittels eigenem Paket typearea
- Anpassung eigentlich nur bei besonders breiten oder engen Schriften nötig: Option DIV=*⟨Faktor⟩*  
Autom. Berechnung anhand der Seitengröße: DIV=calc  
Berechnung nach mittelalterl. Buchseitenkanon: DIV=classic
- Bindekorrektur mittels Option BCOR=*⟨Länge⟩*

```
\documentclass[DIV=9, BCOR=12mm]{scrbook}^^I
```

Bei Nicht-KOMA-Klassen muss typearea direkt geladen werden:

```
\usepackage[DIV=13, BCOR=2cm]{typearea}
```

Paket geometry erlaubt manuelle Einstellung des Satzspiegels:

```
\usepackage[top=2cm, bottom=5cm]{geometry}
```

oder:

```
\usepackage{geometry}  
\geometry{top=2cm, bottom=5cm}
```

## mögliche Optionen

paper

left, right, inner, outer, hmargin

top, bottom, vmargin

margin

bindingoffset, textwidth, textheight

twocolumn, columnsep, marginparsep, footnotesep

headsep, footsep, nofoot, nohead

hoffset, voffset, offset

includehead, includefoot

Paket `setspace` erlaubt Anpassung der Zeilenabstände:

```
\usepackage{setspace}  
\singlespacing  
\onehalfspacing  
\doublespacing
```

Abstand in Fußnoten, etc. bleibt dabei gleich.

Finetuning: `\setstretch{⟨Faktor⟩}`

- Kopf- und Fußzeilen enthalten wichtige Informationen über das Dokument
  - lebende Kolumnentitel
  - Seitenzahlen
- Anpassung mittels verschiedener Pakete
- Auswahl über `\pagestyle{⟨Seitenstil⟩}` oder `\thispagestyle{⟨Seitenstil⟩}`
- Voreinstellungen: `empty`, `plain`, `headings`

# Kopf- und Fußzeilen mit scrlayer-scrpage

Paket definiert zwei Seitenstile: scrheadings und screadings.plain  
Anpassung mittels z. B.

```
\lehead[\langle Inhalt plain.scrheadings \rangle]{\langle Inhalt scrheadings \rangle}
```

<pre>\lehead \cehead \rehead</pre>	<pre>\lohead \cohead \rohead</pre>
<pre>\lefoot \cefoot \refoot</pre>	<pre>\lofoot \cofoot \rofoot</pre>

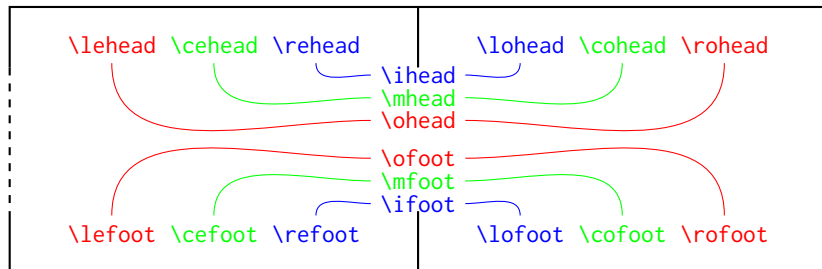
```
\documentclass{scrartcl}  
\usepackage{scrlayer-scrpage}  
\lohead*{Peter Musterheinzl}  
\rohead*{Seitenstile mit KOMA-Script}  
\pagestyle{scrheadings}
```



# Kopf- und Fußzeilen mit scrlayer-scrpage

Paket definiert zwei Seitenstile: scrheadings und screadings.plain  
Anpassung mittels z. B.

`\lehead[<Inhalt plain.scrheadings>]{<Inhalt scrheadings>}`



```
\documentclass{scrartcl}
\usepackage{scrlayer-scrpage}
\lohead*{Peter Musterheinzl}
\rohead*{Seitenstile mit KOMA-Script}
\pagestyle{scrheadings}
```

- viele Schriften sind als Paket verfügbar und können mit `\usepackage{Paketname}` geladen werden

```
\usepackage{nimbusserif}
```

- in T<sub>E</sub>Xlive verfügbare Schriften sind im „L<sup>A</sup>T<sub>E</sub>X Font Catalogue“ zu finden  
<http://www.tug.dk/FontCatalogue/>



- Paket `fontspec` erlaubt es auf Systemschriften (OTF, AAT, TTF) zuzugreifen.
- Fonts werden über spezielle Befehle geladen  
`\setmainfont[Optionen]{Name der Schrift}`

```
\usepackage{fontspec}
\setromanfont{Linux Libertine O}
\setsansfont{Linux Biolinum O}
\setmonofont[Scale=.95]{DejaVu Sans Mono}
```

- Laden bestimmter Schriften oder Features im Dokument mit  
`\fontspec{Name der Schrift}[Features]`

Die Größe der Browserschrift kann durch Klassenoption geändert werden:

```
\documentclass[12pt]{scrartcl}
```

Größe von `\large`, `\small`, etc. passt sich automatisch an.  
Standardklassen unterstützen 10pt, 11pt und 12pt.

Die Größe der Browserschrift kann durch Klassenoption geändert werden:

```
\documentclass[12pt]{scrartcl}
```

Größe von `\large`, `\small`, etc. passt sich automatisch an.  
Standardklassen unterstützen 10pt, 11pt und 12pt.

Wer *genau weiß*, was er will:

```
\fontsize{<Größe>}{<Durchschuss>}\selectfont
```

```
\fontsize{10}{12}\selectfont
```

## Arbeitsauftrag

Passen Sie Ihr Dokument den Vorgaben für Bachelorarbeiten an!

**Format** one-sided DIN A4

**Font Size** 12 pt

**Line Spread** 1.5 pt

**Alignment** justified (“Blocksatz”)

**Left and right margin** 3 cm

- $\text{\LaTeX}$ -Dokumente werden oft von Umgebungen strukturiert:

```
\begin{⟨Umgebung⟩}[⟨ggf. opt. Argumente⟩]{⟨ggf. Argumente⟩}
```

...

```
\end{⟨Umgebung⟩}
```

- Am Anfang und Ende werden Befehle ausgeführt um bestimmtes Verhalten innerhalb der Umgebung zu erreichen.
- Jede Umgebung ist eine Gruppierung (wie `{}`)  
⇒ Alle Einstellungen innerhalb einer Umgebung sind lokal.

## wichtige Umgebungen

Aufzählung	itemize
Nummerierung	enumerate
Beschreibungsliste	description
zeichengenaue Wiedergabe	verbatim
zweispaltiger Satz	twocolumn
Zitat	quotation
kurzes Zitat	quote
zentriert	center
Tabelle	tabular, tabularx, tabulary, supertabular etc.
Abbildung	figure
Gleitumgebung	table
Gleichung	align (Mathe)
Matrix	matrix (Mathe)



```
\begin{itemize}  
  \item Erster Punkt  
  \item Zweiter Punkt  
  \item[3] Dritter Punkt  
\end{itemize}
```

- Erster Punkt
- Zweiter Punkt
- 3 Dritter Punkt

```
\begin{enumerate}  
  \item Erster Punkt  
  \item Zweiter Punkt  
  \item[3] Dritter Punkt  
\end{enumerate}
```

- 1 Erster Punkt
- 2 Zweiter Punkt
- 3 Dritter Punkt

Aussehen von `itemize` und `enumerate` wird von Dokumentenklasse bestimmt.

## Arbeitsauftrag

Ergänzen Sie Ihr Dokument durch ein oder mehrere Zitate. Beobachten Sie dabei den Unterschied zwischen `quote` und `quotation`.

Testen Sie auch das Aussehen anderer Umgebungen wie `itemize` und `description`.

# Mikrotypografie

Mikrotypografie bezeichnet die Gestaltung von Feinheiten auf Buchstabenebene:

**tracking** Anpassung des  
Glyphenabstands innerhalb  
der Wörter ( $\leq 3\%$ )

VA Fo  
VA Fo

# Mikrotypografie

Mikrotypografie bezeichnet die Gestaltung von Feinheiten auf Buchstabenebene:

**tracking** Anpassung des  
Glyphenabstands innerhalb  
der Wörter ( $\leq 3\%$ )

**expansion** Anpassung der Glyphenbreite  
( $\leq 2\%$ )

Text  
Text

Mikrotypografie bezeichnet die Gestaltung von Feinheiten auf Buchstabenebene:

**tracking** Anpassung des  
Glyphenabstands innerhalb  
der Wörter ( $\leq 3\%$ )

**expansion** Anpassung der Glyphenbreite  
( $\leq 2\%$ )

**protrusion** Optischer Randausgleich

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodo consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Mikrotypografie

Mikrotypografie bezeichnet die Gestaltung von Feinheiten auf Buchstabenebene:

<b>tracking</b>	Anpassung des Glyphenabstands innerhalb der Wörter ( $\leq 3\%$ )	fi fi fl fl
<b>expansion</b>	Anpassung der Glyphenbreite ( $\leq 2\%$ )	ff ff ffl ffl
<b>protrusion</b>	Optischer Randausgleich	Qu Qu
<b>ligatures</b>	Verbindung mehrerer Buchstaben zu einer Glyphe	

Das Paket `microtype` kümmert sich um diese typografischen Feinheiten.

In der Regel reicht die Voreinstellung:

```
\usepackage{microtype}
```

- Aktiviert automatisch `protrusion` (in `pdfTeX`, `XYTeX` und `LuaTeX`) und `expansion` (in `pdfTeX` und `LuaTeX`)
- Für weitere Optionen: Dokumentation

Das Paket `microtype` kümmert sich um diese typografischen Feinheiten.

In der Regel reicht die Voreinstellung:

```
\usepackage{microtype}
```

- Aktiviert automatisch `protrusion` (in `pdfTeX`, `XYTeX` und `LuaTeX`) und `expansion` (in `pdfTeX` und `LuaTeX`)
- Für weitere Optionen: Dokumentation

## Arbeitsauftrag

Aktivieren Sie in Ihrem Dokument den optischen Randausgleich.



# Leerräume und Striche

Gute Typografie unterscheidet zwischen verschieden breiten Leerzeichen und horizontalen Strichen

- normales Leerzeichen
- schmales Leerzeichen (Spatium): \,                      z. B.    z. B.    z.B.
- kleiner Abstand (Halbgeviert): \enskip                      a   b
- weißes Quadrat (Geviert): \quad                      a    b
- negativer Abstand: \!                      ab

# Leerräume und Striche

Gute Typografie unterscheidet zwischen verschieden breiten Leerzeichen und horizontalen Strichen

- normales Leerzeichen
- schmales Leerzeichen (Spatium): `\,` z. B. z. B. z.B.
- kleiner Abstand (Halbgeviert): `\enskip` a b
- weißes Quadrat (Geviert): `\quad` a b
- negativer Abstand: `\!` ab
- explizites Ändern des Abstands (Kerning): `a\kern-.1em b` ab



## Teil IV

# Dokumentation & Fehlermeldungen

- $\text{\LaTeX}$  ist hervorragend dokumentiert
- Jede Klasse und jedes Paket bringt normalerweise eine eigene Anleitung mit.
- Dokumentation kann mittels des `texdoc`-Befehls aufgerufen werden

Auf der Kommandozeile:

- `$ texdoc` durchsucht die  $\text{\LaTeX}$ -Ordner nach Dokumentationen
- `$ texdoc amsmath` öffnet `amsmath.pdf`
- `$ texdoc -l amsmath` listet alle Ergebnisse auf
- `$ texdoc -s amsmath` liefert Ergebnisse aus erweiterter Suche
- `$ texdoc --help` zeigt eine Hilfe an

Graphische Oberfläche: `texdoctk` / `texdoc-gui`

Webservice: <http://texdoc.net>

Auf der Kommandozeile:

- `$ texdoc` durchsucht die  $\text{\LaTeX}$ -Ordner nach Dokumentationen
- `$ texdoc amsmath` öffnet `amsmath.pdf`
- `$ texdoc -l amsmath` listet alle Ergebnisse auf
- `$ texdoc -s amsmath` liefert Ergebnisse aus erweiterter Suche
- `$ texdoc --help` zeigt eine Hilfe an

Graphische Oberfläche: `texdoctk` / `texdoc-gui`

Webservice: <http://texdoc.net>

## Arbeitsauftrag

Öffnen Sie über den `texdoc`-Mechanismus die deutschsprachige Dokumentation der KOMA-Skript-Klassen.

## Was tun, wenn $\text{\LaTeX}$ anhält?

- Ruhe bewahren! (tex-Dateien können nicht beschädigt werden)
- Mit der Fehlersuche beim den letzten Änderungen anfangen.
- Ggf. Schreibfehler korrigieren.
- log-Datei Lesen!
- Viele Editoren helfen bei der Fehlersuche, indem sie zur Zeile springen, in der der Fehler aufgetreten ist.  
(Das muss nicht die fehlerhafte Zeile sein.)



## Typische Fehlermeldung:

```
! Undefined control sequence.  
1.3 Ein \Latex-Dokument  
.  
?  
! Emergency stop.  
1.3 Ein \Latex-Dokument.  
.  
No pages of output.  
Transcript written on document.log.
```

⇒ Befehl in Zeile 3 falsch geschrieben

## Typische Fehlermeldung:

```
Runaway argument?  
{itemize \item Erstes Item  
! Paragraph ended before \begin was complete.  
<to be read again>  
                                \par  
1.60  
  
?
```

⇒ Irgendwo nach itemize ein } oder ein \end{ } vergessen.

# Vollständiges Minimalbeispiel

Bei Hilfestellung in Webforen/Usenet wird in der Regel ein *vollständiges Minimalbeispiel* (MWE) verlangt.

- ① solange Code aus dem Dokument löschen, bis der Fehler gerade noch auftritt
- ② alle überflüssigen Pakete entfernen
- ③ falls Dokumentenklasse keine Rolle spielt, `minimal` verwenden
- ④ wenn Fehler nur bei viel Text auftritt, `blindtext` verwenden

Oft findet man den Fehler beim erstellen des MWE schon ganz alleine.

# Vollständiges Minimalbeispiel

Bei Hilfestellung in Webforen/Usenet wird in der Regel ein *vollständiges Minimalbeispiel* (MWE) verlangt.

- ① solange Code aus dem Dokument löschen, bis der Fehler gerade noch auftritt
- ② alle überflüssigen Pakete entfernen
- ③ falls Dokumentenklasse keine Rolle spielt, `minimal` verwenden
- ④ wenn Fehler nur bei viel Text auftritt, `blindtext` verwenden

Oft findet man den Fehler beim erstellen des MWE schon ganz alleine.

## Arbeitsauftrag

Laden Sie sich das Dokument `uebung_fehlermeldungen.tex` von der Workshophomepage, erstellen Sie ein MWE und beheben Sie falls möglich alle Fehler.

Teil V

# Sprachen

Dokument muss je nach Eingabesprache lokalisiert werden.

- Umbruchregeln
- Bezeichnungen von Verzeichnissen, Kapiteln, ...
- typografische Besonderheiten

```
\usepackage{polyglossia}  
\setmainlanguage{german}  
\setotherlanguage{english}
```

# Sprachen laden

```
\setmainlanguage[\Optionen]{\Sprache}  
\setotherlanguage[\Optionen]{\Sprache}  
\setotherlanguages{\Sprachen}
```

# Sprachen laden

```
\setmainlanguage[\langle Optionen \rangle]{\langle Sprache \rangle}  
\setotherlanguage[\langle Optionen \rangle]{\langle Sprache \rangle}  
\setotherlanguages{\langle Sprachen \rangle}
```

## Vefügbare Sprachen:

---

albanian	danish	icelandic	nko	slovenian
amharic	divehi	interlingua	norsk	spanish
arabic	dutch	irish	nynorsk	swedish
armenian	english	italian	occitan	syriac
asturian	esperanto	kannada	piedmontese	tamil
bahasai	estonian	khmer	polish	telugu
bahasam	farsi	korean	portuges	thai
basque	finnish	lao	romanian	tibetan
bengali	french	latin	romansh	turkish
brazil[ian]	friulan	latvian	russian	turkmen
breton	galician	lithuanian	samin	ukrainian
bulgarian	german	lsorbian	sanskrit	urdu
catalan	greek	magyar	scottish	usorbian
coptic	hebrew	malayalam	serbian	vietnamese
croatian	hindi	marathi	slovak	welsh
czech				

---



# Sprache umschalten

Befehl `\text{<Sprache>{<Text>}}` für einzelne Wörter

Umgebung `\begin{<Sprache>}` für längere Passagen

# Sprache umschalten

Befehl `\text{<Sprache>{<Text>}}` für einzelne Wörter

Umgebung `\begin{<Sprache>}` für längere Passagen

% in der Präambel:

```
\setmainlanguage{english}  
\setotherlanguages{french, greek}
```

% im Dokument:

The document body is in English, but single words can be in  
`\textgreek{ ελληνικά }` or `\textfrench{français}`.

```
\begin{french}
```

Il est également possible d'écrire des phrases entières  
en français.

```
\end{french}
```

Bezeichnung von Elementen im Text passen sich der Sprache an:

```
heute ist der \today \\
\textenglish{today is \today}\\
\texttrussian{ сегодня, является \today }
```

```
heute ist der 4. Januar 2020
today is January 4, 2020
сегодня, является 4 января 2020 г.
```

# Lokalisierte Objekte

Bezeichnung von Elementen im Text passen sich der Sprache an:

```
heute ist der \today \\  
\textenglish{today is \today} \\  
\textrussian{ сегодня, является \today }
```

heute ist der 4. Januar 2020  
today is January 4, 2020  
сегодня, является 4 января 2020 г.

## Arbeitsauftrag

Sorgen Sie in Ihrem Dokument für korrekten Umbruch in mindestens zwei Sprachen.

Teil VI

# Gleitobjekte

# Was sind Gleitobjekte?

- Objekte, die frei im Dokument „gleiten“ können
- Gleiten vermeidet große Leerräume
- T<sub>E</sub>X versucht optimale Positionierung
- zu beachten:
  - Objekte sollen nicht vor Referenzen auftauchen
  - Objekte sollen nicht die Reihenfolge tauschen
  - Seitenumbruch stark abhängig von Gleitobjekten
  - *optimaler Seitenumbruch ist mit T<sub>E</sub>X nicht möglich!*

Eine Gleitumgebung besteht aus verschiedenen Teilen:

- Inhalt (Bild, Tabelle, Text, ...)
- automatische Bezeichnung: „Tabelle 1:“ (`\caption`)
- Beschriftung: „Messergebnisse“ (Argument von `\caption{}`)
- Markierung für Verweise: `\label{fig:vergleichsdaten}`

Eine Gleitumgebung besteht aus verschiedenen Teilen:

- Inhalt (Bild, Tabelle, Text, ...)
- automatische Bezeichnung: „Tabelle 1:“ (`\caption`)
- Beschriftung: „Messergebnisse“ (Argument von `\caption{}`)
- Markierung für Verweise: `\label{fig:vergleichsdaten}`
- Label kann mit `\ref{fig:vergleichsdaten}` im Text referenziert werden



Eine Gleitumgebung besteht aus verschiedenen Teilen:

- Inhalt (Bild, Tabelle, Text, ...)
- automatische Bezeichnung: „Tabelle 1:“ (`\caption`)
- Beschriftung: „Messergebnisse“ (Argument von `\caption{}`)
- Markierung für Verweise: `\label{fig:vergleichsdaten}`
- Label kann mit `\ref{fig:vergleichsdaten}` im Text referenziert werden
- `\listoffigures` und `\listoftables` erstellen automatisch Abbildungs- bzw. Tabellenverzeichnis

- $\LaTeX$  verfügt über verschiedene Gleitumgebungen:
- `table` für Tabellen
- `figure` für Abbildungen
- Paket `float` ermöglicht Definition eigener Umgebungen
- für zweispaltigen Satz: `table*`, `figure*` über beide Spalten

## Positionierungsparameter für Gleitumgebungen:

```
\begin{table}[\langle Parameter \rangle]
```

- ! überschreibt interne Parameter
- h Objekt genau an dieser Stelle setzen
- t Objekt am Seitenanfang setzen
- b Objekt am Seitenende setzen
- p Objekt in Gleitobjektseite bzw. -spalte setzen
- H „genau hier und sonst nirgends“ – Paket `float`

```
\begin{table}
  \centering
  \begin{tabular}{ccc}
    a & b & c
  \end{tabular}
  \caption{Eine sinnlose
Tabelle}
  \label{tab:sinnlos}
\end{table}
```

Im Text kann man auf Tabelle  
`\ref{tab:sinnlos}` verweisen.

a	b	c
---	---	---

**Tabelle 1:** Eine sinnlose Tabelle

Im Text kann man auf Tabelle 1  
verweisen.

# Nichtgleitende Gleitumgebungen

nichtgleitende Umgebungen als Gleitumgebungen ausgeben:

Paket `caption`

Eine kleine Abbildung in einem Text, die eigentlich gar keine ist:

```
\begin{minipage}[b]{3cm}
  \fbox{ ich bin kein Bild }
  \captionof{figure}{test}
\end{minipage}
```

In der `\verb/minipage/` kann jeder beliebige Inhalt stehen `\dots`

Eine kleine Abbildung in einem Text, die eigentlich gar keine ist:

ich bin kein Bild

**Abbildung 1:** test

In der `minipage` kann jeder beliebige Inhalt stehen ...

```
\usepackage{graphicx}
```

- Grundbefehl: `\includegraphics[<optionen>]{<datei>}`
- key=value-Interface:  
[scale = 0.5, angle=50]
- Dateiendung muss nicht angegeben werden
- keine absoluten Pfadangaben verwenden (Portabilität)

# Einbinden von Grafiken

```
\includegraphics[width=2cm]{raptor.pdf}  
\includegraphics[width=.3\textwidth,angle=25]{raptor}
```



# Optionen für includegraphics

`\includegraphics` kennt viele Optionen, z. B.

<code>scale</code>	<code>0.8</code>
<code>width</code>	<code>.2\textwidth, 15pt, ...</code>
<code>height</code>	<code>2em, 40mm, ...</code>
<code>keepaspectratio</code>	<code>true</code> oder <code>false</code>
<code>angle</code>	<code>50</code>
<code>bb</code>	<code>0 0 10 20</code>
<code>clip</code>	<code>true</code> oder <code>false</code>

⇒ siehe Dokumentation zu [graphicx](#)



# Mehrere Bilder in einer Abbildung

```
\usepackage{subcaption}

\begin{figure}
  \begin{subfigure}{.5\textwidth}
    \includegraphics{bild1}
    \caption{Erstes Teilbild}
  \end{subfigure}
  \begin{subfigure}{.5\textwidth}
    \includegraphics{bild2}
    \caption{Zweites Teilbild}
  \end{subfigure}
  \caption{Bildunterschrift für beide Bilder}
\end{figure}
```

Paket `subcaption` bietet Umgebung `subfigure` innerhalb von `figure`.



Herbert Voß.

„Einführung in  $\text{\LaTeX}$ “.

Lehmanns Media, 2012.



Marco Daniel u. a.

„ $\text{\LaTeX}$  2<sub>ε</sub>-Kurzbeschreibung“.

`texdoc 12kurz`



Robert Bringhurst.

„The Elements of Typographic Style“.

Vancouver: Hartley & Marks, 1992.



Markus Kohm und Jens-Uwe Morawski.

„KOMA-Skript“.

`texdoc koma-script`

Lehmanns Media, 2012.



Arthur Reutenauer, François Charette und Elie Roux.  
„polyglossia: An Alternative to Babel for XeLaTeX and  
LuaLaTeX“.

`texdoc polyglossia`



Michael Goossens, Sebastian Rahtz und Frank Mittelbach.  
„The  $\text{\LaTeX}$  Graphics Companion“.  
Upper Saddle River: Addison-Wesley, 2008.

Happy T<sub>E</sub>Xing