

SOURCE CODE 1. Beispiel: Einfache Schleife (englisch: loop) über eine Liste.

```
# Listen (auch Arrays oder Vektoren genannt) gibt es in den meisten
# Programmiersprachen; sie werden über einen INDEX (also eine "Nummer")
# adressiert (wobei die Numerierung meist bei 0 beginnt); also z.B. so:
eine_liste = [1,2,3,4,5]
# Diese _Indizierung_ (also die Identifizierung eines Listenelements
# durch seinen Index ergibt _implizit_ (also ohne dass man das ausdrücklich
# festlegen muss) eine ORDNUNG der Listenelemente:
for i in range(5):
    print(f'Das {i}-te Listenelement (mit Index {i}) ist {eine_liste[i]}'.)
# In Python kann man auf Listenelemente aber auch
# "direkter" (ohne Indizierung) zugreifen:
for element in eine_liste:
    print(f'{element} ist in der Liste')
# Das erste Beispiel, indem auch immer der Index des Listenelements
# ausgegeben wird, könnte man so implementieren:
for i,element in enumerate(eine_liste):
    print(f'Das {i}-te Listenelement (mit Index {i}) ist {element}'.)
# (Das schaut jetzt vielleicht nicht sehr beeindruckend aus:
# Die durchdachten syntaktischen "Tricks", die Python insbesondere für
# die Verarbeitung von Listen bietet, lernt man aber schnell zu schätzen!)
```