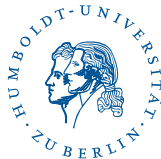


HUMBOLDT-UNIVERSITÄT ZU BERLIN



# Wissenschaftliches Arbeiten in der Linguistik (Technische Übung)

TEX 1: Grundlagen

Antonio Machicao y Priemer

[www.linguistik.hu-berlin.de/staff/amyp](http://www.linguistik.hu-berlin.de/staff/amyp)

Institut für deutsche Sprache und Linguistik

# Inhaltsverzeichnis

## 1 Was ist $\text{\LaTeX}$ ?

- Geschichte
- WYSIWYG vs. WYGIWYN
- Warum sollte ich es benutzen?
- Wie funktioniert  $\text{\LaTeX}$ ?
- Software

## 2 Befehle

## 3 Zeichen & Umbrüche

- Zeichen & Sonderzeichen
- Leerzeichen & Zeilenumbrüche

## 4 (Aus-)Kommentieren

## 5 Hausaufgabe 0

# Grundlage & empfohlene Lektüre

...basierend auf Freitag und Machicao y Priemer (2015) und auf Machicao y Priemer und Kerkhof (2016)

→ LINK

1 Was ist  $\text{\LaTeX}$ ?

2 Befehle

3 Zeichen & Umbrüche

4 (Aus-)Kommentieren

5 Hausaufgabe 0

# Geschichte

- $\TeX$  ( $\TeX$ ) wurde zwischen 1977 und 1986 von Donald E. Knuth entwickelt, da er mit den damaligen Textverarbeitungsprogrammen unzufrieden war.
- $\LaTeX$  ist ein Interface, das eine Sammlung an nützlichen **Makros** zur Verfügung stellt, um einfacher das  $\TeX$ -System verwenden zu können.
- $\LaTeX$  wurde von Leslie Lamport geschrieben. Der Name ist eine Abkürzung für **Lamport TeX**.
- Aussprache:  $[ˈlaː.tɛç]$
- Siehe auch Kopka (1994).

# WYSIWYG vs. WYGIWYN

- *MS Word oder Libre Office:*  
→ WYSIWYG-Prinzip (*what-you-see-is-what-you-get*)
- $\text{\LaTeX}$ :  
→ WYGIWYN-Prinzip (*what-you-get-is-what-you-need*) oder  
→ WYGIWYM-Prinzip (*what-you-get-is-what-you-mean*)
- $\text{\LaTeX}$  arbeitet mit einer **logischen Textauszeichnung** – ähnlich wie HTML – um Textelemente zu definieren.

# Warum sollte ich es benutzen?

- Zeitersparnis (nicht am Anfang!)

# Warum sollte ich es benutzen?

- Zeitersparnis (nicht am Anfang!)
- Man hört auf, Sachen zu tun, die der Computer für einen erledigen kann. Daher hat man mehr Zeit, um sich mit dem Inhalt zu befassen.



# Warum sollte ich es benutzen?

- Zeitersparnis (nicht am Anfang!)
- Man hört auf, Sachen zu tun, die der Computer für einen erledigen kann. Daher hat man mehr Zeit, um sich mit dem Inhalt zu befassen.
- Die Texte werden sehr gut gesetzt!

# Warum sollte ich es benutzen?

- Zeitersparnis (nicht am Anfang!)
- Man hört auf, Sachen zu tun, die der Computer für einen erledigen kann. Daher hat man mehr Zeit, um sich mit dem Inhalt zu befassen.
- Die Texte werden sehr gut gesetzt!
- Ein Programm für alle Funktionen: Artikel, Bücher, Poster, Präsentationen, ...

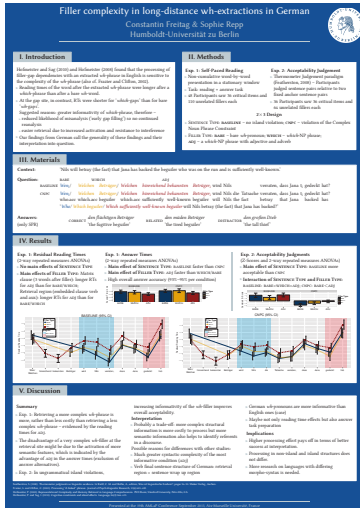
# Warum sollte ich es benutzen?

- Zeitersparnis (nicht am Anfang!)
- Man hört auf, Sachen zu tun, die der Computer für einen erledigen kann. Daher hat man mehr Zeit, um sich mit dem Inhalt zu befassen.
- Die Texte werden sehr gut gesetzt!
- Ein Programm für alle Funktionen: Artikel, Bücher, Poster, Präsentationen, ...
- Gratis!!

Einige Beispiele, was man mit  $\text{\LaTeX}$  tun kann:

## 9 / 38

# Poster & Briefe



Universität  
Konstanz



Prof. Dr. Marc H. Scholl

FB Informatik & Informationswissenschaft

Universität Konstanz - Postfach 78057 Konstanz

Frau

Julia Werd

Universität Konstanz

- Kommunikation und Marketing -

Universitätsstraße 10

78464 Konstanz

16. 10. 2014

Die KOMA-Script Letter Klasse umbricht und eine Briefvorlage für  $\text{\LaTeX}$ -Nutzer

Seite 1/1

Sehr geehrte Frau Werd,

hier der Standardbrief mit Bankleiste im Fuß.

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

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

Mit freundlichen Grüßen

Marc H. Scholl

# Bäume

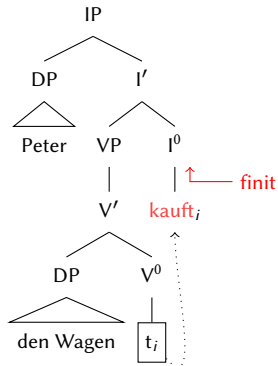


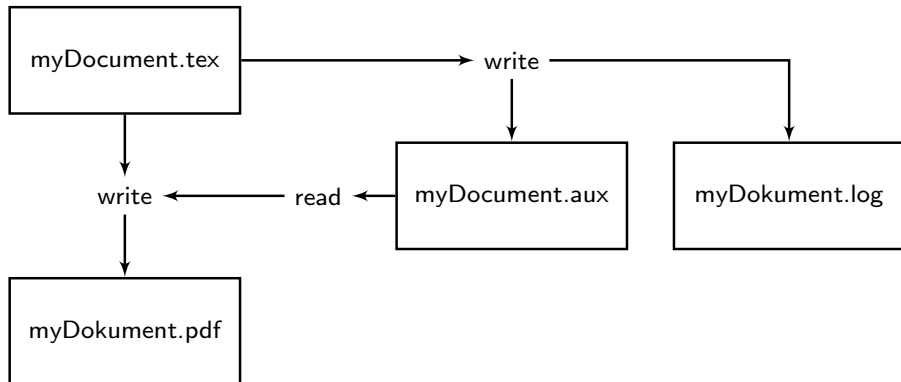
Abb. 1: Kopfbewegung

# Glossen & IPA

- (1) Der Mann hat dem Jungen ein Buch über Linguistik  
the man.NOM has the boy.DAT a book.ACC about linguistics  
gegeben.  
give.PTCP.PRF/gave  
'The man gave the boy a book about linguistics.'
- (2) ⟨phonetics⟩  
/fə'.nɛ.tɪks/  
[fə'nɛtɪks]



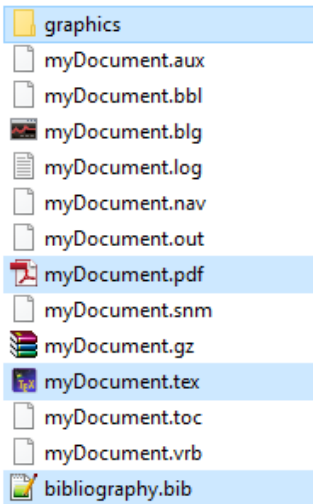
# Wie funktioniert $\text{\LaTeX}$ ?



Beim Kompilieren kreiert  $\text{\LaTeX}$  eine Menge an **Hilfsdateien**, um die Kompilierungsprozesse zu optimieren.

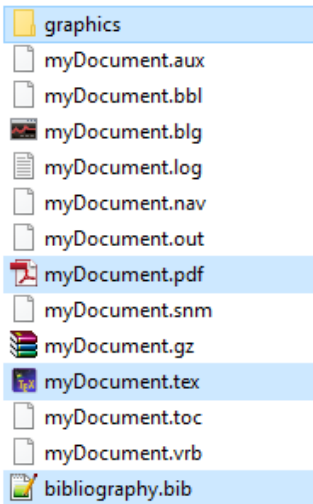
Diese Hilfsdateien können **gelöscht** werden.

- `.bb1` → Information für das Literaturverzeichnis
- `.nav` → Information für die Foliennavigation
- `.toc` → Information für das Inhaltsverzeichnis
- ...



Dateien, die für Sie **wichtig** sind  
(**NICHT LÖSCHEN!**):

- `.tex` → Das ist die Datei, die Sie bearbeiten, d. h. das ist Ihr Dokument!
- `.pdf` → das Ergebnis Ihres Dokuments als PDF
- `.bib` → diese Datei enthält Ihre Bibliographie
- Ordner `graphics` → Hier sind die Abbildungen, die Sie verwenden



# Software

In diesem Kurs werden wir mit dem **Editor** TeXstudio arbeiten, welcher folgende Vorteile anbietet:

- kostenlos,
- kompatibel mit Linux-, Windows- und Mac OS-Rechnern,
- Unicode-Unterstützung, Rechtschreibkontrolle, Autovervollständigung von Befehlen, PDF-Viewer,
- einfach zu verwenden und zu konfigurieren, ...

Außer dem Editor benötigt man eine L<sup>A</sup>T<sub>E</sub>X-**Distribution**. In diesem Fall werden wir MiKTeX (für Windows-User) verwenden. Für Linux-User ist TeXLive eine sehr bekannte Alternative und für Mac OS-User MacTeX. **Zuerst** soll die Distribution (z. B. MiKTeX) und erst dann der Editor (z. B. TeXstudio) installiert werden!

1 Was ist  $\text{\LaTeX}$ ?

2 **Befehle**

3 Zeichen & Umbrüche

4 (Aus-)Kommentieren

5 Hausaufgabe 0

# Befehle (commands)

- $\text{\LaTeX}$  arbeitet mit einer **logischen Textauszeichnung**
- Wenn Sie etwas kursiv setzen wollen, müssen Sie nicht irgendwo „klicken“, sondern **einen Befehl eingeben**

## Syntax von Befehlen:

## Syntax von Befehlen:

- Alle  $\text{\LaTeX}$ -Befehle beginnen mit einem **Backslash** „\“,
- gefolgt vom **Namen des Befehls**,
- von optionalen Argumenten in **eckigen Klammern** und
- von obligatorischen Argumenten in **geschweiften Klammern**.

```
\Befehlsname[optional]{obligatorisch}  
\Befehlsname[opt1, opt2=Wert]{obl1}{obl2}
```



## Syntax von Befehlen:

- Alle  $\text{\LaTeX}$ -Befehle beginnen mit einem **Backslash** „\“,
- gefolgt vom **Namen des Befehls**,
- von optionalen Argumenten in **eckigen Klammern** und
- von obligatorischen Argumenten in **geschweiften Klammern**.

```
\Befehlsname[optional]{obligatorisch}
\Befehlsname[opt1, opt2=Wert]{obl1}{obl2}
```

- Beispiele:

```
\textbf{fettgedruckter Text}
\cite[vgl.][3--5]{Freitag&MyP15a}
```

## Syntax von Befehlen:

- Alle  $\text{\LaTeX}$ -Befehle beginnen mit einem **Backslash** „ $\backslash$ “,
- gefolgt vom **Namen des Befehls**,
- von optionalen Argumenten in **eckigen Klammern** und
- von obligatorischen Argumenten in **geschweiften Klammern**.

```
\Befehlsname[optional]{obligatorisch}
\Befehlsname[opt1, opt2=Wert]{obl1}{obl2}
```

- Beispiele:

```
\textbf{fettgedruckter Text}
\cite[vgl.][3--5]{Freitag&MyP15a}
```

- (3) **fettgedruckter Text**
- (4) (vgl. Freitag und Machicao y Priemer, 2015: 3–5)

LaTeX kennt im Prinzip **drei Arten von Befehlen**:

- **Einfache Befehle:**

Backslash „\“ + Befehlsnamen + optionale Argumenten in eckigen Klammern + obligatorische Argumenten in geschweiften Klammern

```
\Befehlsname[optional]{obligatorisch}  
\Befehlsname[opt1, opt2=Wert]{obl1}{obl2}
```

LaTeX kennt im Prinzip **drei Arten von Befehlen**:

- **Einfache Befehle:**

Backslash „\“ + Befehlsnamen + optionale Argumenten in eckigen Klammern + obligatorische Argumenten in geschweiften Klammern

```
\Befehlsname[optional]{obligatorisch}  
\Befehlsname[opt1, opt2=Wert]{obl1}{obl2}
```

- **Beispiele:**

```
\textit{kursive Hervorhebung}  
\citet[vgl.][3--5]{Freitag&MyP15a}
```

LaTeX kennt im Prinzip **drei Arten von Befehlen**:

- **Einfache Befehle:**

Backslash „\“ + Befehlsnamen + optionale Argumenten in eckigen Klammern + obligatorische Argumenten in geschweiften Klammern

```
\Befehlsname[optional]{obligatorisch}
\Befehlsname[opt1, opt2=Wert]{obl1}{obl2}
```

- **Beispiele:**

```
\textit{kursive Hervorhebung}
\citet[vgl.][3--5]{Freitag&MyP15a}
```

(5) *kursive Hervorhebung*

(6) Freitag und Machicao y Priemer (vgl. 2015: 3–5)

- **Umgebungen:**

Umgebungen bestehen aus einem `begin`- und einem `end`-Befehl. Der Befehl gilt im Bereich zwischen `begin` und `end`.

```
\begin{Umgebung} [optional]  
...  
\end{Umgebung}
```

- **Umgebungen:**

Umgebungen bestehen aus einem `begin`- und einem `end`-Befehl. Der Befehl gilt im Bereich zwischen `begin` und `end`.

```
\begin{Umgebung} [optional]  
...  
\end{Umgebung}
```

- **Beispiele:**

```
\begin{multicols}{2}[Beispiel]  
Das ist die erste Spalte  
  
Das ist die zweite Spalte  
\end{multicols}
```

- **Umgebungen:**

Umgebungen bestehen aus einem `begin`- und einem `end`-Befehl. Der Befehl gilt im Bereich zwischen `begin` und `end`.

```
\begin{Umgebung} [optional]
...
\end{Umgebung}
```

- **Beispiele:**

```
\begin{multicols}{2}[Beispiel]
Das ist die erste Spalte

Das ist die zweite Spalte
\end{multicols}
```

(7) Beispiel

Das ist die erste Spalte

Das ist die zweite Spalte



- **Deklarationen:**

Deklarationen verändern Parameter.

Der **Skopus** von Deklarationen kann so definiert sein, dass er an bestimmten Grenzen – wie an einem Absatzschluss – endet, oder dass er nur auf einen **von geschweiften Klammern bestimmten Skopus** beschränkt ist.

```
\Deklaration ... [Skopusende]  
{\Deklaration ...} ausserhalb des Skopus
```

- **Deklarationen:**

Deklarationen verändern Parameter.

Der **Skopus** von Deklarationen kann so definiert sein, dass er an bestimmten Grenzen – wie an einem Absatzschluss – endet, oder dass er nur auf einen **von geschweiften Klammern bestimmten Skopus** beschränkt ist.

```
\Deklaration ... [Skopusende]  
{\Deklaration ...} ausserhalb des Skopus
```

- **Beispiele:**

```
{\small Hello world!} \Huge Hello world!
```

- **Deklarationen:**

Deklarationen verändern Parameter.

Der **Skopus** von Deklarationen kann so definiert sein, dass er an bestimmten Grenzen – wie an einem Absatzschluss – endet, oder dass er nur auf einen **von geschweiften Klammern bestimmten Skopus** beschränkt ist.

```
\Deklaration ... [Skopusende]
{\Deklaration ...} ausserhalb des Skopus
```

- Beispiele:

```
{\small Hello world!} \Huge Hello world!
```

(8)    Hello world! **Hello world!**

1 Was ist  $\text{\LaTeX}$ ?

2 Befehle

3 Zeichen & Umbrüche

4 (Aus-)Kommentieren

5 Hausaufgabe 0

# Zeichen & Sonderzeichen

- Die folgenden Zeichen können problemlos verwendet werden:

```
a...z A...Z 0...9
. , : ; ? ! ' ' " ( ) [ ] + - * =
```

- Achten Sie darauf, welche Art von **Anführungszeichen** durch ‘ ’ “ generiert werden (vgl. Machicao y Priemer, 2018).
- Die **Umlaute** „ä, ö, Ä, Ö, ...“, **Akzente** „á, à, ...“ und das **Eszett** „ß“ können (bei PDF- $\text{\LaTeX}$ ) mithilfe des folgenden Pakets `\usepackage[utf8]{inputenc}` direkt eingegeben werden. Andernfalls müssen sie mit Extrabefehlen geschrieben werden:

```
\ "A \ "O \ "a \ "o \ 'a \ 'o \ ss {}
oder \ " {A} \ " {O} \ {ss}
```

(9) Ä Ö ä ö á à ß oder Ä Ö ß

- Die folgenden Zeichen haben in  $\text{\LaTeX}$  eine **besondere Bedeutung** und können nicht einfach im Fließtext verwendet werden:

# \$ & ~ \_ ^ { } < > | \ %

- Die folgenden Zeichen haben in  $\text{\LaTeX}$  eine **besondere Bedeutung** und können nicht einfach im Fließtext verwendet werden:

# \$ & ~ \_ ^ { } < > | \ %

- Um diese Zeichen verwenden zu können, musst man den in  $\text{\LaTeX}$  vordefinierten Funktionen dieser Zeichen **entkommen**. Bei einigen Zeichen kann man den vordefinierten Funktionen durch **Voranstellen eines Backslashes** entkommen.

\# \\$ \% \\_ \{ \} \%

- Dem **Backslash**, der **Größer-als**- und **Kleiner-als**-Zeichen, der **Tilde**, dem **Zirkumflex** und dem **senkrechten Strich** (*pipe*) kann man nicht mit dem Backslash entkommen.



- Da die Folge `\\` für **Zeilenumbrüche** reserviert ist, kann man dem einfachen **Backslash** „\“ nicht mit Verwendung eines vorangestellten Backslashes entkommen. Dafür sollte der folgende Befehl benutzt werden:

```
\textbackslash
```

- Die **Größer-als-** „>“ und **Kleiner-als-Symbole** „<“ können im Text durch die folgenden Befehle oder durch die Verwendung des Mathematikmodus', d. h. durch die **Klammerung in \$-Zeichen** erzeugt werden (mehr zum Mathematikmodus später).

```
\textgreater $>$  
\textless $<$
```

- Um den **senkrechten Strich** („pipe“) darzustellen, kann man entweder den Befehl **vert** oder den Strich in der **Mathematikmodusklammerung** eingeben oder den Befehl **textbar** außerhalb des Mathematikmodus’.

```
$\textbf{vert}$ $| $ \textbar
```

- Die **Tilde** „~“ hat in  $\text{\LaTeX}$  die Funktion eines geschützten Leerzeichens. Um dieser Funktion zu entkommen, kann man nicht den Backslash verwenden (`\~`), denn dadurch erscheint der folgende Buchstabe mit einer Tilde. So bei der Eingabe „`\~nicht`“, erscheint „ñicht“. Will man auch dieser Funktion entkommen, muss der folgende Befehl (ähnlich wie bei dem Backslash) benutzt werden:

```
\textasciitilde
```

- Das gleiche Problem taucht beim **Zirkumflex** „^“ auf, welcher als Akzent z. B. im Französischen gebraucht wird. Daher erscheint bei der Eingabe „s\^ur“ der folgende Output: „sûr“. Aus diesem Grund benötigt man den folgenden Befehl um den Zirkumflex als Output zu haben:

```
\textasciicircum
```

- Weiteres zu Sonderzeichen in  $\text{\LaTeX}$ :  
[https://de.wikibooks.org/wiki/LaTeX/\\_Akzente\\_und\\_Sonderzeichen](https://de.wikibooks.org/wiki/LaTeX/_Akzente_und_Sonderzeichen)

# Leerzeichen & Zeilenumbrüche

- $\text{\LaTeX}$  hat eine **gesonderte Behandlung von Leerzeichen**, die viele typographische Fehler automatisch korrigiert.
- Es macht **keinen Unterschied** zwischen einem Leerzeichen (,blank‘) oder einem Tabulator (,tab‘).
- Es zählt **keine aufeinanderfolgenden Leerzeichen**, d. h. mehrere konsequente Leerzeichen werden nur als eins behandelt.
- Ein Leerzeichen zu **Beginn einer Zeile** wird einfach ignoriert.
- Ein **Zeilenumbruch** im Code wird als einzelnes Leerzeichen interpretiert.
- Eine **Leerzeile** (d. h. zwei Zeilenumbrüche hintereinander) legen das Ende eines Absatzes fest.
- **Mehr Leerzeilen** (oder Zeilenumbrüche) werden als *eine* einzelne Leerzeile interpretiert.

Hier ein Beispiel:

Hier ist ein Beispieltext mit viel  
zu vielen Leerzeichen .  
In Word sind sie immer zu sehen.  
Hier verwenden wir einen  
Zeilenumbruch.

Zwei Zeilenumbrüche ergeben einen neuen Absatz.

Mehr als zwei Umbrüche ergeben nur einen neuen Absatz.

Hier ist ein Beispieltext mit viel zu vielen Leerzeichen . In Word sind sie immer zu sehen. Hier verwenden wir einen Zeilenumbruch.  
Zwei Zeilenumbrüche ergeben einen neuen Absatz.  
Mehr als zwei Umbrüche ergeben nur einen neuen Absatz.

1 Was ist  $\text{\LaTeX}$ ?

2 Befehle

3 Zeichen & Umbrüche

4 (Aus-)Kommentieren

5 Hausaufgabe 0

# (Aus-)Kommentieren

Das % ist das Zeichen um  $\text{\LaTeX}$ -Code auszukommentieren, d. h.  $\text{\LaTeX}$  wird den gesamten folgenden Text bis zum Zeilenumbruch **ignorieren**. Der Text nach dem Prozentzeichen wird weder interpretiert noch im Output wiedergegeben.

Kommentare sind sehr hilfreich beim Programmieren.

Durchs Auskommentieren kann man:

- **Code/Text verstecken**, ohne ihn zu löschen;
- leichter in Zeilen oder größeren Regionen **Fehler** finden;
- **Leerzeichen oder Leerzeilen** in langen Eingabezeilen **unterbinden**;
- **Kommentare in den Code schreiben**, ohne dass sie als Text gedruckt werden

# (Aus-)Kommentieren

Hier ist etwas Code, der angezeigt werden soll.

*%hier sind wichtige Notizen*

Kommentare können sogar ein Wort teilen:

Rindfleischetikettierungs*% Notiz: Fugen-s*

überwachungsaufgaben*% Notiz: Fugen-n*

übertragungsgesetz.

Hier ist etwas Code, der angezeigt werden soll.

Kommentare können sogar ein Wort teilen: Rindfleischetikettierungsüberwachungsaufgabenübertragungsgesetz.



1 Was ist  $\text{\LaTeX}$ ?

2 Befehle

3 Zeichen & Umbrüche

4 (Aus-)Kommentieren

5 Hausaufgabe 0

# Hausaufgabe 0: $\text{\LaTeX}$ -Vorbereitung

- Laden Sie `MiKTeX` und `TeXstudio` – wie in der Anleitung in Moodle angegeben – herunter.
- Installieren Sie beide Programme.
- Folgen Sie dabei der Anleitung in Moodle.
- Falls Sie **Probleme bei der Installation** haben, melden Sie sich bitte bei Pia Linscheid *vor* der nächsten Sitzung! Andernfalls werden Sie die kommenden Hausaufgaben nicht abgeben können.
- **Alternative:** Anstatt die Programme zu installieren, können Sie versuchen Ihre Hausaufgaben mit Overleaf zu lösen (Siehe Anleitung in Moodle).

# Hausaufgabe 0: Lektüre

- Lesen Sie den **LingStudi-Guide** (s. Moodle/Allgemeines)

# Hausaufgabe 0: Mitgestalten

- Schreiben Sie bei Moodle im Bereich „Was will ich in diesem Kurs lernen?“ **einen bis zwei Stichpunkte** auf. Erklären Sie kurz – wenn nötig – was Sie damit meinen.

# Quellen I

- Grafik: File Extensions – xkcd, A webcomic of romance, sarcasm, math, and language  
<https://xkcd.com/1301/>  
[Zugriff: 10.04.2017]
- Link: Akzente und Sonderzeichen in  $\text{\LaTeX}$ .  
[https://de.wikibooks.org/wiki/LaTeX/\\_Akzente\\_und\\_Sonderzeichen](https://de.wikibooks.org/wiki/LaTeX/_Akzente_und_Sonderzeichen)  
[Zugriff: 10.10.2017]
- Software: MiKTeX  
<https://miktex.org/>  
[Zugriff: 10.04.2017]
- Software: TeXstudio  
<https://www.texstudio.org/>  
[Zugriff: 10.04.2017]

# Literatur I

- Freitag, Constantin und Antonio Machicao y Priemer (2015).  $\LaTeX$ -Einführung für Linguisten. Manuskript. URL <https://www.linguistik.hu-berlin.de/de/staff/amp/latex-einfuehrung>, Zugriff: 12.04.2015.
- Knuth, Donald E. (1986). *The  $\TeX$ book*. Boston, MA: Addison-Wesley.
- Kopka, Helmut (1994).  *$\LaTeX$ : Einführung. Band 1*. Bonn: Addison-Wesley.
- Machicao y Priemer, Antonio (2018). Hinweise für Seminararbeiten. Manuskript. URL [https://www.linguistik.hu-berlin.de/de/staff/amp/downloads/myp2018-04-06-hinweise\\_seminararbeit.pdf](https://www.linguistik.hu-berlin.de/de/staff/amp/downloads/myp2018-04-06-hinweise_seminararbeit.pdf), Zugriff: 16.10.2018.
- Machicao y Priemer, Antonio und Robyn Kerkhof (2016).  $\LaTeX$ -Einführung für Linguisten – Slides. Präsentation beim 7. linguistischen Methodenworkshop an der Humboldt-Universität zu Berlin – 22.–24. Februar 2016. URL <https://www.linguistik.hu-berlin.de/de/staff/amp/latex-einfuehrung>, Zugriff: 23.02.2016.