

03b_Miniuebungen_Loesungen

0.1 Mini-Aufgaben zur Überprüfung des Verständnis: Dictionaries

Erstellen Sie aus der folgenden Liste ein Dictionary, ausgehend von einem leeren Dictionary. Hierbei ist der erste Wert im Tupel stets der Schlüssel und der zweite Wert der zugehörige Wert für das Dictionary. Die Lösung ist auf viele verschiedene Arten möglich.

Hinweis: Die beiden Elemente eines Zweiertupels erhalten Sie mit [0] und [1].

```
[4]: my_list = [('Hans', 23), ('Petra', 42), ('Georg', 91), ('Klaus', 12), ('Lea', 44)]
```

```
# Lösung per for Schleife
```

```
dict1 = {}
```

```
for t in my_list:
```

```
    dict1[t[0]] = t[1]
```

```
print(dict1)
```

```
# Lösung per for Schleife mit automatischem Entpacken
```

```
dict2 = {}
```

```
for key, val in my_list:
```

```
    dict2[key] = val
```

```
print(dict2)
```

```
# Lösung mit zip(), und zwei list comprehensions
```

```
keys = [ key for key, val in my_list ]
```

```
values = [ val for key, val in my_list ]
```

```
dict3 = dict(zip(keys, values))
```

```
print(dict3)
```

```
# und natürlich noch 200 Möglichkeiten als Mischungen,
```

```
# und mit while oder for Schleifen und Indizes
```

```
{'Hans': 23, 'Petra': 42, 'Georg': 91, 'Klaus': 12, 'Lea': 44}
```

```
{'Hans': 23, 'Petra': 42, 'Georg': 91, 'Klaus': 12, 'Lea': 44}
```

```
{'Hans': 23, 'Petra': 42, 'Georg': 91, 'Klaus': 12, 'Lea': 44}
```

Überprüfen Sie mit einem passenden Befehl, ob der Dictionary alle Einträge enthält. Fügen Sie dann sich selbst dem Dictionary hinzu, und löschen Sie dafür den Eintrag von Petra.

```
[6]: # natürlich tut es auch der print() Befehl. Schöner ist das hier,
# funktioniert aber nur, wenn der Code vorher bereits ausgeführt wurde

if len(dict1.keys()) == len(my_list):
    print("hat geklappt")
else:
    print("Dackelkacke")

del dict1['Petra']
dict1['Dominik'] = 22
print(dict1)
```

```
hat geklappt
{'Hans': 23, 'Georg': 91, 'Klaus': 12, 'Lea': 44, 'Dominik': 22}
```

Die Klausur zur Veranstaltung "Angewandte Textilwissenschaften 4" ist folgendermaßen ausgefallen: * 1.0: Felix Faden, Sarah Spindel * 1.7: Frederik Flicken * 2.0: Stefanie Schiffchen * 2.3: Nadine Nähkästchen, Willi Wolle * 3.7: Zidane Zwirn * 4.0: Willhelm Webstuhl * 5.0: Simon Straßenkehrer, Max Müllmann, Fritz Fleischer, Klara Kellnerin

Legen Sie diesen Notenspiegel in einem geeigneten Dictionary ab, wobei den Noten die Rolle der Keys zukommt. Erstellen Sie dann ein Histogramm des Notenspiegels, in einem zweiten Dictionary. Ermitteln Sie schließlich die Durchschnittsnote, nur mit Hilfe von Befehlen die von den beiden Dictionaries zur Verfügung gestellt werden.

```
[15]: results = {
    'Felix Faden' : 1.0,
    'Sarah Spindel' : 1.0,
    'Frederik Flicken' : 1.7,
    'Stefanie Schiffchen' : 2.0,
    'Nadine Nähkästchen' : 2.3,
    'Willi Wolle' : 2.3,
    'Zidane Zwirn' : 3.7,
    'Willhelm Webstuhl' : 4.0,
    'Simon Straßenkehrer' : 5.0,
    'Max Müllmann' : 5.0,
    'Fritz Fleischer' : 5.0,
    'Klara Kellnerin' : 5.0
}
#print(results)

# beinahe wörtliches Abschreiben des pi-Histogramms
counts = {}
for r in list(results.values()):
    if r not in counts:
        counts[r] = 0
    counts[r] += 1
#print(counts)
```

```
# und der Notendurchschnitt
average = 0
for c in counts:
    average += c*counts[c]
average /= len(results)

print(average)
```

3.1666666666666665

0.2 Impressum

0.2.1 Programmierkurs Python, Dominik Göddeke <https://www.ians.uni-stuttgart.de>,
Universität Stuttgart

Version vom April 2023

Lizenziert unter der Creative Commons Namensnennung 4.0 International Lizenz



Veröffentlicht auf <https://zoerr.de>, (alle Rechte am Logo vorbehalten)



Gefördert durch die Stiftung Innovation in der Hochschullehre. (alle Rechte am Logo vorbehalten)



Gefördert mit Mitteln der Deutschen Forschungsgemeinschaft (EXC 2075 - 390740016) im Rahmen der Exzellenzstrategie.

[]: