

ZfS-Kurs „ $\text{\LaTeX}$ für Naturwissenschaftler“

Karin Halupczok

SoSe 2009

Erste Kurssitzung

Einführung

Aufbau eines $\text{\LaTeX}$ -Dokuments

Was ist $\text{\LaTeX}$? – Zur Geschichte von $\text{\LaTeX}$

$\text{\LaTeX}$ (gesprochen „Laa-Tech“) ist ein Computer -
Satzsatzprogramm mit sehr umfangreichen Möglichkeiten.

Was ist $\text{\LaTeX}$? – Zur Geschichte von $\text{\LaTeX}$

$\text{\LaTeX}$ (gesprochen „Laa-Tech“) ist ein Computer -
Satzprogramm mit sehr umfangreichen Möglichkeiten.



1977: Donald E. Knuth
entwickelte die Programmiersprache $\text{\TeX}$ (tau epsilon chi – sprich
„Tech“) und Metafont zum professionellen Textsatz.

Was ist $\text{\LaTeX}$? – Zur Geschichte von $\text{\LaTeX}$

$\text{\LaTeX}$ (gesprochen „Laa-Tech“) ist ein Computer -
Satzprogramm mit sehr umfangreichen Möglichkeiten.



1977: Donald E. Knuth
entwickelte die Programmiersprache $\text{\TeX}$ (tau epsilon chi – sprich
„Tech“) und Metafont zum professionellen Textsatz.

In den 1980er Jahren wurde $\text{\LaTeX}$ (sprich „Laa-Tech“) von Leslie
Lamport zunächst als ein Makropaket für $\text{\TeX}$ entwickelt, d. h. als
eine Sammlung von Befehlen und Funktionen in $\text{\TeX}$.

Was ist $\text{\LaTeX}$? – Zur Geschichte von $\text{\LaTeX}$

$\text{\LaTeX}$ (gesprochen „Laa-Tech“) ist ein Computer -
Satzprogramm mit sehr umfangreichen Möglichkeiten.



1977: Donald E. Knuth
entwickelte die Programmiersprache $\text{\TeX}$ (tau epsilon chi – sprich
„Tech“) und Metafont zum professionellen Textsatz.

In den 1980er Jahren wurde $\text{\LaTeX}$ (sprich „Laa-Tech“) von Leslie
Lamport zunächst als ein Makropaket für $\text{\TeX}$ entwickelt, d. h. als
eine Sammlung von Befehlen und Funktionen in $\text{\TeX}$.

Die aktuelle, vereinheitlichte Version ist $\text{\LaTeX}2\text{e}$, sie existiert seit
1994 und wird stets aktualisiert. Es gibt $\text{\LaTeX}$ -Versionen für alle
gängigen Betriebssysteme.

Für welche Betriebssysteme gibt es $\text{\LaTeX}$?

$\text{\LaTeX}$ ist frei nutzbar und betriebsystemunabhängig.

Für welche Betriebssysteme gibt es $\text{\LaTeX}$?

$\text{\LaTeX}$ ist frei nutzbar und betriebssystemunabhängig.

Die älteste Implementierung: Unter Unix/Linux.

Besonderheiten von Unix-Systemen: Rechnernetze für mehrere Benutzer, Dateien mit Benutzerrechten, „multitasking“, Stabilität, etc.

Für welche Betriebssysteme gibt es $\text{\LaTeX}$?

$\text{\LaTeX}$ ist frei nutzbar und betriebssystemunabhängig.

Die älteste Implementierung: Unter Unix/Linux.

Besonderheiten von Unix-Systemen: Rechnernetze für mehrere Benutzer, Dateien mit Benutzerrechten, „multitasking“, Stabilität, etc.

Für Linux bei DANTE e. V. verfügbar unter

<http://www.dante.de/software/ctan/>

Für welche Betriebssysteme gibt es $\text{\LaTeX}$?

$\text{\LaTeX}$ ist frei nutzbar und betriebssystemunabhängig.

Die älteste Implementierung: Unter Unix/Linux.

Besonderheiten von Unix-Systemen: Rechnernetze für mehrere Benutzer, Dateien mit Benutzerrechten, „multitasking“, Stabilität, etc.

Für Linux bei DANTE e. V. verfügbar unter

<http://www.dante.de/software/ctan/>

Bekannteste Implementierung für Macintosh: $\text{\OzTeX}$ unter

<http://www.trevorrow.com/oztex/>

Für welche Betriebssysteme gibt es $\text{\LaTeX}$?

$\text{\LaTeX}$ ist frei nutzbar und betriebssystemunabhängig.

Die älteste Implementierung: Unter Unix/Linux.

Besonderheiten von Unix-Systemen: Rechnernetze für mehrere Benutzer, Dateien mit Benutzerrechten, „multitasking“, Stabilität, etc.

Für Linux bei DANTE e. V. verfügbar unter

<http://www.dante.de/software/ctan/>

Bekannteste Implementierung für Macintosh: $\text{\OzTeX}$ unter

<http://www.trevorrow.com/oztex/>

Bekannteste Implementierung für MS-Windows: $\text{\MiKTeX}$ unter

<http://www.miktex.org/>

Grundlegendes Prinzip von L^AT_EX

- ▶ Texte werden strukturiert: Erst kommt der Inhalt, dann das Layout, welches zentral veränderbar ist. Damit ist es leicht, ein konsistentes Aussehen zu erzeugen.

Grundlegendes Prinzip von $\text{\LaTeX}$

- ▶ Texte werden strukturiert: Erst kommt der Inhalt, dann das Layout, welches zentral veränderbar ist. Damit ist es leicht, ein konsistentes Aussehen zu erzeugen.
- ▶ Anweisungen an den „Setzer“ (also an $\text{\LaTeX}$) des Dokuments, wie z. B. „hier aktuelles Datum einsetzen“, stehen in der $\text{\LaTeX}$ -Quelldatei.

Beispiel: `\today` setzt das aktuelle Datum in den Text ein.

Grundlegendes Prinzip von L^AT_EX

- ▶ Texte werden strukturiert: Erst kommt der Inhalt, dann das Layout, welches zentral veränderbar ist. Damit ist es leicht, ein konsistentes Aussehen zu erzeugen.
- ▶ Anweisungen an den „Setzer“ (also an L^AT_EX) des Dokuments, wie z. B. „hier aktuelles Datum einsetzen“, stehen in der L^AT_EX-Quelldatei.
Beispiel: `\today` setzt das aktuelle Datum in den Text ein.
- ▶ Man erstellt also eine Datei *name.tex* mit dem Quellcode. In dieser wird das gewünschte Ergebnis programmiert.

... Grundlegendes Prinzip von L^AT_EX

- ▶ Die Quelldatei *name.tex* wird mit dem Befehl *latex* bzw. *pdflatex* kompiliert. Dies erzeugt eine Ausgabedatei *name.dvi* bzw. *name.pdf*

... Grundlegendes Prinzip von L^AT_EX

- ▶ Die Quelldatei *name.tex* wird mit dem Befehl *latex* bzw. *pdflatex* kompiliert. Dies erzeugt eine Ausgabedatei *name.dvi* bzw. *name.pdf*
- ▶ *dvi* ist das ursprüngliche Ausgabeformat von L^AT_EX; es kann mit *dvips* in PostScript (Druckersprache; Dateiendung *ps*) verwandelt werden.

... Grundlegendes Prinzip von L^AT_EX

- ▶ Die Quelldatei *name.tex* wird mit dem Befehl *latex* bzw. *pdflatex* kompiliert. Dies erzeugt eine Ausgabedatei *name.dvi* bzw. *name.pdf*
- ▶ *dvi* ist das ursprüngliche Ausgabeformat von L^AT_EX; es kann mit *dvips* in PostScript (Druckersprache; Dateiendung *ps*) verwandelt werden.
- ▶ Daneben gibt es Hilfsdateien:
stets *name.log* (Log-Datei), *name.aux* (allgemeine Hilfsdatei);
bei Bedarf andere, z. B. *name.toc* (für Inhaltsverzeichnis)

Nachteile von $\text{\LaTeX}$

- ▶ man sieht nicht sofort das Ergebnis

Nachteile von $\text{\LaTeX}$

- ▶ man sieht nicht sofort das Ergebnis
- ▶ man muss die „Programmiersprache“ lernen
→ lange Einarbeitungszeit

Nachteile von $\text{\LaTeX}$

- ▶ man sieht nicht sofort das Ergebnis
- ▶ man muss die „Programmiersprache“ lernen
→ lange Einarbeitungszeit
- ▶ etwas umständlich für schnelle, kleine Texte

Nachteile von L^AT_EX

- ▶ man sieht nicht sofort das Ergebnis
- ▶ man muss die „Programmiersprache“ lernen
→ lange Einarbeitungszeit
- ▶ etwas umständlich für schnelle, kleine Texte
- ▶ der Quellcode ist sensibel gegenüber kleinen Fehlern

Nachteile von $\text{\LaTeX}$

- ▶ man sieht nicht sofort das Ergebnis
- ▶ man muss die „Programmiersprache“ lernen
→ lange Einarbeitungszeit
- ▶ etwas umständlich für schnelle, kleine Texte
- ▶ der Quellcode ist sensibel gegenüber kleinen Fehlern
- ▶ manchmal gibt es Unverträglichkeiten zwischen verschiedenen Erweiterungspaketen

Vorteile von L^AT_EX

- ▶ präzise Programmierung des Ergebnisses möglich

Vorteile von L^AT_EX

- ▶ präzise Programmierung des Ergebnisses möglich
- ▶ keine versteckten Steuerzeichen

Vorteile von L^AT_EX

- ▶ präzise Programmierung des Ergebnisses möglich
- ▶ keine versteckten Steuerzeichen
- ▶ „ästhetischer Textsatz“

Vorteile von L^AT_EX

- ▶ präzise Programmierung des Ergebnisses möglich
- ▶ keine versteckten Steuerzeichen
- ▶ „ästhetischer Textsatz“
- ▶ bestes Programm für mathematischen Formelsatz

Vorteile von L^AT_EX

- ▶ präzise Programmierung des Ergebnisses möglich
- ▶ keine versteckten Steuerzeichen
- ▶ „ästhetischer Textsatz“
- ▶ bestes Programm für mathematischen Formelsatz
- ▶ automatisches Inhaltsverzeichnis, bequeme Querverweise, u.ä.

Vorteile von L^AT_EX

- ▶ präzise Programmierung des Ergebnisses möglich
- ▶ keine versteckten Steuerzeichen
- ▶ „ästhetischer Textsatz“
- ▶ bestes Programm für mathematischen Formelsatz
- ▶ automatisches Inhaltsverzeichnis, bequeme Querverweise, u.ä.
- ▶ Möglichkeit zum Programmieren eigener Befehle

Vorteile von L^AT_EX

- ▶ präzise Programmierung des Ergebnisses möglich
- ▶ keine versteckten Steuerzeichen
- ▶ „ästhetischer Textsatz“
- ▶ bestes Programm für mathematischen Formelsatz
- ▶ automatisches Inhaltsverzeichnis, bequeme Querverweise, u.ä.
- ▶ Möglichkeit zum Programmieren eigener Befehle
- ▶ Quelldateien sind Textdateien, also klein, lesbar und systemunabhängig (aber Zeichenkodierung beachten!)

Vorteile von L^AT_EX

- ▶ präzise Programmierung des Ergebnisses möglich
- ▶ keine versteckten Steuerzeichen
- ▶ „ästhetischer Textsatz“
- ▶ bestes Programm für mathematischen Formelsatz
- ▶ automatisches Inhaltsverzeichnis, bequeme Querverweise, u.ä.
- ▶ Möglichkeit zum Programmieren eigener Befehle
- ▶ Quelldateien sind Textdateien, also klein, lesbar und systemunabhängig (aber Zeichenkodierung beachten!)
- ▶ Open–Source–Programm:
kostenlos und kann von jedem weiterentwickelt werden
(daher reichhaltiges Angebot an Erweiterungen)

Vorteile von L^AT_EX

- ▶ präzise Programmierung des Ergebnisses möglich
- ▶ keine versteckten Steuerzeichen
- ▶ „ästhetischer Textsatz“
- ▶ bestes Programm für mathematischen Formelsatz
- ▶ automatisches Inhaltsverzeichnis, bequeme Querverweise, u.ä.
- ▶ Möglichkeit zum Programmieren eigener Befehle
- ▶ Quelldateien sind Textdateien, also klein, lesbar und systemunabhängig (aber Zeichenkodierung beachten!)
- ▶ Open–Source–Programm:
kostenlos und kann von jedem weiterentwickelt werden
(daher reichhaltiges Angebot an Erweiterungen)
- ▶ verschiedene Versionen sind untereinander verträglich

Welche Möglichkeiten hat L^AT_EX?

Je nach Anwendung erlaubt L^AT_EX eine Fülle an Erweiterungen mittels dazuladbaren Paketen.

Diese können sein:

- ▶ Sprachen (griechisch, russisch, arabisch, japanisch, hindi, ägyptische Hieroglyphen. . .)

Welche Möglichkeiten hat $\text{\LaTeX}$?

Je nach Anwendung erlaubt $\text{\LaTeX}$ eine Fülle an Erweiterungen mittels dazuladbaren Paketen.

Diese können sein:

- ▶ Sprachen (griechisch, russisch, arabisch, japanisch, hindi, ägyptische Hieroglyphen. . .)
- ▶ Diagramme, Grafik, Schaltkreise, Feynman-Diagramme

Welche Möglichkeiten hat $\text{\LaTeX}$?

Je nach Anwendung erlaubt $\text{\LaTeX}$ eine Fülle an Erweiterungen mittels dazuladbaren Paketen.

Diese können sein:

- ▶ Sprachen (griechisch, russisch, arabisch, japanisch, hindi, ägyptische Hieroglyphen. . .)
- ▶ Diagramme, Grafik, Schaltkreise, Feynman-Diagramme
- ▶ Chemische Formeln

Welche Möglichkeiten hat $\text{\LaTeX}$?

Je nach Anwendung erlaubt $\text{\LaTeX}$ eine Fülle an Erweiterungen mittels dazuladbaren Paketen.

Diese können sein:

- ▶ Sprachen (griechisch, russisch, arabisch, japanisch, hindi, ägyptische Hieroglyphen. . .)
- ▶ Diagramme, Grafik, Schaltkreise, Feynman-Diagramme
- ▶ Chemische Formeln
- ▶ Musik: Notensetzen

Welche Möglichkeiten hat $\text{\LaTeX}$?

Je nach Anwendung erlaubt $\text{\LaTeX}$ eine Fülle an Erweiterungen mittels dazuladbaren Paketen.

Diese können sein:

- ▶ Sprachen (griechisch, russisch, arabisch, japanisch, hindi, ägyptische Hieroglyphen. . .)
- ▶ Diagramme, Grafik, Schaltkreise, Feynman-Diagramme
- ▶ Chemische Formeln
- ▶ Musik: Notensetzen
- ▶ Schachpartien

Welche Möglichkeiten hat $\text{\LaTeX}$?

Je nach Anwendung erlaubt $\text{\LaTeX}$ eine Fülle an Erweiterungen mittels dazuladbaren Paketen.

Diese können sein:

- ▶ Sprachen (griechisch, russisch, arabisch, japanisch, hindi, ägyptische Hieroglyphen. . .)
- ▶ Diagramme, Grafik, Schaltkreise, Feynman-Diagramme
- ▶ Chemische Formeln
- ▶ Musik: Notensetzen
- ▶ Schachpartien
- ▶ Mathematik

Welche Möglichkeiten hat $\text{\LaTeX}$?

Je nach Anwendung erlaubt $\text{\LaTeX}$ eine Fülle an Erweiterungen mittels dazuladbaren Paketen.

Diese können sein:

- ▶ Sprachen (griechisch, russisch, arabisch, japanisch, hindi, ägyptische Hieroglyphen. . .)
- ▶ Diagramme, Grafik, Schaltkreise, Feynman-Diagramme
- ▶ Chemische Formeln
- ▶ Musik: Notensetzen
- ▶ Schachpartien
- ▶ Mathematik
- ▶ beamer-Paket statt Powerpoint!

Welche Möglichkeiten hat $\text{\LaTeX}$?

Je nach Anwendung erlaubt $\text{\LaTeX}$ eine Fülle an Erweiterungen mittels dazuladbaren Paketen.

Diese können sein:

- ▶ Sprachen (griechisch, russisch, arabisch, japanisch, hindi, ägyptische Hieroglyphen. . .)
- ▶ Diagramme, Grafik, Schaltkreise, Feynman-Diagramme
- ▶ Chemische Formeln
- ▶ Musik: Notensetzen
- ▶ Schachpartien
- ▶ Mathematik
- ▶ beamer-Paket statt Powerpoint!
- ▶ und viel mehr. . .

Literatur und Hilfe

- ▶ „T_EX users group“: www.tug.org

Literatur und Hilfe

- ▶ „T_EX users group“: www.tug.org
- ▶ „Dante e. V.“: www.dante.de

Literatur und Hilfe

- ▶ „T_EX users group“: www.tug.org
- ▶ „Dante e. V.“: www.dante.de
- ▶ verschiedene Einführungen im Netz, z.B.
Oetiker et.al. „The not so short introduction to L^AT_EX 2_ε“
www.dante.de/CTAN/info/lshort/english/lshort.pdf

Literatur und Hilfe

- ▶ „T_EX users group“: www.tug.org
- ▶ „Dante e. V.“: www.dante.de
- ▶ verschiedene Einführungen im Netz, z.B.
Oetiker et.al. „The not so short introduction to L^AT_EX 2_ε“
www.dante.de/CTAN/info/lshort/english/lshort.pdf
- ▶ Thomas F. Sturm: „L^AT_EX – Einführung in das Textsatzsystem“,
RRNZ-Handbuch (günstiges Skript, gibt's im Uni-Shop)

Literatur und Hilfe

- ▶ „T_EX users group“: www.tug.org
- ▶ „Dante e. V.“: www.dante.de
- ▶ verschiedene Einführungen im Netz, z.B.
Oetiker et.al. „The not so short introduction to L^AT_EX 2_ε“
www.dante.de/CTAN/info/lshort/english/lshort.pdf
- ▶ Thomas F. Sturm: „L^AT_EX – Einführung in das Textsatzsystem“,
RRNZ-Handbuch (günstiges Skript, gibt's im Uni-Shop)
- ▶ Helmut Kopka „L^AT_EX“, Addison Wesley)
(früher *das* Buch für Einsteiger, jetzt in drei Bänden)

Literatur und Hilfe

- ▶ „T_EX users group“: www.tug.org
- ▶ „Dante e. V.“: www.dante.de
- ▶ verschiedene Einführungen im Netz, z.B.
Oetiker et.al. „The not so short introduction to L^AT_EX 2_ε“
www.dante.de/CTAN/info/lshort/english/lshort.pdf
- ▶ Thomas F. Sturm: „L^AT_EX – Einführung in das Textsatzsystem“, RRNZ-Handbuch (günstiges Skript, gibt's im Uni-Shop)
- ▶ Helmut Kopka „L^AT_EX“, Addison Wesley
(früher *das* Buch für Einsteiger, jetzt in drei Bänden)
- ▶ Mittelbach, Goossens „Der L^AT_EX-Begleiter“, Addison Wesley
„Das Bernhardiner-Buch“, einer der Klassiker. Beschreibt vollständig den neuen Standard L^AT_EX 2_ε und zahlreiche Pakete unter anderem aus den Bereichen Grafik, Tabellen und Mathematik. Geeignet für Einsteiger und Fortgeschrittene

...mehr Literatur und Hilfe

- ▶ Walter Schmidt u. a.: $\text{\LaTeX}$ 2e Kurzbeschreibung, sehr gute Einführung in $\text{\LaTeX}$, ist für den Anfang völlig ausreichend, unter <ftp://ftp.dante.de/tex-archive/info/lshort/german/>

...mehr Literatur und Hilfe

- ▶ Walter Schmidt u. a.: $\text{\LaTeX}$ 2e Kurzbeschreibung, sehr gute Einführung in $\text{\LaTeX}$, ist für den Anfang völlig ausreichend, unter <ftp://ftp.dante.de/tex-archive/info/lshort/german/>
- ▶ Liste der $\text{\LaTeX}$ -Symbole im Netz:
www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf

...mehr Literatur und Hilfe

- ▶ Walter Schmidt u. a.: $\text{\LaTeX}$ 2e Kurzbeschreibung, sehr gute Einführung in $\text{\LaTeX}$, ist für den Anfang völlig ausreichend, unter <ftp://ftp.dante.de/tex-archive/info/lshort/german/>
- ▶ Liste der $\text{\LaTeX}$ -Symbole im Netz:
www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf
- ▶ Manuela Jürgens: $\text{\LaTeX}$ - eine Einführung und ein bißchen mehr... Einführung in $\text{\LaTeX}$, Teil 1: <ftp://ftp.fernuni-hagen.de/pub/pdf/urz-broschueren/broschueren/a0260003.pdf>

...mehr Literatur und Hilfe

- ▶ Walter Schmidt u. a.: $\text{\LaTeX}$ 2e Kurzbeschreibung, sehr gute Einführung in $\text{\LaTeX}$, ist für den Anfang völlig ausreichend, unter <ftp://ftp.dante.de/tex-archive/info/lshort/german/>
- ▶ Liste der $\text{\LaTeX}$ -Symbole im Netz:
www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf
- ▶ Manuela Jürgens: $\text{\LaTeX}$ - eine Einführung und ein bißchen mehr... Einführung in $\text{\LaTeX}$, Teil 1: <ftp://ftp.fernuni-hagen.de/pub/pdf/urz-broschueren/broschueren/a0260003.pdf>
- ▶ Manuela Jürgens: : $\text{\LaTeX}$ - Fortgeschrittene Anwendungen Einführung in $\text{\LaTeX}$, Teil 2: <ftp://ftp.fernuni-hagen.de/pub/pdf/urz-broschueren/broschueren/a0279510.pdf>

...mehr Literatur und Hilfe

- ▶ Walter Schmidt u. a.: $\text{\LaTeX}$ 2e Kurzbeschreibung, sehr gute Einführung in $\text{\LaTeX}$, ist für den Anfang völlig ausreichend, unter <ftp://ftp.dante.de/tex-archive/info/lshort/german/>
- ▶ Liste der $\text{\LaTeX}$ -Symbole im Netz:
www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf
- ▶ Manuela Jürgens: $\text{\LaTeX}$ - eine Einführung und ein bißchen mehr... Einführung in $\text{\LaTeX}$, Teil 1: <ftp://ftp.fernuni-hagen.de/pub/pdf/urz-broschueren/broschueren/a0260003.pdf>
- ▶ Manuela Jürgens: : $\text{\LaTeX}$ - Fortgeschrittene Anwendungen Einführung in $\text{\LaTeX}$, Teil 2: <ftp://ftp.fernuni-hagen.de/pub/pdf/urz-broschueren/broschueren/a0279510.pdf>
- ▶ Beschreibung der $\text{\TeX}$ -Pakete:
<http://texcatalogue.sarovar.org/brief.html>

...mehr Literatur und Hilfe

- ▶ Walter Schmidt u. a.: $\text{\LaTeX}$ 2e Kurzbeschreibung, sehr gute Einführung in $\text{\LaTeX}$, ist für den Anfang völlig ausreichend, unter <ftp://ftp.dante.de/tex-archive/info/lshort/german/>
- ▶ Liste der $\text{\LaTeX}$ -Symbole im Netz:
www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf
- ▶ Manuela Jürgens: $\text{\LaTeX}$ - eine Einführung und ein bißchen mehr... Einführung in $\text{\LaTeX}$, Teil 1: <ftp://ftp.fernuni-hagen.de/pub/pdf/urz-broschueren/broschueren/a0260003.pdf>
- ▶ Manuela Jürgens: : $\text{\LaTeX}$ - Fortgeschrittene Anwendungen Einführung in $\text{\LaTeX}$, Teil 2: <ftp://ftp.fernuni-hagen.de/pub/pdf/urz-broschueren/broschueren/a0279510.pdf>
- ▶ Beschreibung der $\text{\TeX}$ -Pakete:
<http://texcatalogue.sarovar.org/brief.html>
- ▶ Viel Material unter:
<http://cips02.physik.uni-bonn.de/pool/soft/tex/#a5b1>

Einführung

Aufbau eines $\text{\LaTeX}$ -Dokuments

Wie sieht eine .tex-Datei aus?

Erst kommt der sogenannte Vorspann, auch Präambel genannt, in dem allgemeine Parameters des Dokuments beschrieben werden, dann in der `document`-Umgebung der eigentliche Text.

Wie sieht eine .tex-Datei aus?

Erst kommt der sogenannte Vorspann, auch Präambel genannt, in dem allgemeine Parameters des Dokuments beschrieben werden, dann in der document-Umgebung der eigentliche Text. Z. B.:

```
\documentclass[12pt]{article}  
\usepackage{german}  
\usepackage[latin1]{inputenc}  
\usepackage[T1]{fontenc}
```

Wie sieht eine .tex-Datei aus?

Erst kommt der sogenannte Vorspann, auch Präambel genannt, in dem allgemeine Parameters des Dokuments beschrieben werden, dann in der document-Umgebung der eigentliche Text. Z. B.:

```
\documentclass[12pt]{article}
\usepackage{german}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}

\begin{document}
```

Hier steht nun der `\textit{eigentliche}` Text des Dokuments.

```
\end{document}
```

Eine Anweisung in $\text{\LaTeX}$ beginnt immer mit einem Schrägstrich nach links, dem sogenannten *Backslash*, dann folgt der Befehlsname, dem in geschweiften Klammern Argumente übergeben werden können.

Eine Anweisung in $\text{\LaTeX}$ beginnt immer mit einem Schrägstrich nach links, dem sogenannten *Backslash*, dann folgt der Befehlsname, dem in geschweiften Klammern Argumente übergeben werden können.

`\documentclass` ist immer der erste Befehl in $\text{\LaTeX}$. Einige der wichtigsten Optionen für `\documentclass` lauten

`a4paper` Din A4-Format, falls Anpassung nötig

`11pt` etwas größere Schrift, auch `12pt`

`twocolumn` zweispaltige Ausgabe

Eine Anweisung in $\text{\LaTeX}$ beginnt immer mit einem Schrägstrich nach links, dem sogenannten *Backslash*, dann folgt der Befehlsname, dem in geschweiften Klammern Argumente übergeben werden können.

`\documentclass` ist immer der erste Befehl in $\text{\LaTeX}$. Einige der wichtigsten Optionen für `\documentclass` lauten

`a4paper` Din A4-Format, falls Anpassung nötig

`11pt` etwas größere Schrift, auch `12pt`

`twocolumn` zweispaltige Ausgabe

Mehrere Optionen können, mit Kommata getrennt, gleichzeitig genutzt werden, also etwa

```
\documentclass[12pt,twocolumn]{article}
```

Texte werden direkt in den Hauptteil innerhalb der `document`-Umgebung eingegeben, $\text{\LaTeX}$ kümmert sich dann automatisch um die Verteilung der Wörter pro Zeile, den Blocksatz, die Ausrichtung von Abständen, Trennung, Seitentrennung usw.

Texte werden direkt in den Hauptteil innerhalb der `document`-Umgebung eingegeben, $\text{\LaTeX}$ kümmert sich dann automatisch um die Verteilung der Wörter pro Zeile, den Blocksatz, die Ausrichtung von Abständen, Trennung, Seitentrennung usw.

Eine Leerzeile in der `.tex`-Datei bedeutet, daß $\text{\LaTeX}$ an dieser Stelle einen neuen Absatz beginnen soll. Abstände zwischen Absätzen, Leerzeilen usw. muß man mit Befehlen eingeben.

Texte werden direkt in den Hauptteil innerhalb der document-Umgebung eingegeben, $\text{\LaTeX}$ kümmert sich dann automatisch um die Verteilung der Wörter pro Zeile, den Blocksatz, die Ausrichtung von Abständen, Trennung, Seitentrennung usw.

Eine Leerzeile in der .tex-Datei bedeutet, daß $\text{\LaTeX}$ an dieser Stelle einen neuen Absatz beginnen soll. Abstände zwischen Absätzen, Leerzeilen usw. muß man mit Befehlen eingeben.

Leerzeichen und Zeilenenden werden überlesen, können aber mit „\ “ oder ~ erzwungen werden, falls nötig

Befehlszeichen, Befehle und Umgebungen (1)

Die folgenden zehn Zeichen haben in $\text{\LaTeX}$ eine besondere Bedeutung und können nicht als Zeichen im Text verwendet werden. Mit dem *backslash* kann man Sie aber bequem durch die darunterstehenden $\text{\LaTeX}$ -Befehle erhalten:

Zeichen: { } \$ & # % _ ^ ~ \

in $\text{\LaTeX}$: \{ \} \\$ \& \# \% _ \^{} \~{ }

Befehlszeichen, Befehle und Umgebungen (1)

Die folgenden zehn Zeichen haben in $\text{\LaTeX}$ eine besondere Bedeutung und können nicht als Zeichen im Text verwendet werden. Mit dem *backslash* kann man Sie aber bequem durch die darunterstehenden $\text{\LaTeX}$ -Befehle erhalten:

Zeichen: { } \$ & # % _ ^ ~ \

in $\text{\LaTeX}$: \{ \} \\$ \& \# \% _ \^{} \~{ }

Umlaute können problemlos in den Text eingefügt werden, sofern die Präambel die folgenden Befehle enthält:

```
\usepackage{german}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
```

Man stellt so sicher, daß $\text{\LaTeX}$ nach deutschen Trennregeln trennt, Umlaute richtig umsetzt usw.

Befehlszeichen, Befehle und Umgebungen (2)

Befehle ohne Parameter haben die Form `\Befehl`

Befehlszeichen, Befehle und Umgebungen (2)

Befehle ohne Parameter haben die Form `\Befehl`

Z.B. `\newline`

Befehlszeichen, Befehle und Umgebungen (2)

Befehle ohne Parameter haben die Form `\Befehl`

Z.B. `\newline`

Die Eingabe

`Text und \newline noch viel mehr Text`

Befehlszeichen, Befehle und Umgebungen (2)

Befehle ohne Parameter haben die Form `\Befehl`

Z.B. `\newline`

Die Eingabe

```
Text und \newline noch viel mehr Text
```

ergibt die Ausgabe

```
Text und  
noch viel mehr Text
```

Befehlszeichen, Befehle und Umgebungen (2)

Befehle ohne Parameter haben die Form `\Befehl`

Z.B. `\newline`

Die Eingabe

```
Text und \newline noch viel mehr Text
```

ergibt die Ausgabe

```
Text und  
noch viel mehr Text
```

`\\` ist eine bequeme Abkürzung für `\newline`

Befehlszeichen, Befehle und Umgebungen (3)

Die geschweiften Klammern { und } fassen einen Teil des Quellcodes zu einer Gruppe zusammen. Befehle innerhalb einer Gruppe oder Umgebung verlieren in der Regel außerhalb davon ihre Gültigkeit, wie man am Beispiel des Befehls `\itshape` sieht:

Befehlszeichen, Befehle und Umgebungen (3)

Die geschweiften Klammern { und } fassen einen Teil des Quellcodes zu einer Gruppe zusammen. Befehle innerhalb einer Gruppe oder Umgebung verlieren in der Regel außerhalb davon ihre Gültigkeit, wie man am Beispiel des Befehls `\itshape` sieht:

Die Eingabe

```
Text und {\itshape noch mehr Text} und dann  
wieder anderer Text
```

Befehlszeichen, Befehle und Umgebungen (3)

Die geschweiften Klammern { und } fassen einen Teil des Quellcodes zu einer Gruppe zusammen. Befehle innerhalb einer Gruppe oder Umgebung verlieren in der Regel außerhalb davon ihre Gültigkeit, wie man am Beispiel des Befehls `\itshape` sieht:

Die Eingabe

```
Text und {\itshape noch mehr Text} und dann  
wieder anderer Text
```

ergibt die Ausgabe

```
Text und noch mehr Text und dann wieder anderer Text
```

Befehlszeichen, Befehle und Umgebungen (4)

Im allgemeinen haben Befehle ein oder mehrere *Parameter*; optionale Parameter stehen in eckigen Klammern, notwendige Parameter in geschweiften Klammern. Bei einem optionalen und zwei notwendigen Parameters sieht ein allgemeiner Befehl so aus:

`\Befehl[optionalerParameter]{Parameter1}{Parameter2}`

Befehlszeichen, Befehle und Umgebungen (4)

Im allgemeinen haben Befehle ein oder mehrere *Parameter*; optionale Parameter stehen in eckigen Klammern, notwendige Parameter in geschweiften Klammern. Bei einem optionalen und zwei notwendigen Parameters sieht ein allgemeiner Befehl so aus:

```
\Befehl[optionalerParameter]{Parameter1}{Parameter2}
```

Zum Beispiel hat der Befehl `\textit` einen notwendigen Parameter, den er in kursiv setzt. Die Eingabe

Text mit einem `\textit{kursiven}` Wort

Befehlszeichen, Befehle und Umgebungen (4)

Im allgemeinen haben Befehle ein oder mehrere *Parameter*; optionale Parameter stehen in eckigen Klammern, notwendige Parameter in geschweiften Klammern. Bei einem optionalen und zwei notwendigen Parameters sieht ein allgemeiner Befehl so aus:

```
\Befehl[optionalerParameter]{Parameter1}{Parameter2}
```

Zum Beispiel hat der Befehl `\textit` einen notwendigen Parameter, den er in kursiv setzt. Die Eingabe

Text mit einem `\textit{kursiven}` Wort

ergibt die Ausgabe

Text mit einem *kursiven* Wort

Befehlszeichen, Befehle und Umgebungen (5)

Umgebungen haben die Form

```
\begin{Umgebungsname}
```

```
\end{Umgebungsname}
```

Befehlszeichen, Befehle und Umgebungen (5)

Umgebungen haben die Form

```
\begin{Umgebungsname}
```

```
\end{Umgebungsname}
```

Z. B. setzt die `itshape`-Umgebung allen dazwischenstehenden Text kursiv. Die Eingabe

Text

```
\begin{itshape}
```

und noch mehr

```
\end{itshape}
```

Text

Befehlszeichen, Befehle und Umgebungen (5)

Umgebungen haben die Form

```
\begin{Umgebungsname}
```

```
\end{Umgebungsname}
```

Z. B. setzt die `itshape`-Umgebung allen dazwischenstehenden Text kursiv. Die Eingabe

```
Text  
\begin{itshape}  
  und noch mehr  
\end{itshape}  
Text
```

ergibt die Ausgabe

```
Text und noch mehr Text
```

Klassen

Die verschiedenen Klassen, mit der ein Dokument mittels spezifiziert werden kann, lauten

article für mittelgroße und kleine Dokumente

report für umfangreiche Dokumente

book für komplette Bücher

Klassen

Die verschiedenen Klassen, mit der ein Dokument mittels spezifiziert werden kann, lauten

`article` für mittelgroße und kleine Dokumente

`report` für umfangreiche Dokumente

`book` für komplette Bücher

Auch die Benutzung selbst definierter Klassen ist möglich, wenn die Klassendatei `bsp.cls` im selben Verzeichnis wie die `.tex`-Datei steht, die `bsp.cls` als Klasse benutzt. Oder wenn die Klasse als Klassenpaket mitinstalliert ist.

Klassen

Die verschiedenen Klassen, mit der ein Dokument mittels spezifiziert werden kann, lauten

`article` für mittelgroße und kleine Dokumente

`report` für umfangreiche Dokumente

`book` für komplette Bücher

Auch die Benutzung selbst definierter Klassen ist möglich, wenn die Klassendatei `bsp.cls` im selben Verzeichnis wie die `.tex`-Datei steht, die `bsp.cls` als Klasse benutzt. Oder wenn die Klasse als Klassenpaket mitinstalliert ist.

Auch die Klasse `beamer.cls` funktioniert z. B. entweder so oder als mitinstalliertes Klassenpaket.

Pakete

Für verschiedene Anwendungen steht eine große Auswahl an Paketen zur Verfügung, die mit `\usepackage` und eventuellen Optionen in die Präambel eingefügt werden können. Diese stellen weitere Kommandos zur Verfügung.

Pakete

Für verschiedene Anwendungen steht eine große Auswahl an Paketen zur Verfügung, die mit `\usepackage` und eventuellen Optionen in die Präambel eingefügt werden können. Diese stellen weitere Kommandos zur Verfügung.

Beispiele:

`(n)german` für deutsche Texte

`graphicx` für Bildereinbindung

`mutlicol` für mehrspaltigen Text

`longtable` für mehrseitige Tabellen

`amsmath`, `amsthm`, `amssymb` umfangreiche Mathematik-Pakete

Pakete

Für verschiedene Anwendungen steht eine große Auswahl an Paketen zur Verfügung, die mit `\usepackage` und eventuellen Optionen in die Präambel eingefügt werden können. Diese stellen weitere Kommandos zur Verfügung.

Beispiele:

`(n)german` für deutsche Texte

`graphicx` für Bildereinbindung

`multicol` für mehrspaltigen Text

`longtable` für mehrseitige Tabellen

`amsmath`, `amsthm`, `amssymb` umfangreiche Mathematik-Pakete

Mit `\usepackage` können auch mehrere Pakete gleichzeitig eingeladen werden, z. B.

```
\usepackage{german,multicol,amsmath}
```