

Tupel und Dictionaries - Übung

Richard Müller, Tom Felber

11. November 2021

Python-Kurs

1. Tupel

2. Dictionaries

Tupel

Übung 1

1. Schreibe eine Funktion, die zwei 2-dimensionale Punkte A und B (jeweils als Tupel) entgegennimmt und den Vektor AB als Tupel zurückgibt.

Bsp.: $A = (3, 5)$, $B = (-1, -2)$, Ergebnis: $(-4, -7)$

2. Modifiziere die Funktion so, dass sie nun Vektoren beliebiger Dimension berechnen kann. Der Einfachheit halber gehen wir davon aus, dass beide Punkte die selbe Dimension haben.

Dictionaries

Übung 2

1. Es ist eine Liste von Variablen verschiedener Typen gegeben.
Schreibe eine Funktion, die ein Dictionary zurück gibt, in dem die Variablen als Keys verwendet werden und ihr jeweiliger Typ als Value.

Bsp.: `[1, 'hallo']` Ergebnis: `{1: 'int', 'hallo': 'str'}`

2. Definiere eine zweite Funktion, die stattdessen den Typ einer Variable als Key verwendet. Alle Variablen eines Typs sollen, in einer Liste als Value für den jeweiligen Key, abgespeichert werden.

Bsp.: `[1, 2, 4, 'hallo', 'ok']` Ergebnis: `{'int': [1, 2, 4], 'str': ['hallo', 'ok']}`

Übung 3

1. Gegeben ist ein Dictionary. Schreibe eine Funktion, welche zählt, wie oft jeder Value vorkommt und ein Dictionary ausgibt, indem der jeweilige Value auf die Anzahl seines Vorkommens gemappt wird.

Bsp.: {'hallo': 'eins', 'x': 'eins', 'hi': 'ok'} **Ergebnis:**
{ 'eins': 2, 'ok': 1 }

2. Benutze die in 1. definierte Funktion, um von einem gegebenen Dictionary den Value zu bestimmen, der am häufigsten vorkommt.

Bsp.: { 'eins': 2, 'ok': 1 } **Ergebnis:** 'eins'

Übung 3

1. Gegeben ist ein Dictionary. Schreibe eine Funktion, welche zählt, wie oft jeder Value vorkommt und ein Dictionary ausgibt, indem der jeweilige Value auf die Anzahl seines Vorkommens gemappt wird.

Bsp.: `{'hallo': 'eins', 'x': 'eins', 'hi': 'ok'}` **Ergebnis:**
`{'eins': 2, 'ok': 1}`

2. Benutze die in 1. definierte Funktion, um von einem gegebenen Dictionary den Value zu bestimmen, der am häufigsten vorkommt.

Bsp.: `{'eins': 2, 'ok': 1}` **Ergebnis:** `'eins'`

- Z** Schreibe eine Funktion, die eine Liste von 2-dimensionalen Integer Tupeln entgegen nimmt. Die Tupelelemente sollen addiert werden und in einer Ergebnisliste ausgegeben werden. Da es vorkommen kann, dass das gleiche Tupel in der Liste zweimal vorkommt, soll ein Dictionary benutzt werden, um berechnete Werte zu speichern, sodass diese nicht erneut berechnet werden müssen.

Bsp.: `[(65, 7), (4, 5), (65, 7)]` **Ergebnis:** `[3, 72, 9, 72]`