**Python Dictionary Exercise with Solutions**

It covers questions on the following topics:

* Dictionary operations and manipulations
* Dictionary functions
* Dictionary comprehension
* Python dictionary is a mutable object, and it contains the data in the form of key-value pairs. Each key is separated from its value by a colon (:).
* Dictionary is the most widely used data structure, and it is necessary to understand its methods and operations.

### **Exercise 1: Convert two lists into a dictionary**

Below are the two [lists](https://pynative.com/python-lists/). Write a Python program to convert them into a dictionary in a way that item from list1 is the key and item from list2 is the value

keys = ['Ten', 'Twenty', 'Thirty']

values = [10, 20, 30]

**Expected output:**

{'Ten': 10, 'Twenty': 20, 'Thirty': 30}

**Solution**1: The zip() function and a dict() constructor

* Use the zip(keys, values) to aggregate two lists.
* Wrap the result of a zip() function into a dict() constructor.

keys = ['Ten', 'Twenty', 'Thirty']  
values = [10, 20, 30]  
  
res\_dict = dict(zip(keys, values))  
print(res\_dict)

**Solution 2**: Using a loop and update() method of a dictionary

keys = ['Ten', 'Twenty', 'Thirty']  
values = [10, 20, 30]  
  
*# empty dictionary*res\_dict = dict()  
  
for i in range(len(keys)):  
 res\_dict.update({keys[i]: values[i]})  
print(res\_dict)

### **Exercise 2: Merge two Python dictionaries into one**

dict1 = {'Ten': 10, 'Twenty': 20, 'Thirty': 30}

dict2 = {'Thirty': 30, 'Fourty': 40, 'Fifty': 50}

**Expected output:**

{'Ten': 10, 'Twenty': 20, 'Thirty': 30, 'Fourty': 40, 'Fifty': 50}

dict1 = {'Ten': 10, 'Twenty': 20, 'Thirty': 30}  
dict2 = {'Thirty': 30, 'Fourty': 40, 'Fifty': 50}  
  
dict3 = {\*\*dict1, \*\*dict2}  
print(dict3)

### **Exercise 3: Print the value of key ‘history’ from the below dict**

sampleDict = {

"class": {

"student": {

"name": "Mike",

"marks": {

"physics": 70,

"history": 80

}

}

}

}

**Expected output:**

80

sampleDict = {  
 "class": {  
 "student": {  
 "name": "Mike",  
 "marks": {  
 "physics": 70,  
 "history": 80  
 }  
 }  
 }  
}  
  
print(sampleDict['class']['student']['marks']['history'])

### **Exercise 4: Initialize dictionary with default values**

In Python, we can initialize the keys with the same values.

**Given**:

employees = ['Kelly', 'Emma']

defaults = {"designation": 'Developer', "salary": 8000}

**Expected output:**

{'Kelly': {'designation': 'Developer', 'salary': 8000}, 'Emma': {'designation': 'Developer', 'salary': 8000}}

The fromkeys() method returns a dictionary with the specified keys and the specified value.

employees = ['Kelly', 'Emma']  
defaults = {"designation": 'Developer', "salary": 8000}  
  
res = dict.fromkeys(employees, defaults)  
print(res)  
  
*# Individual data*print(res["Kelly"])

### **Exercise 5: Create a dictionary by extracting the keys from a given dictionary**

Write a Python program to create a new dictionary by extracting the mentioned keys from the below dictionary.

**Given dictionary**:

sample\_dict = {

"name": "Kelly",

"age": 25,

"salary": 8000,

"city": "New york"}

# Keys to extract

keys = ["name", "salary"]

**Expected output:**

{'name': 'Kelly', 'salary': 8000}

**Solution 1**: Dictionary Comprehension

sampleDict = {  
 "name": "Kelly",  
 "age":25,  
 "salary": 8000,  
 "city": "New york" }  
  
keys = ["name", "salary"]  
  
newDict = {k: sampleDict[k] for k in keys}  
print(newDict)

**Solution 2**: Using the update() method and loop

sample\_dict = {  
 "name": "Kelly",  
 "age": 25,  
 "salary": 8000,  
 "city": "New york"}  
  
*# keys to extract*keys = ["name", "salary"]  
  
*# new dict*res = dict()  
  
for k in keys:  
 *# add current key with its value from sample\_dict* res.update({k: sample\_dict[k]})  
print(res)

**Exercise 6: Delete a list of keys from a dictionary**

**Given**:

sample\_dict = {

"name": "Kelly",

"age": 25,

"salary": 8000,

"city": "New york"

}

# Keys to remove

keys = ["name", "salary"]

**Expected output:**

{'city': 'New york', 'age': 25}

### **Solution 1**: Using the pop() method and loop

sample\_dict = {  
 "name": "Kelly",  
 "age": 25,  
 "salary": 8000,  
 "city": "New york"  
}  
*# Keys to remove*keys = ["name", "salary"]  
  
for k in keys:  
 sample\_dict.pop(k)  
print(sample\_dict)

**Solution 2**: Dictionary Comprehension

sample\_dict = {  
 "name": "Kelly",  
 "age": 25,  
 "salary": 8000,  
 "city": "New york"  
}  
*# Keys to remove*keys = ["name", "salary"]  
  
sample\_dict = {k: sample\_dict[k] for k in sample\_dict.keys() - keys}  
print(sample\_dict)

### **Exercise 7: Check if a value exists in a dictionary**

We know how to check if the key exists in a dictionary. Sometimes it is required to check if the given value is present.

Write a Python program to check if value 200 exists in the following dictionary.

**Given**:

sample\_dict = {'a': 100, 'b': 200, 'c': 300}

**Expected output:**

200 present in a dict

sample\_dict = {'a': 100, 'b': 200, 'c': 300}  
if 200 in sample\_dict.values():  
 print('200 present in a dict')

### **Exercise 8: Rename key of a dictionary**

Write a program to rename a key city to a location in the following dictionary.

**Given**:

sample\_dict = {

"name": "Kelly",

"age":25,

"salary": 8000,

"city": "New york"

}

**Expected output:**

{'name': 'Kelly', 'age': 25, 'salary': 8000, 'location': 'New york'}

sample\_dict = {  
 "name": "Kelly",  
 "age": 25,  
 "salary": 8000,  
 "city": "New york"  
}  
  
sample\_dict['location'] = sample\_dict.pop('city')  
print(sample\_dict)

### **Exercise 9: Get the key of a minimum value from the following dictionary**

sample\_dict = {

'Physics': 82,

'Math': 65,

'history': 75

}

**Expected output:**

Math

sample\_dict = {  
 'Physics': 82,  
 'Math': 65,  
 'history': 75  
}  
print(min(sample\_dict, key=sample\_dict.get))

### **Exercise 10: Change value of a key in a nested dictionary**

Write a Python program to change Brad’s salary to 8500 in the following dictionary.

**Given**:

sample\_dict = {

'emp1': {'name': 'Jhon', 'salary': 7500},

'emp2': {'name': 'Emma', 'salary': 8000},

'emp3': {'name': 'Brad', 'salary': 500}

}

**Expected output:**

{

'emp1': {'name': 'Jhon', 'salary': 7500},

'emp2': {'name': 'Emma', 'salary': 8000},

'emp3': {'name': 'Brad', 'salary': 8500}

}

sample\_dict = {  
 'emp1': {'name': 'Jhon', 'salary': 7500},  
 'emp2': {'name': 'Emma', 'salary': 8000},  
 'emp3': {'name': 'Brad', 'salary': 6500}  
}  
  
sample\_dict['emp3']['salary'] = 8500  
print(sample\_dict)

# Dictionary Quiz:

1. Select correct ways to create an empty dictionary

 sampleDict = {}

 sampleDict = dict()

 sampleDict = dict{}

# a, b

2. Please select all correct ways to empty the following dictionary

student = {

"name": "Emma",

"class": 9,

"marks": 75

}

 del student

 del student[0:2]

 student.clear()

C

3. In Python, Dictionaries are immutable

False

4. What is the output of the following

sampleDict = **dict**([

('first', 1),

('second', 2),

('third', 3)

])

**print**(sampleDict)

{‘first’: 1, ‘second’: 2, ‘third’: 3}

5. Dictionary keys must be immutable

True

6. What is the output of the following dictionary operation

dict1 = {"name": "Mike", "salary": 8000}

temp = dict1.get("age")

**print**(temp)

None

7. What is the output of the following code

dict1 = {"key1":1, "key2":2}

dict2 = {"key2":2, "key1":1}

**print**(dict1 == dict2)

True

8. Select the all correct way to **remove** the key **marks** from a dictionary

student = {

"name": "Emma",

"class": 9,

"marks": 75

}

 student.pop("marks")

 del student["marks"]

 student.remove("marks")

 student.popitem("marks")

A, b

* The pop() method used to remove the item by specifying the key name as an argument.
* The popitem() method removes the last inserted item
* Using the del keyword, we can remove the item from a dictionary by specifying the key name

9. Items are accessed by their position in a dictionary and All the keys in a dictionary must be of the same type

False

10. Select all correct ways to copy a dictionary in Python

 dict2 = dict1.copy()

 dict2 = dict(dict1)

 dict2 = dict1

A, b

11. What is the output of the following dictionary operation

dict1 = {"name": "Mike", "salary": 8000}

temp = dict1.pop("age")

**print**(temp)

 KeyError: ‘age’

 None

# A

# The pop() method removes the item from the dictionary. A KeyError will be thrown if the key doesn’t exist.

12. Select the correct way to access the value of a history subject

sampleDict = {

"class":{

"student":{

"name":"Mike",

"marks":{

"physics":70,

"history":80

}

}

}

}

# sampleDict['class']['student']['marks']['history']

13. Select the correct way to print Emma’s age.

student = {1: {'name': 'Emma', 'age': '27', 'sex': 'Female'},

2: {'name': 'Mike', 'age': '22', 'sex': 'Male'}}

# student[1]["age"]

14. Select the correct ways to get the value of marks key.

student = {

"name": "Emma",

"class": 9,

"marks": 75

}

 m = student.get(2)

 m = student.get('marks')

 m = student[2])

 m = student['marks'])

B, d