Python Listen

pySpaceBremen

Agenda

- **Listen**
    - Deklarieren
    - Initialisieren
    - Einträge ändern
    - Slicing
- **Listen Methoden**
    - Objekte einfügen und anhängen
    - Einträge entfernen
    - Listen leeren und löschen
    - Einträge suchen
    - Listen kopieren
    - Listen sortieren
- **List Comprehensions**
    - Listen Werte transformieren
    - Listen Werte filtern
    - Listen Werte filtern und transformieren
    - List Comprehensions

Listen

Deklarieren

```
>>> liste = list()
>>> liste
[]
>>> type(liste)
<class 'list'>
>>> liste = []
>>> liste
[]
>>> type(liste)
<class 'list'>
```

Listen

Initialisieren

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> liste
[1, 2, 3, 4, 5]
>>> liste = list(range(1,6))
>>> liste
[1, 2, 3, 4, 5]
```

Listen

Einträge ändern

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> liste[2] = 42
>>> liste
[1, 2, 42, 4, 5]
>>> liste.remove(2)
>>> liste
[1, 42, 4, 5]
```

Listen

Slicing

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> liste[2:3]
[3]
>>> liste[:3]
[1, 2, 3]
>>> liste[-1:]
[5]
```

Listen Methoden

Objekte anhängen und einfügen

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> liste.append(23)
>>> liste
[1, 2, 3, 4, 5, 23]
>>> liste.insert(3, 42)
>>> liste
[1, 2, 3, 42, 4, 5, 23]
```

Listen Methoden

Einträge entfernen

```
>>> liste = [ 1, 2, 3, 42, 4, 5, 23]
>>> liste
[1, 2, 3, 42, 4, 5, 23]
>>> liste.pop(3)
42
>>> liste
[1, 2, 3, 4, 5, 23]
>>> liste.remove(23)

>>> liste
[1, 2, 3, 4, 5]
```

Listen Methoden

Liste leeren

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> liste
[1, 2, 3, 4, 5]
>>> liste.clear()
>>> liste
[]
```

Liste löschen

```
>>> del(liste)
>>> liste
Traceback (most recent call last):
  File "<input>", line 1, in <module>
NameError: name 'liste' is not defined
```

Listen Methoden

Listen kopieren

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> liste2 = liste
>>> liste[0] = 55
>>> liste
[55, 2, 3, 4, 5]
>>> liste2
[55, 2, 3, 4, 5]
>>> liste2 = liste.copy()
>>> liste[1] = 66
>>> liste
[55, 66, 3, 4, 5]
>>> liste2
[55, 2, 3, 4, 5]
```

Listen Methoden

Listen erweitern

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> liste2 = [11, 12, 13, 14, 15]
>>> liste3 = liste + liste2
>>> liste3
[1, 2, 3, 4, 5, 11, 12, 13, 14, 15]
>>> liste3.extend(range(21,26))
>>> liste3
[1, 2, 3, 4, 5, 11, 12, 13, 14, 15, 21, 22, 23, 24, 25]
```

Listen Methoden

Listen sortieren

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> liste.sort(reverse=True)
>>> liste
[5, 4, 3, 2, 1]
>>> liste.sort()
>>> liste
[1, 2, 3, 4, 5]
>>> liste.reverse()
>>> liste
[5, 4, 3, 2, 1]
```

List Comprehensions

Listen Werte transformieren

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> erg = []
>>> for wert in liste:
      erg.append(wert * 10)

>>> erg
[10, 20, 30, 40, 50]
>>> list(map(lambda x: x*10, liste)
[10, 20, 30, 40, 50]
```

List Comprehensions

Listen Werte filtern

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> erg = []
>>> for wert in liste:
      if wert % 2 == 0:
         erg.append(wert)

>>> erg
[2, 4]
>>> list(filter(lambda x: x%2==0, liste))
[2, 4]
```

List Comprehensions

Listen Werte filtern und transformieren

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> erg = []
>>> for wert in liste:
      if wert % 2 == 0:
         erg.append(wert * 10)

>>> erg
[20, 40]
>>> list(map(lambda x: x*10,
                  filter(lambda x: x%2==0, liste)))
[20, 40]
```

List Comprehensions

```
>>> liste = [ 1, 2, 3, 4, 5]
>>> [wert*10 for wert in liste]
[10, 20, 30, 40, 50]
>>> [wert*10 for wert in liste if wert % 2 == 0]
[20, 40]
```