

L^AT_EX Kurs

Neue Befehle Teil 12

Sascha Frank

<http://www.latex-kurs.de/kurse/kurse.html>

Befehle

neue Befehle

- Bestehendes ändern

- Mit Optionen

Mathematik

Umgebungen

- Umgebungen ändern

Pakete

Befehle

eigene Befehle definieren

Mathematik

neue Bezeichner einführen

Umgebungen

eigene Umgebungen definieren

Pakete

eigene Pakete basteln

Grundsätzliches

- ▶ Keine bereits vorhandene Namen nutzen
- ▶ Alternative
 - ▶ deutsche Bezeichnung
 - ▶ Großgeschrieben
- ▶ Nur Buchstaben (aA – zZ) (und * am Ende)
- ▶ Ort (eigentlich) egal
 - ▶ **Ausnahme** BeamerClass vor Beginn des Dokumentes oder in jedem Frame in dem es verwendet wird.

eigene Befehle

Befehle

Neue Befehle definieren:

```
\newcommand{Name}{Definition}
```

Abkürzungen

```
\newcommand{\GT}{Spieltheorie}
```

\GT ist ein Teil der VWL

Spieltheorie ist ein Teil der VWL

geschachtelte Befehle

```
\newcommand{\nbs}{\nobreakspace}
```

```
\newcommand{\GTn}{Spieltheorie\nbs}
```

\GTn ist ein Teil der VWL

Spieltheorie ist ein Teil der VWL

mehr Möglichkeiten

Befehle

Neue Befehle mit zusätzlichen Argumenten definieren:

```
\newcommand{\Name}[Anzahl]{Definition}
```

Abkürzungen II

```
\newcommand{\GTB}[1]{\GT \ Blatt Nr.#1}
```

Eingabe

```
\GTB{2}
```

Ausgabe

Spieltheorie Blatt Nr.2

mehr Möglichkeiten

Befehle

Neue Befehle mit zusätzlichen Argumenten definieren:

```
\newcommand{\Name}[Anzahl]{Definition}
```

Abkürzungen II

```
\newcommand{\GTB}[1]{\GT \ Blatt Nr.#1}
```

Eingabe

```
\GTB{2}
```

Ausgabe

Spieltheorie Blatt Nr.2

Achtung!

Nur 9 Elemente möglich!

vorhandene Befehle ändern

Befehle

Die Wirkung eines Befehls ändern:

```
\renewcommand{\name}{neue Definition}
```


vorhandene Befehle ändern

Befehle

Die Wirkung eines Befehls ändern:

```
\renewcommand{\name}{neue Definition}
```

Achtung!

Überschreibt bestehende Befehle **ohne** Nachfrage!

vorhandene Befehle ändern

Befehle

Die Wirkung eines Befehls ändern:

```
\renewcommand{\name}{neue Definition}
```

Achtung!

Überschreibt bestehende Befehle **ohne** Nachfrage!

Beispiel

```
\newcommand{\stadt}{Freiburg ist eine sch\ "one Stadt!}  
\renewcommand{\stadt}{Moskau ist eine sch\ "one Stadt!}
```

vorhandene Befehle ändern

Befehle

Die Wirkung eines Befehls ändern:

```
\renewcommand{\name}{neue Definition}
```

Achtung!

Überschreibt bestehende Befehle **ohne** Nachfrage!

Beispiel

```
\newcommand{\stadt}{Freiburg ist eine sch\ "one Stadt!}  
\renewcommand{\stadt}{Moskau ist eine sch\ "one Stadt!}
```

Ausgabe

Freiburg ist eine schöne Stadt!

vorhandene Befehle ändern

Befehle

Die Wirkung eines Befehls ändern:

```
\renewcommand{\name}{neue Definition}
```

Achtung!

Überschreibt bestehende Befehle **ohne** Nachfrage!

Beispiel

```
\newcommand{\stadt}{Freiburg ist eine sch\ "one Stadt!}  
\renewcommand{\stadt}{Moskau ist eine sch\ "one Stadt!}
```

Ausgabe

Freiburg ist eine schöne Stadt!

Moskau ist eine schöne Stadt!

Optionen

Befehle

```
\newcommand{\Name}[Anzahl] [Default]{Definition}
```

Abkürzungen

```
\newcommand{\Studium}[1] [VWL]{Studienfach: #1}
```

```
\Studium von \dots bis \\\
```

```
\Studium[Info] von \dots bis \\\
```

```
\Studium[MST] von \dots bis \\\
```

Studienfach: VWL von ... bis

Studienfach: Info von ... bis

Studienfach: MST von ... bis

Wenn's mal mehr seien soll

```
\newcommand\fot [2]{#1}
```

```
\newcommand\sot [2]{#2}
```

```
\newcommand{\szmatrix}[8]{
```

```
\begin{vmatrix}
```

```
\fot#1 & \sot#1 & \fot#2 & \sot#2 \\
```

```
\fot#3 & \sot#3 & \fot#4 & \sot#4 \\
```

```
\fot#5 & \sot#5 & \fot#6 & \sot#6 \\
```

```
\fot#7 & \sot#7 & \fot#8 & \sot#8 \\
```

```
\end{vmatrix}
```

```
}
```

Wenn's mal mehr seien soll

```
\[ \szmatrix{{1}{2}}{{3}{4}}{{5}{6}}{{7}{8}}{{9}{10}}  
{{11}{12}}{{13}{14}}{{15}{16}} \]
```

$$\begin{vmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{vmatrix}$$

Neue Operatoren

einmalig

```
\operatorname{NeuerOperator}
```

mehr als einmal

```
\newcommand{\NeuOp}{\operatorname{NeuerOperator}}
```

besser

```
\DeclareMathOperator{\NeuOp}{NeuerOperator}
```

besser mit ...

```
\DeclareMathOperator*{\NeuOp}{NeuerOperator \, }
```

Umgebungen

Im Prinzip wie Befehle

```
\newenvironment{Name}{Anfang}{Ende}
```

Besitzt auch Parameter und optionales Element

```
\newenvironment{Name}[Parameter][Option]{Anfang}{Ende}
```

Anfang/Ende

Anfang und Ende Block sind Befehle.

Parameter

Parameter werden nur im Anfangsblock verwendet!

vorhandene Umgebungen ändern

Umgebung

Die Wirkung einer Umgebung ändern:

```
\renewenvironment{Name}[Anzahl]{Begin}{End}
```

vorhandene Umgebungen ändern

Umgebung

Die Wirkung einer Umgebung ändern:

```
\renewenvironment{Name}[Anzahl]{Begin}{End}
```

Achtung!

Überschreibt bestehende Umgebungen **ohne** Nachfrage!

vorhandene Umgebungen ändern

Umgebung

Die Wirkung einer Umgebung ändern:

```
\renewenvironment{Name}[Anzahl]{Begin}{End}
```

Achtung!

Überschreibt bestehende Umgebungen **ohne** Nachfrage!

Beispiel

```
\renewenvironment{center}{\begin{flushleft}}{\end{flushleft}}
```

vorhandene Umgebungen ändern

Umgebung

Die Wirkung einer Umgebung ändern:

```
\renewenvironment{Name}[Anzahl]{Begin}{End}
```

Achtung!

Überschreibt bestehende Umgebungen **ohne** Nachfrage!

Beispiel

```
\renewenvironment{center}{\begin{flushleft}}{\end{flushleft}}
```

Ausgabe

Die center-Umgebung zentriert nun nicht mehr, sondern macht jetzt linksbündigen Satz.

eigene Pakete

- ▶ Format `\NeedsTeXFormat{LaTeX2e}`
- ▶ Paketname `\ProvidesPackage{paketname}`
- ▶ benötigte Pakete `\RequirePackage{...,...}`
- ▶ ein Ende `\endinput`

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{MeinStil}
\RequirePackage[ngerman]{babel}
\newenvironment{MyQuote}
...
\renewenvironment{quote}{\begin{MyQuote}}{\end{MyQuote}}
\endinput
```

Aufgaben

Aufgabe 1: Erstellen Sie einen neuen mathematischen Operator.

Aufgabe 2: Erstellen Sie, ausgehend von der Standard quote Umgebung, eine neue MyQuote Umgebung.

```
\newenvironment{quote}
    {\list{}{\rightmargin\leftmargin}%
    \item\relax}
{\endlist}
```

Aufgabe 3: Erstellen Sie ein eigenes Usepackage (zum Beispiel MeinStil) in das Sie die neue Umgebung (MyQuote) einfügen und die bisherige Variante ersetzen. So dass Sie nach einfügen Ihres Paktes bei der Verwendung der quote Umgebung die neue Variante erhalten.