



Programmieren
lernen mit Python

Listen (3/3)

Kira Grammel, Nina Ihde, Sebastian Serth & Selina Reinhard
Hasso-Plattner-Institut
Universität Potsdam

Datentypen in Listen

```
1 # Strings
2 schloss = ["rot", "grün", "gelb"]
3 # Integer
4 code = [1, 1, 2, 3, 5, 8]
5 # vermischte Datentypen
6 schloss_mit_code = ["rot", 1, 8]
7 print(schloss_mit_code)
8 print(schloss_mit_code[0])
```

```
["rot", 1, 8]
"rot"
```



Datentypen

- Elemente einer Liste können verschiedene Datentypen besitzen

Verschachtelte Listen

```
1 schloss_mit_code = ["rot", [1, 8]]
2 print(schloss_mit_code[0][0])
3 print(schloss_mit_code[1][1])
```



r
8

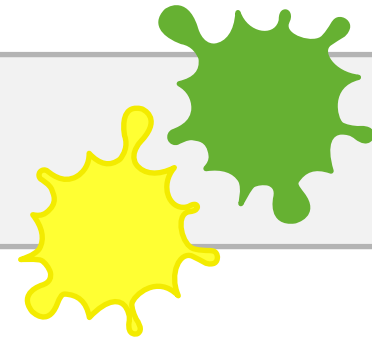
Verschachtelung

- Eine Liste kann eine andere Liste als Element haben
- Zugriff über die "verschachtelte Indexierung"

Entnehmen von Elementen

| | | | | | | |
|---|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 | 5 |
| 1 | codes = ["rot", [1, 8], "grün", [1, 5], "gelb", [2, 3]] | | | | | |
| 2 | print(codes[2:4]) # gibt Index 2 bis 3 aus | | | | | |
| 3 | print(codes[4:]) # gibt Index 4 bis 5 (letzter Index) aus | | | | | |

```
["grün", [1, 5]]
["gelb", [2, 3]]
```



"Slicing"

- bedeutet "aufschneiden"
- Liste[<Startindex> : <Endindex>], wobei das Element am **Startindex** eingeschlossen und am **Endindex** ausgeschlossen ist
- Nur wenn gewünschte Werte für **Startindex** und **Endindex** von den Standardwerten abweichen, müssen sie mit angegeben werden

Listen und Zeichenketten

```
1 satz = "Simon und Leonie sind auf Schnitzeljagd"  
2 geteilter_satz = satz.split()  
3 print(geteilter_satz)
```

```
["Simon", "und", "Leonie", "sind", "auf", "Schnitzeljagd"]
```

Split()

- Teilt eine Zeichenkette in einzelne Wörter auf
- Gibt eine Liste zurück

Listen und Zeichenketten

```
1 geteilter_satz = ["Simon", "und", "Leonie", "sind", "auf",  
  "Schnitzeljagd"]  
2 verbundener_satz = " ".join(geteilter_satz)  
3 print(verbundener_satz)
```

```
Simon und Leonie sind auf Schnitzeljagd
```

Join()

- Fügt Liste von Strings zu einem String zusammen
- Wird auf String aufgerufen, der Trennzeichen enthalten kann

Listen und Zeichenketten

```
1 geteilter_satz = ["Simon", "und", "Leonie", "sind", "auf",  
   "Schnitzeljagd"]  
2 verbundener_satz = "#".join(geteilter_satz)  
3 print(verbundener_satz)
```

```
Simon#und#Leonie#sind#auf#Schnitzeljagd
```

Join()

- Fügt Liste von Strings zu einem String zusammen
- Wird auf String aufgerufen, der Trennzeichen enthalten kann

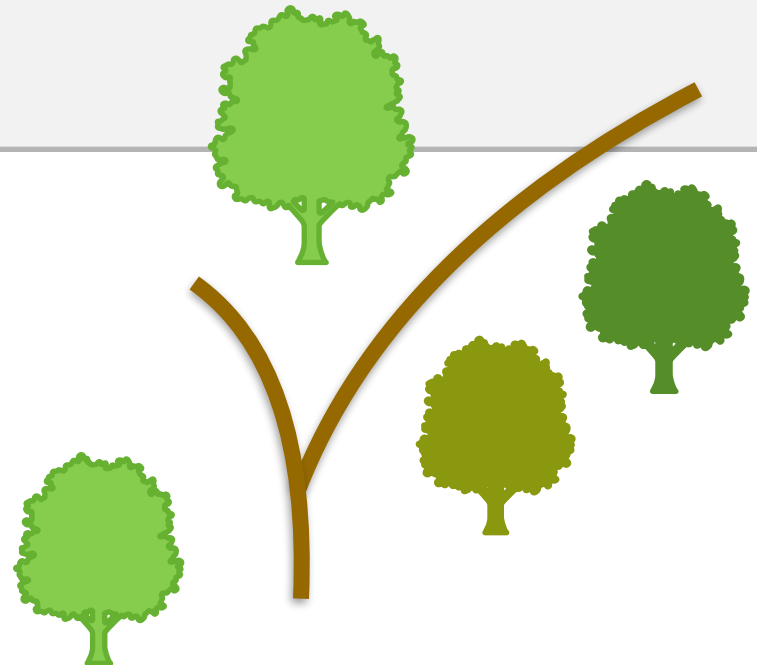
Iterieren über Listen

```
1 hinweise = ["Der", "Baum", "linke", "rechte", "Weg"]  
2 for wort in hinweise:  
3     print(wort)
```

```
Der  
Baum  
linke  
rechte  
Weg
```

For-Schleife

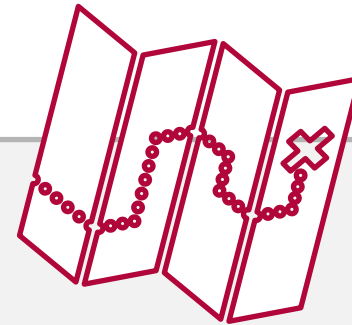
- Iteriert über Elemente der Liste
- `for <Variable> in <Liste>:`
 <Aktion mit Variable>



Rätsel lösen

```
1 hinweise = ["Der", "Baum", "linke", "rechte", "Weg"]
2 for wort in hinweise:
3     if wort[1] == "e":
4         print(wort)
```

Der
rechte
Weg



- Nur Ausgabe von Elementen, die ein "e" an zweiter Stelle besitzen
- Einrückung beachten!

Zusammenfassung

- Listen können enthalten:
 - unterschiedliche Datentypen
 - weitere Listen
- Elemente entnehmen:
 - `Liste[<Startindex> : <Endindex>]`
 - `Startindex` eingeschlossen, `Endindex` ausgeschlossen
- Liste → String: `join()`
- String → Liste: `split()`
- Iterieren über Listen z.B. mit der For-Schleife möglich

