

L^AT_EX

Programmieren - Längen

LaTeX-Programmierung

- Das zu Grunde liegende TeX-System ist eine Programmiersprache.
- LaTeX ist eine Sammlung von Makros die in TeX programmiert sind.
- Es gibt entsprechend für den Anwender viele Elemente einer Programmiersprache um den Textsatz zu gestalten.
 - ▶ Variablen: alle Format-Eigenschaften sind in Variablen abgelegt, die programmatische geändert werden können.
 - ▶ Diese Variablen können über viele LaTeX-Befehle verändert werden.
 - ▶ Es gibt verschiedene Variablen-Typen (vor allem Längen und Zähler)
 - ▶ Auf TeX-Ebene: Kontrollstrukturen (if then, Schleifen, ...)

Beispiel: `\setlength`

\setlength

Paragraph Indentation: Absatz-Einzug

Paragraph Skip: Absatz-Lücke

- In LaTeX treten sehr viele Längen auf.
- Diese können über den Befehl `\setlength` verändert werden.

```
\documentclass[10pt]{article}
```

```
\setlength{\parindent}{10pt}
```

```
\setlength{\parskip}{0.5\baselineskip}
```

```
\begin{document}
```

Hier steht der eigentliche Text des Dokumentes.

```
\end{document}
```

\setlength

- Hier treten gleich drei Längen auf:
 - ▶ `\parindent`: definiert den Einzug eines Absatzes.
 - ▶ `\parskip`: definiert den Abstand zwischen zwei Absätzen.
 - ▶ `\baselineskip`: Der Zeilenabstand.

```
\documentclass[10pt]{article}
```

```
\setlength{\parindent}{10pt}
```

```
\setlength{\parskip}{0.5\baselineskip}
```

```
\begin{document}
```

Hier steht der eigentliche Text des Dokumentes.

```
\end{document}
```

\setlength

- In dem Beispiel wird also der Einzug auf 10pt gesetzt.
- Der Abstand zwischen zwei Absätzen wird auf die Höhe einer halben Zeile gesetzt.
- **Alle Längen haben eine Einheit!**

```
\documentclass[10pt]{article}
```

```
\setlength{\parindent}{10pt}
```

```
\setlength{\parskip}{0.5\baselineskip}
```

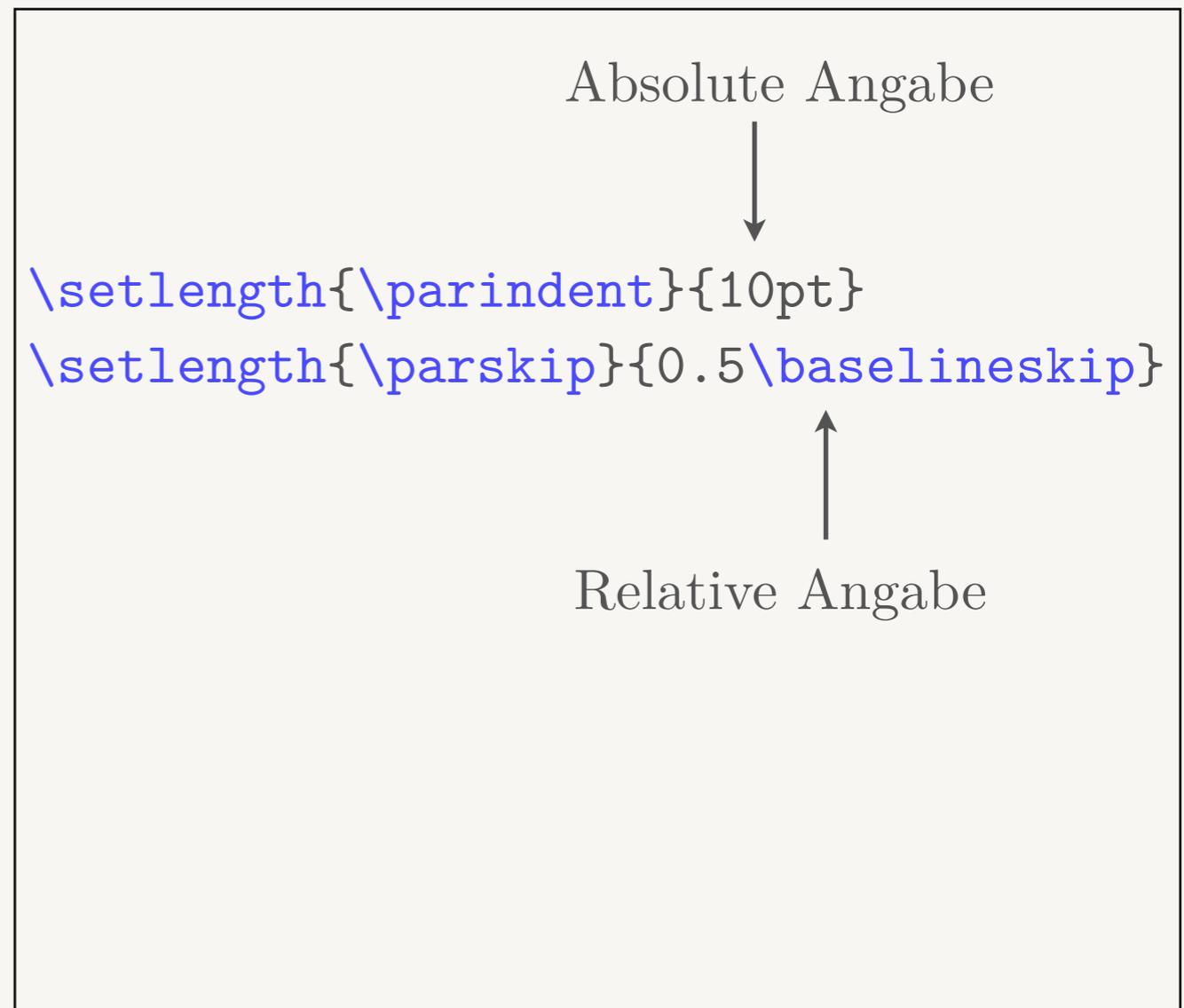
```
\begin{document}
```

Hier steht der eigentliche Text des Dokumentes.

```
\end{document}
```

Relative und absolute Länge

- Relative Längenangaben beziehen sich auf einen Referenzwert.
- Absolute Längenangaben werden mit Maßzahl und Einheit direkt angegeben.
- Vergleiche auch die Diskussion zwischen logischer und physischer Formatierung.



\setlength

- **Gebrauchen Sie immer relative Längenangaben!**
- Schreiben Sie also nicht 10pt als Argument für den `\setlength`-Befehl (absolute Länge) (Ausnahme: 0pt).
- Relative Längenangaben erledigen die Arbeit für Sie, wenn Sie z.B. den Schriftgrad von 10pt auf 12pt ändern. Dann ändert sich der Abstand zwischen den Absätzen genau richtig mit.
- Das gilt für alle Längenangaben, z.B. Breite von Bildern, Spaltenweite von Tabellen, ...

Aufgabe

- Kopieren Sie das Beispiel.
- Ändern Sie den `\parskip` wie Sie möchten und beobachten die Ausgabe.
- Relativ und absolut:
 - ▶ Ändern Sie den Schriftgrad von 9pt auf 14pt. Achten Sie auf den Absatz-Abstand in der Ausgabe.
 - ▶ Setzen Sie `\parskip` auf 10pt absolut fest und ändern dann den Schriftgrad zwischen 9pt und 14pt.

```
\documentclass[10pt]{article}

\setlength{\parindent}{10pt}
\setlength{\parskip}{0.5\baselineskip}

\begin{document}

Hier steht der eigentliche Text des
Dokumentes.

\end{document}
```

Längen

<http://en.wikibooks.org/wiki/LaTeX/Lengths>

Längeneinheiten

- Jede Länge hat eine Einheit.
- Diese **muss** mit angegeben werden!

Name	LaTeX	Wert (in pt)
Punkt	<code>pt</code>	1
Millimeter	<code>mm</code>	2.84
Zentimeter	<code>cm</code>	28.4
Inch	<code>in</code>	72.27

<http://en.wikibooks.org/wiki/LaTeX/Lengths#Units>

Längenangaben

- Längen können absolut mit Einheit angegeben werden.
- Längen können als Multiplikation mit einer Variablen angegeben werden.
- Addition geht nicht! Das ist etwas anderes (*Stretching*).

Absolut mit Einheit



```
\setlength{\parindent}{10pt}  
\setlength{\parskip}{0.5\baselineskip}
```



Multiplikation mit Variable

Längen anzeigen

- Wenn Sie eine Länge anzeigen wollen müssen Sie `\the` gebrauchen.
- `\the` zeigt den Wert einer Variablen an.
- `\the` geht für fast alles. Einfach ausprobieren.

```
\the\baselineskip
```

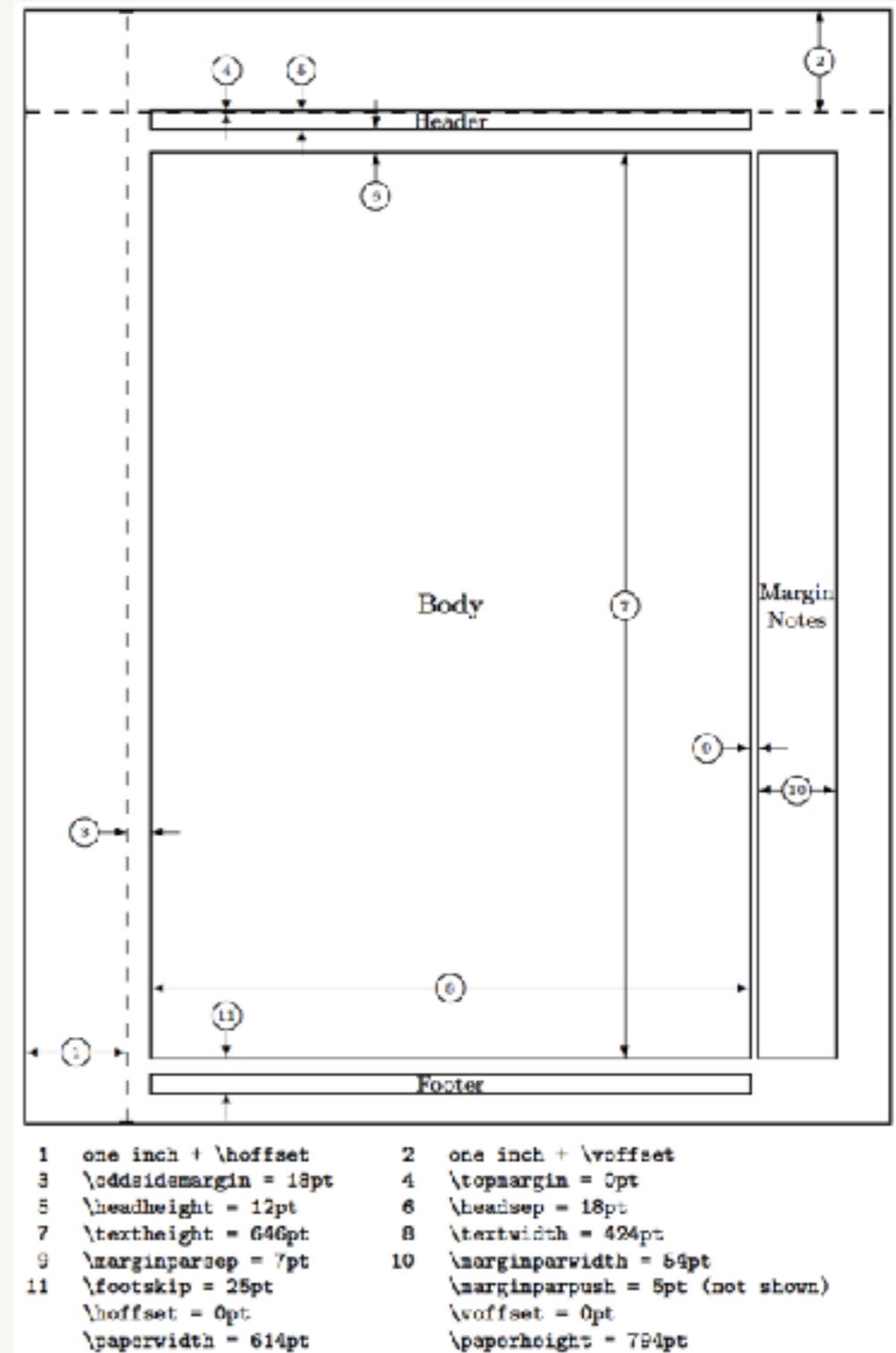
Aufgabe

- Lassen Sie sich die folgenden Werte mit `\the` anzeigen:
 - ▶ `\textheight`
 - ▶ `\textwidth`
 - ▶ `\paperheight`
 - ▶ `\paperwidth`

Seitenlängen

- Die wichtigsten Längen für die Gestaltung der Seite sind hier dargestellt.
- Diese Information können Sie über den Befehl `\layout{}` erhalten.
- Dazu müssen Sie das Paket `layout` mit `\usepackage` laden.
- Typischerweise wollen Sie diese Längen nicht modifizieren sondern als Referenz gebrauchen.

https://en.wikibooks.org/wiki/LaTeX/Page_Layout



Vordefinierte Längen

- Interessante weitere Längen sind:
 - ▶ `\columnsep`, `\columnwidth` (bei mehrspaltiger Gestaltung)
 - ▶ `\tabcolsep` (für Tabellen)
 - ▶ `\itemsep` (für Listen)

- Eine Liste finden Sie unter diesem Link:

http://en.wikibooks.org/wiki/LaTeX/Lengths#LaTeX_default_lengths

Längen ändern

Längenänderungsbefehle

- Längen lassen sich über diverse Befehle verändern:
 - ▶ `\setlength`
 - ▶ `\addtolength`
 - ▶ `\settowidth`
 - ▶ `\settoheight`
 - ▶ `\settodepth`

Längenänderungsbefehle

- `\setlength` setzt eine Länge auf einen Wert und hat zwei Argumente:

```
\setlength{\länge}{wert}
```

- `\addtolength` addiert einen Wert zu einer Länge und hat zwei Argumente:

```
\addtolength{\länge}{wert}
```

Aufgabe

- Verlängern Sie nacheinander mit `\addtolength` den Einzug `\parindent` um
 - ▶ 1pt
 - ▶ 1mm
 - ▶ 1cm
 - ▶ 1in
- Machen Sie für jeden Befehl einen neuen Absatz, um untereinander das Ergebnis zu sehen.

Längenänderungsbefehle

- Die Befehle `\settoheight`,
`\settowidth` und `\settodepth`
setzen eine Länge auf die Höhe,
Breite oder Tiefe eines
Beispieltextes.
- Der Beispieltext wird nicht
angezeigt.
- Ist eher interessant für selbst
definierte Längen (s. dort).
- Der Beispieltext kann selber das
Ergebnis eines Befehls sein.

Zwei Argumente:
Länge
Beispieltext

```
\settowidth{\parindent}{Digga bin ich  
breit!}  
\the\parindent
```

Stretching

Längen strecken

Quelle: 

- Um ein schönes Layout zu erreichen streckt LaTeX einige Größen.
- Beispiele:
 - Abstand zwischen Wörtern für Blocksätze.
 - Abstände zwischen Absätzen um die Seite optimal auszunutzen.
- Wenn das nicht klappt meckert LaTeX über `overfull` oder `underfull \vbox`.
- Damit beschäftigen wir uns heute nicht.



http://en.wikipedia.org/wiki/File:Siberian_Tiger_by_Malene_Th.jpg