

Einführung in die Programmierung

15.01.2025

1 Umgang mit Variablen und Datentypen

In dieser Aufgabe sollen Sie ein Gefühl dafür bekommen, wie Python mit Werten und Datentypen umgeht, und wie Sie Variablen für komplexere Ausdrücke verwenden können.

a) Definieren Sie die folgenden Variablen:

- `a` mit dem Wert 10
- `b` mit dem Wert 20
- `c` mit der Summe von `a` und `b`
- `d` mit dem Wert 4,2
- `e` mit dem Produkt von `c` und `d`
- `f` mit dem Quadrat von `b`
- `g` mit Division von `f` und `e`
- `h` mit Ganzzahldivision von `f` und `e`
- `i` mit dem Rest der Ganzzahldivision von `f` und `e`

Lassen Sie sich nach jeder Variablendefinition den Wert der gerade definierten Variable ausgeben (`print`).

```
a = ...  
print(a)  
b = ...  
print(b)  
...
```

b) Definieren Sie eine Variable `s` mit dem Wert `"Hallo Welt!"` und lassen Sie sich diese mit `print` ausgeben.

Erweitern Sie den String `s` um zwei weitere Ausrufezeichen `"!!"` und lassen Sie sich das Ergebnis wieder ausgeben.

c) Re-Implementieren Sie in einem Ausdruck den Operator `//`, der in Python eine Ganzzahldivision `dividend // divisor` durchführt. Greifen Sie dafür jedoch auf andere Operatoren von Python zurück, insbesondere `/` (Division) und `%` (Modulo) sowie falls notwendig Subtraktion und Addition. Testen Sie Ihre Implementierung in Python.

```
dividend = 111  
divisor = 23  
  
quotient = #CODE HIER  
print("Der Ganzzahl-Quotient von ", dividend, "//", divisor, " ist ",  
      quotient)
```

- d) Gegeben sei ein Dreieck mit den Seitenlängen a , b und c . Implementieren Sie den Satz des Heron zur Berechnung der Fläche des Dreiecks $A = \sqrt{s * (s - a) * (s - b) * (s - c)}$ mit $s = \frac{a+b+c}{2}$. Die drei Seitenlängen werden in Ihrem Programm nacheinander über die Kommandozeile eingelesen (`input()`-Funktion). Stellen Sie sicher, dass die eingelesenen Variablen keinen negativen Wert, sondern größer als 0 sind und ein gültiges Dreieck formen können. Falls die Eingaben diesen Bedingungen nicht entsprechen, brechen Sie mit einer Fehlermeldung ab. Ansonsten geben Sie am Ende das Ergebnis der Berechnung im Terminal aus.

```
#Verwenden Sie die Funktion sqrt(x) zur Berechnung der Quadratwurzel
from math import sqrt;

# Kantenwerte einlesen
a = int(input("Geben Sie Seitenlänge a an:"))

# ... sicherstellen, dass a/b/c größer als 0 sind

s = #... s berechnen
area = #... Fläche berechnen

#... Ergebnis ausgeben
```

- e) Im Folgenden sind mehrere mehr oder minder korrekte Wertzuweisungen zu unterschiedlichen Variablen gegeben. Versuchen Sie zunächst ohne die Verwendung des Python-Interpreters die Gültigkeit der folgenden Ausdrücke zu bestimmen. Ist eine Wertzuweisung Ihrer Meinung nach gültig, dann bestimmen und notieren Sie für jeden Ausdruck den resultierenden Datentyp und Wert. Beschreiben Sie andernfalls, welche(r) Fehler bei der jeweiligen Zuweisung unterlaufen sind/ist. Die Auswertung jeder Zeile steht dabei für sich.

```
> #Beispiel: z=14; gültig, Typ int mit Wert 14
> a = "12"
> var_b = true
> c = 'python'
> d = 5
> 2e = 7
> f = 5; _g = f*"ha"
> hi2 = 3.1415
> ieeh = 'Pfui' + 8 + "Spinne"
> j-2 = 17+2
> kalamari = not False and not True
> l = True and 2
> m = 3,1415
> n = False or 2 and 5
> o = False and 2 or 5
> p = 25 == 5**2
> q = 25.0/5
> r = 26//5.0
> s = 26-(26//5*5)
> t = 26%5
```

2 Übungen zur Stringformatierung

In Python wie auch in vielen anderen Programmiersprachen ist Stringformatierung eine in der Praxis sehr häufig eingesetzte Funktionalität zur Darstellung von Werten unterschiedlichen Typs in einem gewünschten Format. Komplizierte Vorverarbeitung (z.B. Umrechnung von Integer-Darstellungen) können so vermieden werden.

Als Hilfestellung können Sie die Python-Dokumentation heranziehen:

<https://docs.python.org/3/library/string.html>

- a) Gegeben sei ein Dezimalzahl-Wert 159_{10} . Geben Sie durch Stringformatierung den Wert in folgenden Varianten als String zurück (mittels der Funktion `format()`, die auf einem String-Objekt aufgerufen werden kann) und schreiben sie ihn auf die Konsole aus. Die erste Aufgabe stellt ein Beispiel dar.

- BEISPIEL: Der Wert als Hexadezimalzahl.

```
print("In Hex ist die Zahl {:x}".format(159))
```

- Der Wert als Hexadezimalzahl, aufgefüllt mit führenden Nullen zu acht Zeichen.
- Der Wert als Float.
- Der Wert als Float mit drei Nachkommastellen.
- Der Wert als Float mit drei Nachkommastellen und fünf Vorkommastellen.
- Den Wert als Integer-Dezimalzahl mit Vorzeichen und einem Padding auf 9 Zeichen zwischen Vorzeichen und Wert.

- b) Gegeben ist der String `"Python"` und die Dezimalzahl 3. Nehmen Sie die folgenden Stringformatierungen vor.

- Konkatenation der beiden Werte in der gegebenen Reihenfolge.
- Konkatenation mit Padding auf 15 Zeichen, zentral ausgerichtet. (Benötigt zwei `format`-Aufrufe)
- Den String `"Python"` links ausgerichtet mit einem Padding auf 10 Zeichen und nachfolgender Dezimalzahl 3.