

Programmierung mit Python

Übung 6: Funktionen

0. Aufgabe: Code-Verständnis

Was geben die folgenden Programm-Zeilen auf die Console aus?

```
def aFun1(a)-> int:
                                                 def aFun2(a)-> str:
  if isinstance(a, list) or isinstance(a, str):
                                                   if isinstance(a, list) or isinstance(a, set):
     res = len(a * 2)
                                                      res = str(a)
  elif isinstance(a, set):
                                                   elif isinstance(a, str):
    a.add(0)
                                                      res = a.replace("b", "Q")
    res = len(a)
                                                   else:
  else:
                                                      res = a + 1
    res = a + 1
                                                   return str(res)
  return res
                                                 print(aFun2([1,2,3]))
print(aFun1([1,2,3]))
                                                 print(aFun2("abcb"))
print(aFun1("abcb"))
                                                 print(aFun2({2,3,1,2}))
print(aFun1({2,3,1,2}))
                                                 print(aFun2(17))
print(aFun1(17))
```

1. Aufgabe: Rabattrechner

Schreiben Sie eine Funktion rabattRechner(preis, rabatt), welches aus dem Normalpreis einer Ware und dem Rabatt den Endpreis berechnet. Falls der Wert für Preis oder Rabatt negativ sind, soll eine Fehlermeldung ausgegeben werden. Der Rückgabewert der Funktion soll dann None sein.

2. Aufgabe: Fakultät

Schreiben Sie eine Funktion fakultät(n), welche die Fakultätsfunktion von n berechnet. Diese ist definiert als Produkt aller Zahlen kleiner gleich n, also n! = 1 * 2 * 3 * 4 * ... * n

3. Aufgabe: Funktion zum Listen erzeugen

Schreiben Sie eine Funktion, erzeugeListe(a), welche als Resultat die Liste mit den ersten 10 Vielfachen von a erzeugt:

```
erzeugeListe(1) -> [0, 1, 2, ..., 9]
erzeugeListe(5) -> [0, 5, 10, 15, ..., 45]
```

4. Aufgabe: Funktion zum Elemente aufsummieren

Schreiben Sie eine Funktion summiere(liste), welches mit Hilfe eines for-Konstrukts alle Elemente einer Liste aufsummiert.

```
summiere([0, 1, 2, ..., 9]) -> 45
```

5. Aufgabe: Funktion innerhalb

Schreiben Sie eine Funktion innerhalb(a, b, n), welche True zurück gibt, falls a <= n <= b innerhalb (5, 9, 2) -> False innerhalb (5, 9, 7) -> True

6. Aufgabe: Funktion innerOderAusserhalb

Schreiben Sie eine Funktion innerOderAusserhalb(a, b, n, w), welche w zurück gibt, falls a <= n und n <= b, sonst not(w)

innerOderAusserhalb (5, 9, 2, True) -> False innerOderAusserhalb (5, 9, 2, False) -> True innerOderAusserhalb (5, 9, 7, True) -> True innerOderAusserhalb (5, 9, 7, False) -> False

7. Aufgabe: Listen zusammenhängen und sortieren

Schreiben Sie eine Funktion zusammen fügen(liste1, liste2), welche die beiden Listen zusammenfügt und dann sortiert. Die sortierte Liste wird als Resultat zurückgegeben.

8. Aufgabe: Listen vergleichen

Schreiben Sie eine Funktion gemeinsame Elemente(liste1, liste2), welches die gemeinsamen Elemente von liste1 und liste2 zurück gibt.

Module

9. Aufgabe: Modul erzeugen

Schreiben Sie ein Modul für die erzeugten Funktionen. Schreiben Sie dann ein separates Python-File, welches das Modul importiert und dann benutzt, um die verschiedenen Funktionen zu testen.