

Ab jetzt ist es möglich in Deutsch korrekte Auflistungen zu setzen:

```
1 \typesetList{Hallo Welt, na wie geht es dir?}
```

Ergibt: Hallo Welt und na wie geht es dir?

Warum? möchte man sich nun fragen. Nun, in dieser Konstellation ist es recht einfach einen Befehl zu Konstruiere der auf jedes Element angewendet werden kann. Der Inhalt wird als ein Token eingelesen und kann entsprechend weiterverarbeitet werden:

```
1 \def\tollercmd#1{%
2   \expandafter\Acronym{#1}%
3 }
4 \typesetList[tollercmd]{Hallo du,Welten sind super,Na Wie Gehts dir?}
```

Ergibt: **Hallo du, Welten sind super** und **Na Wie Gehts dir?**

```
$ echo Hallo
```

```
Hallo
```

```
echo na du? | ls :
```

```
jake.conf
ListPreview-OUT
ListPreview.tex
```

```
echo na du? | ls
\documentclass{latex}
print (5+23):
```

```
28
```

Konstruiere Liste Peter mit Separator Komma:

```
1 \constructList[,]{Peter}
```

Füge Elemente hinzu:

```
1 \pushList{Peter}{2}\pushList{Peter}{55}
2 \pushList{Peter}{3}\pushList{Peter}{26}
```

Peter enthält nun: 2,55,3,26,

```
1 \thePeter
```

Über eine Liste kann man iterieren:

```
1 \getPeter
2 \foreach \x in \lillyxlist {
```

```

3   X:\x
4 }

```

Ergebnis: X:2 X:55 X:3 X:26 X:

Konstruiere eine neue Liste 'Waffel':

```

1 \constructList[, ]{Waffel}

```

Füge Elemente hinzu:

```

1 \pushList{Waffel}{A}\pushList{Waffel}{\thePeter}
2 \pushList{Waffel}{X}\pushList{Waffel}{21}

```

Ausgabe von \theWaffel : A, 2,55,3,26,, X, 21, Weiter ergibt:

```

1 \getWaffel
2 \foreach \x in \lillyxlist {
3   => \x
4 }

```

Ergebnis: => A => 2,55,3,26, => X => 21 =>

Listen werden also vereinigt und nicht durch einzelne Elemente separiert. Wir können testen ob ein Element in einer Liste enthalten ist, mit:

```

1 \containsList{Waffel}{A}
2 \containsList{Waffel}{410021}

```

A: TRUE

410021: FALSE

!2!!55!!3!!26!