



Faculty of Technology and Engineering

U & P U. Patel Department of Computer Engineering

Date: 15 / 1 / 2022

Practical List

DICTIONARY

- a. Write a Python script to check whether a given key already exists in dictionary.

CODE:

CASE 1:

```
D1 = {1:10, 2:20, 3:30}
K = 1
if K in D1:
    print('KEY IS PRESENT IN DICTIONARY')
else:
    print('KEY IS NOT PRESENT IN DICTIONARY')
```

OUTPUT:

```
KEY IS PRESENT IN DICTIONARY
```

CASE2:

```
D1 = {1:10, 2:20, 3:30}
K = 10
if K in D1:
    print('KEY IS PRESENT IN DICTIONARY')
else:
    print('KEY IS NOT PRESENT IN DICTIONARY')
```

OUTPUT:

```
KEY IS NOT PRESENT IN DICTIONARY
```

- b. Write a Python script to merge two Python dictionaries.

CODE:

```
D1 = {1:10, 3:30, 2:20}
D2 = {4:40, 5:50, 6:60}
D3 = D1.copy()
D3.update(D2)
print(D3)
```

OUTPUT:

```
{1: 10, 3: 30, 2: 20, 4: 40, 5: 50, 6: 60}
```

- c. Write a Python program to sum all the items in a dictionary.

CODE:

```
D1 = {1:100, 2:200, 3:300}
print('TOTAL SUM OF VALUES IN THE DICTIONARY:')
print(sum(D1.values()))
```

OUTPUT:

```
TOTAL SUM OF VALUES IN THE DICTIONARY:
600
```

- d. Write a Python script to add a key to a dictionary.

CODE:

```
D1 = {1:10, 2:20, 3:30}
D1.update({4:40})
print(D1)
```

OUTPUT:

```
{1: 10, 2: 20, 3: 30, 4: 40}
```

- e. Write a Python script to concatenate following dictionaries to create a new one.

Sample Dictionary :

dic1={1:10, 2:20}

dic2={3:30, 4:40}

dic3={5:50,6:60}

Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}

CODE:

```
dic1 = {1:10, 2:20}
dic2 = {3:30, 4:40}
dic3 = {5:50, 6:60}
dic4 = {}
dic4.update(dic1)
dic4.update(dic2)
dic4.update(dic3)
print(dic4)
```

OUTPUT:

```
{1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}
```

TUPLE

- a. Write a Python program to create a tuple with different data types.

CODE:

```
TUPLE1 = (1, 2, 3, 4, 5)
TUPLE2 = ('A', 'B', 'C')
TUPLE3 = ("HELLO", "WORLD")
print(TUPLE1)
print(TUPLE2)
print(TUPLE3)
```

OUTPUT:

```
(1, 2, 3, 4, 5)
('A', 'B', 'C')
('HELLO', 'WORLD')
```

- b. Write a Python program to create a tuple with numbers and print one item.

CODE:

```
TUPLE = [11, 22, 33, 44, 55]
print(TUPLE[2])
```

OUTPUT:

```
33
```

- c. Write a Python program to add an item in a tuple.

CODE:

```
TUPLE = (1, 2, 3, 4)
TUPLE = TUPLE + (5,)
print(TUPLE)
```

OUTPUT:

```
(1, 2, 3, 4, 5)
```

- d. Write a Python program to convert a tuple to a string.

CODE:

```
TUPLE = ('H', 'E', 'L', 'L', 'O')
STR = ''.join(TUPLE)
print(STR)
```

OUTPUT:

```
HELLO
```

- e. Write a Python program to find the length of a tuple.

CODE:

```
TUPLE = ('A', 'B', 'C', 'D', 'E')  
print('LENGTH OF TUPLE IS: ', len(TUPLE))
```

OUTPUT:

```
LENGTH OF TUPLE IS:  5
```

SET

- a. Write a Python program to add member(s) in a set and clear a set.

CODE:

```
SET: Set[str] = {'A', 'B', 'C', 'D'}  
SET.add('E')  
print('LETTERS ARE: ', SET)  
print('SET BEFORE CLEAR: ', SET)  
SET.clear()  
print('SET AFTER CLEAR', SET)
```

OUTPUT:

```
LETTERS ARE: {'E', 'B', 'A', 'D', 'C'}  
SET BEFORE CLEAR: {'E', 'B', 'A', 'D', 'C'}  
SET AFTER CLEAR set()
```

- b. Write a Python program to remove an item from a set if it is present in the set.

CODE:

```
SET = {'PYTHON', 'JAVA', 'PHP', 'ANDROID', '.NET'}  
SET.remove('ANDROID')  
print(SET)
```

OUTPUT:

```
{'JAVA', 'PYTHON', 'PHP', '.NET'}
```

- c. Write a Python program to create an intersection, Union, difference of sets.

CODE:

```
A = {0, 2, 4, 6, 8}  
B = {1, 2, 3, 4, 5}  
print("INTERSECTION IS: ", A & B)  
print("UNION IS: ", A | B)  
print("DIFFERENCE IS: ", A - B)
```

OUTPUT:

```
INTERSECTION IS: {2, 4}  
UNION IS: {0, 1, 2, 3, 4, 5, 6, 8}  
DIFFERENCE IS: {0, 8, 6}
```

- d. Write a Python program to find maximum and the minimum value in a set.

CODE:

```
SET = {1, 2, 3, 4, 5}  
print(min(SET))  
print(max(SET))
```

OUTPUT:

```
1  
5
```

- e. Write a Python program to find the most common elements and their counts from list, tuple, dictionary.

CODE:

```
def IntersecOfSets(arr1, arr2, arr3):  
    s1 = set(arr1)  
    s2 = set(arr2)  
    s3 = set(arr3)  
    print('List = ', arr1)  
    print('Tuple = ', arr2)  
    print('Dictionary = ', arr3)  
    set1 = s1.intersection(s2)  
    result_set = set1.intersection(s3)  
    final_list = set(result_set)  
    print('common of members of list, tuple & dictionary = ', final_list)  
  
    if_name_=='__main__'  
    list1 = [1, 2, 'ABC', 'xyz']  
    tuple1 = (80, 50, 'ABC', 'xyz')  
    dictionary1 = {300, 900, 'ABC', 'xyz'}  
    IntersecOfSets(list1, tuple1, dictionary1)
```

OUTPUT:

```
{'xyz', 'ABC'}
```