Berner Fachhochschule

Programmierung mit Python

Repetitionsaufgaben

1. Aufgabe: Datentypen

Geben Sie die Datentypen der Variablen a, b, c, d und e an:

a = -7, b = False, c = str(3.1), d = "Hallo", e = Adresse("Bahnhofstrasse", 2)

2. Aufgabe: Fallunterscheidung

Was ist die Ausgabe auf die folgenden Aufrufe

```
def aufg2(a, b):
    if a < 3 or a > 8 :
        print(str(a))
    elif len(b) > 1:
        print(b[0])
    else:
        print("0")
```

a) `aufg2(1,[2,3])`
b) `aufg2(3,[4,5])`
c) `aufg2(4,[2])`
d) `aufg2(8,[4])`

3. Aufgabe: Range

Was ist die Ausgabe der folgenden Aufrufe

```
def aufg3(a, b, c):
    liste = list()
    for ele in range(a,b,c):
        liste.append(ele)
    return liste
```

a) `print(aufg3(0,3,2))`
b) `print(aufg3(0,10,3))`
c) `print(aufg3(1,11,5))`

4. Aufgabe: Schleife

Was ist die Ausgabe der folgenden Aufrufe:

```
def aufg4(liste):
    resultat = ""
    for ele in liste:
        resultat = resultat + str(ele) +" "
    return resultat
```

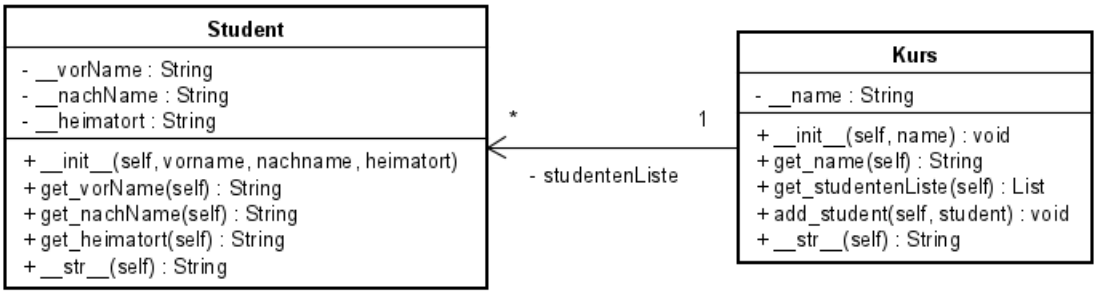
a)
```
liste1 = [1, 5, 17]
print(aufg4(liste1))
```
b)
```
liste2 = ['a', 'b']
print(aufg4(liste2))
```
c)
```
liste3 = liste1 + liste2
print(aufg4(liste3))
```

5. Aufgabe: Exception

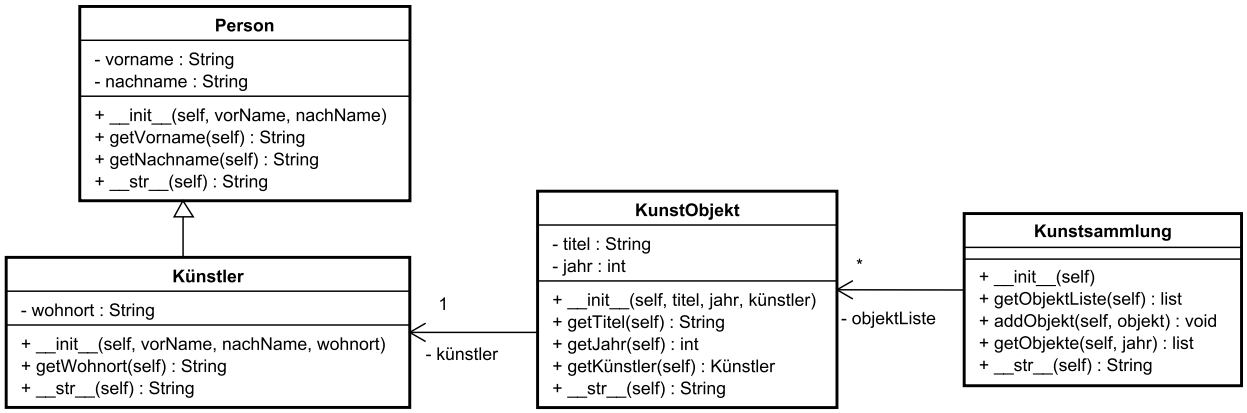
Ergänzen Sie die Funktion der Aufgabe 4, so dass diese nicht abstürzt, falls *liste* keine Liste ist. Sie soll dann eine Fehlermeldung ausgeben.

6. Aufgabe: Student, Kurs

a) Schreiben Sie eine Klasse **`Student`** mit seinem `Vornamen`, `Nachnamen und Heimatort`, sowie dessen Methoden `init`, den drei get-Methoden und der str-Methode.



b) Schreiben Sie eine Klasse **Kurs** gemäss dem Klassendiagramm.
c) Erzeugen Sie einen Kurs und fügen Sie drei erfundene Studenten ein.
d) Schreiben Sie die Funktion str so, dass der Name und die Anzahl Studenten des Kurses ausgegeben werden.
e) Ergänzen Sie die Klasse Kurs um eine Funktion save(), welcher den Kurs-Namen und die Studenten-Liste in ein File kurs.txt abspeichert.

7.Kunst

Implementieren Sie die folgenden Klassen gemäss Klassendiagramm.