

Python Aufgaben zu Listen

Torben Friedrich Görner

November 2022

1 Listen

Listen werden verwendet, um mehrere Elemente in einer einzigen Variablen zu speichern. Listenelemente sind indiziert, das erste Element hat den Index [0], das zweite Element den Index [1] usw.

```
fruits = ["apple", "banana", "cherry", "apple", "cherry"]
apple = fruits[0] # holt das erste Element aus 'fruits'
```

1.1 Listen anlegen und Elemente hinzufügen

Die Leere Liste könnt ihr wie folgt anlegen, sie enthält keine Elemente und hat eine Länge von 0.

```
fruits = []
```

Um Elemente hinzuzufügen könnt ihr `append()` nutzen.

```
fruits = ["apple", "banana", "cherry"]
fruits.append("orange")
```

1.2 Über Listen iterieren (Listen durchlaufen)

Die Länge einer Liste erhaltet ihr mit `len()`.

```
fruits = ["apple", "banana", "cherry"]
print(len(fruits))
```

Es gibt verschiedene Möglichkeiten Listen zu durchlaufen. Ein Beispiel ist zählen mit einem Schleifenzähler.

```
fruits = ["apple", "banana", "cherry"]
for i in range(len(fruits)):
    print(fruits[i])
```

Eine andere (elegantere) Möglichkeit sieht so aus.

```
fruits = ["apple", "banana", "cherry"]
for fruit in fruits:
    print(fruit)
```

2 Aufgabe 1

Schreibe ein Programm welches alle Zahlen von 0 - 100 die durch 3 teilbar sind in einer Liste speichert. Um zu prüfen ob eine Zahl durch 3 teilbar ist könnt ihr den Modulo-Operator nutzen. Ist eine Zahl Modulo 3 == 0 (Bsp. $9 \% 3 == 0$), so ist die Zahl durch 3 teilbar.

Modulo Wikipedia: https://de.wikipedia.org/wiki/Division_mit_Rest

3 Aufgabe 2

Gegeben sei eine Liste l mit Integern. Schreibe ein Programm, welches das größte Element aus der Liste l holt.

```
l = [0,5,2,42,8,30]
```

4 Aufgabe 3

Geben sei eine Liste l mit Integern. Schreibe ein Programm, welches alle ungeraden Elemente entfernt. Ein Tipp: Der Modulo Operator kann helfen zu erkennen ob eine Zahl gerade oder ungerade ist. Ein Beispiel für das Entfernen eines Elementes siehst du hier.

```
fruits = ["apple", "banana", "cherry"]
fruits.remove("banana")
```

Alternativ geht es auch so...

```
fruits = ["apple", "banana", "cherry"]
del fruits[0]
```