Python Dictionary exercise

Exercise 1: Convert two lists into a dictionary

Below are the two 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. The output should be:

{'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.copy()

dict3.update(dict2)

print(dict3)
```

Exercise 3: 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}
```

Solution

```
employees = ['Kelly', 'Emma']

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

res = dict.fromkeys(employees, defaults)

print(res)

# Individual data

print(res["Kelly"])
```

Exercise 4: 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"]
```

Solution1:

Expected output:

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

```
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 va;ue from sample_dict
    res.update({k: sample_dict[k]})
print(res)
```

Exercise 5: 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 6: 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 7: 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"
}
```

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

Exercise 8: 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}
}
```

Solution

```
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)
```