

LAB-4 Programs on Dictionary Operations

Aim:

To explore programs on dictionary operations using python 3

Task:

1. Write a Python script to sort (ascending and descending) a dictionary by value
2. Write a Python Program to generate a temperature profile dictionary (values in range:30 to 100)for ten days randomly (from august 1 to august 31) and check whether august 10 data exist in a dictionary or not. (Key=date,value=temperate value)
3. In program 2, find the count of temperature 30,40 in the dictionary.
4. Repeat the program 2 for July month and write a python program to concatenate the two dictionaries into one dictionary named July_aug_temp.
5. Write a Python Program to multiply all the items in the dictionary "July_aug_temp " which is generated in program 4 and find average temperature for the two months.

Algorithm:

Task 1:

- 1.Start
- 2.Initialize a **sample_dict** with key-value pairs.
- 3.Sort **sample_dict** items by their values in ascending and descending order using the **sorted** function and a lambda parameter as the key for sorting.
- 4.Maintain two separate dictionaries for storing the ascending and descending order of values.
- 5.End

Task 2:

- 1.Start
- 2.Import **random** module to generate random temperature profiles.
- 3.Initialize an empty dictionary **aug_temp_profile** to store temperature for each day .
- 4.Generate the temperature data for 10 days inside the loop,use the **random** module to generate temperatures between 30 and 100(inclusive).
- 5.Print the **aug_temp_profile** dictionary with the key-value pairs for 10 days.
- 6.Use the **get()** function to check whether the temperature for August 10 is recorded or not.
- 7.End

Task 3:

- 1.Start
- 2.Initialize an empty dictionary **temp_profile** and two count variables **count1** and **count2** as 0.
3. Inside the for loop update the **temp_profile** dictionary with random values of temperatures in between 30 and 40(inclusive).
- 4.Check and count the occurrences of the values 30 and 40 and increment their count variables.
- 5.Print the **temp_profile** dictionary and the final counts of occurrences for the temperature values.
- 6.End

Task 4:

- 1.Start
- 2.Import **random** module to generate random temperature profiles.
- 3.Initialize two empty dictionaries **aug_temp_profile** and **july_temp_profile**.
- 4.Generate August temperature profile from August 1 to 10 and July temperature profile from July 10 to 20 using the random module and update them in their respective dictionaries.
- 5.Create a new dictionary which combines the contents for both the dictionaries and print it.
- 6.End

Task 5:

1. Start
- 2.Initialize two empty dictionaries **aug_temp_profile** and **july_temp_profile** and concatenate them and form a new dictionary **Con_dict**.
- 3.In order to find the product of temperature, iterate through each temperature value in **Con_dict** and multiply each value with the current value.
- 4.In order to find the average temperature, calculate the sum of all the temperature values and divide it by the total number of temperature values in **Con_dict**.
- 5.Display the concatenated dictionary and the calculated product and average of temperature values.
- 6.End

Programs:

Task 1:

1. Write a Python script to sort (ascending and descending) a dictionary by value

```
In [15]: sample_dict = {'a': 30, 'b': 15, 'c': 45, 'd': 10}
ascending_sorted = dict(sorted(sample_dict.items(), key=lambda item: item[1]))
descending_sorted = dict(sorted(sample_dict.items(), key=lambda item: item[1], reverse=True))

print("Ascending:", ascending_sorted)
print("Descending:", descending_sorted)
```

Task 2:

2. Write a Python Program to generate a temperature profile dictionary (values in range:30 to 100) for ten days randomly (from august 1 to august 31) and check whether august 10 data exist in a dictionary or not. (Key=date,value=temperate value)

```
In [27]: import random
aug_temp_profile = {}
for i in range(1,11):
    aug_temp_profile.update({i:random.randint(30,100)})
print(aug_temp_profile)
print(aug_temp_profile.get(10))
```

Task 3:

3. In program 2, find the count of temperature 30,40 in the dictionary.

```
In [26]: import random
temp_profile = {}
count1=0
count2=0
for i in range(1,11):
    temp_profile.update({i:random.randint(30,40)})
print(temp_profile)
for a in temp_profile:
    if temp_profile[a]==30:
        count1+=1
for b in temp_profile:
    if temp_profile[b]==40:
        count2+=1
print("Count of temperature 30 in temperature profile :",count1)
print("Count of temperature 40 in temperature profile :",count2)
```

Task 4:

4. Repeat the program 2 for July month and write a python program to concatenate the two dictionaries into one dictionary named July_aug_temp.

```
In [31]: import random
aug_temp_profile = {}
for i in range(1,11):
    aug_temp_profile.update({i:random.randint(30,100)})
print("August Temperature Profile :",aug_temp_profile)
july_temp_profile = {}
for i in range(10,21):
    july_temp_profile.update({i:random.randint(30,100)})
print("July Temperature Profile :",july_temp_profile)
Con_dict={**aug_temp_profile,**july_temp_profile}
print("Concatenated Dictionaries :",Con_dict)
```

Task 5:

5. Write a Python Program to multiply all the items in the dictionary "July_aug_temp" which is generated in program 4 and find average temperature for the two months.

```
In [32]: Con_dict={**aug_temp_profile,**july_temp_profile}
print("Concatenated Dictionaries :",Con_dict)
product = 1
for temperature in Con_dict.values():
    product *= temperature

average_temperature = sum(Con_dict.values()) / len(Con_dict)
print("Product of all temperatures:", product)
print("Average temperature:", average_temperature)
```

Output:

Task 1:

```
Ascending: {'d': 10, 'b': 15, 'a': 30, 'c': 45}
Descending: {'c': 45, 'a': 30, 'b': 15, 'd': 10}
```

Task 2:

```
{1: 86, 2: 37, 3: 46, 4: 74, 5: 92, 6: 94, 7: 38, 8: 50, 9: 51, 10: 70}
70
```

Task 3:

```
{1: 32, 2: 37, 3: 40, 4: 32, 5: 39, 6: 34, 7: 33, 8: 34, 9: 30, 10: 30}
Count of temperature 30 in temperature profile : 2
Count of temperature 40 in temperature profile : 1
```

Task 4:

August Temperature Profile : {1: 79, 2: 59, 3: 36, 4: 75, 5: 84, 6: 57, 7: 85, 8: 100, 9: 45, 10: 56}

July Temperature Profile : {10: 92, 11: 36, 12: 40, 13: 59, 14: 60, 15: 67, 16: 82, 17: 82, 18: 100, 19: 56, 20: 93}

Concatenated Dictionaries : {1: 79, 2: 59, 3: 36, 4: 75, 5: 84, 6: 57, 7: 85, 8: 100, 9: 45, 10: 92, 11: 36, 12: 40, 13: 59, 14: 60, 15: 67, 16: 82, 17: 82, 18: 100, 19: 56, 20: 93}

Task 5:

Concatenated Dictionaries : {1: 79, 2: 59, 3: 36, 4: 75, 5: 84, 6: 57, 7: 85, 8: 100, 9: 45, 10: 92, 11: 36, 12: 40, 13: 59, 14: 60, 15: 67, 16: 82, 17: 82, 18: 100, 19: 56, 20: 93}

Product of all temperatures: 2536036575657829737839861760000000000

Average temperature: 69.35

Pre Lab Questions:

1. Write the function that removes the element with the specified key.
2. Write the function that returns the value of the specified key.
3. Add an item to the below dictionary with key 'color' and assign value to it as 'red':

```
thisdict = {  
    "brand": "Ford",  
    "model": "Mustang",  
    "year": 1964  
}
```

Post Lab Questions:

1. Write a python program that repeatedly asks the user to enter product names and prices. Store all of them in a dictionary whose keys are product names and values are prices. And also write a code to search an item from the dictionary.

1. Write a python program, that repeatedly asks the user to enter product names and prices. Store all of them in a dictionary whose keys are product names and values are prices. And also write a code to search an item from the dictionary.

```
In [3]: def main():
products = {}
while True:
    product_name = input("Enter product name (or 'exit' to stop): ")
    if product_name.lower() == 'exit':
        break
    product_price = float(input("Enter product price: "))
    products[product_name] = product_price
print("\nProduct dictionary:")
print(products)
search_item = input("\nEnter product name to search: ")
if search_item in products:
    print(f"The price of {search_item} is {products[search_item]}")
else:
    print(f"{search_item} not found in the dictionary.")

Enter product name (or 'exit' to stop): Apple
Enter product price: 50
Enter product name (or 'exit' to stop): Orange
Enter product price: 40
Enter product name (or 'exit' to stop): Banana
Enter product price: 15
Enter product name (or 'exit' to stop): Potato
Enter product price: 30
Enter product name (or 'exit' to stop): Onion
Enter product price: 45
Enter product name (or 'exit' to stop): exit

Product dictionary:
{'Apple': 50.0, 'Orange': 40.0, 'Banana': 15.0, 'Potato ': 30.0, 'Onion ': 45.0}

Enter product name to search: Orange
The price of Orange is 40.0
```

2. Write a Python program to get the maximum and minimum value in a dictionary.

2. Write a Python program to get the maximum and minimum value in a dictionary.

```
In [4]: def get_max_min(dictionary):
        if not dictionary:
            return None, None
        max_value = max(dictionary.values())
        min_value = min(dictionary.values())
        return max_value, min_value
def main():
    data = {
        'item1': 400,
        'item2': 150,
        'item3': 20,
        'item4': 85,
        'item5': 220
    }
    max_val, min_val = get_max_min(data)
    if max_val is not None and min_val is not None:
        print(f"Maximum value: {max_val}")
        print(f"Minimum value: {min_val}")
    else:
        print("Dictionary is empty.")

Maximum value: 400
Minimum value: 20
```

Result:

Thus, the various dictionary operations were studied and performed in python3.