

while- und for-Schleifen

2.1 Wiederholungen

Du hast die Aufgabe ein Programm zu schreiben, das die Zahlen von 1 bis 100 ausgibt. Wie könnte man diese Aufgabe lösen? Einfach hintereinander hinschreiben:

```
1 print(1)
2 print(2)
3 # ... 96 Zeilen mehr
4 print(99)
5 print(100)
```

Diese Methode ist umständlich und sehr aufwendig. Glücklicherweise gibt es dafür eine schlaue Lösung.

2.2 Die for-Schleife

Die `for`-Schleife dient dazu Elemente, z.B. die einer Liste einzeln durchzugehen und aufzulisten.

```
1 for zahl in range(1, 100):
2     print(zahl)
```

`range(1, 100)` (engl. für Bereich) erzeugt eine Liste mit allen Zahlen vom Startwert 1 bis zum Endwert 100 (1, 2, ..., 98, 99). Diese werden dann nacheinander in die Variable `zahl` geschrieben und daraufhin werden jeweils alle eingerückten Zeilen der `for`-Schleife ausgeführt.

Der obige Code gibt also alle Zahlen von 1 bis 100 (1, 2, ..., 98, 99) aus!

```
1 for zahl in range(1, 100, 2):
2     print(zahl)
```

`range(1, 100, 2)` erzeugt eine Liste mit allen Zahlen zwischen 1 und 100 in 2-er Schritten. Also 1, 3, ..., 97, 99. Anstelle von 2 kann jede beliebige andere Zahl verwendet werden.

for-Schleife mit `range`

Für einfache Schleifen gilt der Syntax:

```
for <variable> in range(<Startwert>, <Endwert>, <Schritt>):
    # Anweisungen für jeden Durchlauf
```

Hinweis: Die Verwendung von `range(<Startwert>, <Endwert>)` entspricht `range(<Startwert>, <Endwert>, 1)`.

2.3 Aufgaben

- Schreibe eine `for`-Schleife, die alle Zahlen zwischen 40 und 100 in 3-er Schritten ausgibt
- Experimentiere mit den Werten von `range`. Was passiert, wenn der Startwert größer ist als der Endwert? Was passiert, wenn der Schritt (z.B. die 2 im obigen Beispiel) größer ist als der Endwert. Was passiert, wenn der Schritt negativ ist?
- Schreibe eine `for`-Schleife, die alle Zahlen von 100 bis einschließlich 1 ausgibt. Also 100, 99, ..., 2, 1.

2.4 Die while-Schleife

Zusätzlich zur `for`-Schleife gibt es die `while`-Schleife. Die `while`-Schleife kann verwendet werden, um Anweisungen wiederholt auszuführen, bis eine Bedingung erfüllt ist.

```
1 zaehler = 1
2 while zaehler <= 100:
3     print(zaehler)
4     zaehler = zaehler +1
```

Im obigen Beispiel wiederholt die `while`-Schleife den eingerückten Code solange die Bedingung `1 <= 100` erfüllt ist. Dieses Beispiel gibt somit die Zahlen von 1 bis einschließlich 100 aus.

while -Schleife

Eine `while`-Schleife hat den Syntax:

```
while <Bedingung>:
    # Anweisungen, die wiederholt werden
    # solange die Bedingung erfüllt ist
```

Tipp: Erklärungen und Beispiele zu Bedingungen findest du auf dem Blatt zu `if`-Abfragen.

2.5 Aufgaben

- Schreibe eine `while`-Schleife, die die Zahlen von 100 bis 0 in 2er Schritten ausgibt.
- Was tut folgendes Programm?

```
1 wert = 1
2 while wert < 10:
3     wert = wert + wert
4     print(wert)
```

Versuche die Aufgabe mit Stift und Papier zu lösen, indem Du die Werte einzeln ausrechnest. Schreibe das Programm nicht ab und führe es aus! Erst wenn Du eine Lösung hast, kannst du das Programm abtippen um deine Lösung zu überprüfen.