Deshmukh Shubham Kailas
Roll No: 11

Assignment 3

SET A List

```
# 1) Write a Python program to sum all the items in a list.

total = 0

list = [17, 9, 00, 9, 17]

for item in range(0, len(list)):
    total = total + list[item]

print("Sum of all elements in given list:",total)

"""
Sum of all elements in given list: 52
"""



# 2) Write a Python program to multiplies all the items in a list.

def mult_list(list):

    product = 1
    for i in list:
        product = product * i
    return product

list1 = [17, 9, 8, 1]
print(list1)
print("product: ", mult_list(list1))

"""
[17, 9, 8, 1]
product:   1224
"""

# 3) Write a Python program to get a list, sorted in increasing order by the last
element in each tuple from a given list of non-empty tuples.

def last(n):
    return n[-1]

def sort(tuples):
```

```
    return sorted(tuples, key=last)

a=[(1, 3), (3, 2), (2, 1)]
print("Sorted List:",sort(a))

"""
Sorted List: [(2, 1), (3, 2), (1, 3)]
"""
```

SET A Tuples

```
# 1) Write a Python program to create a tuple.
x = (10, 20, 30, 40, 50)
print(x)
print("Datatype of y= ", type(x))

# 2) Write a Python program to create a tuple with different data types.

t1 = ("tuple", False, 3.2, 17)
print(t1)


# 3) Write a Python program to check whether an element exists within a tuple.

t1 = ("p", "y", "t", "h", "o", "n", "d", "s", "k")
print("d" in t1)
print("s" in t1)
print(5 in t1)

"""
True
True
False
"""
```

SET A Sets

```
# 1) Write a Python program to create a set.
s1 = {2, 4, 6, 8, 10}
print(s1)
print(type(s1))

x=set(['zoo','cat','jaz','zoo','box'])
print(x)
print(type(x))

"""
{2, 4, 6, 8, 10}
<class 'set'>
```

```
{'cat', 'box', 'zoo', 'jaz'}
<class 'set'>
"""



# 2) Write a Python program to iterate over sets.

num_set = set([0, 1, 2, 3, 4, 5])
for n in num_set:
  print(n, end=' ')

print("\n\nCreating a set using string:")
char_set = set("Python")

for val in char_set:
     print(val, end=' ')

"""
0 1 2 3 4 5

Creating a set using string:
t h o P y n

0 1 2 3 4 5

Creating a set using string:
h y P o t n
"""


# 3) Write a Python program to create set difference.

set1 = set([1, 1, 2, 3, 4, 5])
set2 = set([1, 5, 6, 7, 8, 9])

print("\nOriginal sets:")
print(set1)
print(set2)

r1 = set1.difference(set2)
print("\nDifference of set1 - set2:")
print(r1)

r2 = set2.difference(set1)
print("\nDifference of set2 - set1:")
print(r2)

"""


Original sets:
```

```
{1, 2, 3, 4, 5}
{1, 5, 6, 7, 8, 9}

Difference of set1 - set2:
{2, 3, 4}

Difference of set2 - set1:
{8, 9, 6, 7}
"""
```

SET A Dictionary

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

import operator
d = {1: 2, 3: 4, 4: 3, 2: 1, 0: 0}
print('Original dictionary : ',d)

Sort_dict = dict( sorted(d.items(), key=operator.itemgetter(1)))
print('Ascending order by value : ',Sort_dict)

Sort_dict = dict( sorted(d.items(), key=operator.itemgetter(1),reverse=True))
print('Descending order by value : ',Sort_dict)

"""
Original dictionary :   {1: 2, 3: 4, 4: 3, 2: 1, 0: 0}
Ascending order by value :   {0: 0, 2: 1, 1: 2, 4: 3, 3: 4}
Descending order by value :   {3: 4, 4: 3, 1: 2, 2: 1, 0: 0}
"""


# 2) Write a Python script to add a key to a dictionary.

d = {0:10, 1:20}
print(d)
d.update({2:30})
print("Updated Dictionary with key :")
print(d)

"""
{0: 10, 1: 20}
Updated Dictionary with key :
{0: 10, 1: 20, 2: 30}
"""


# 3) Write a Python program to iterate over dictionaries using for loops.

d = {'Red': 1, 'Green': 2, 'Blue': 3}
for color_key, value in d.items():
```

```
    print(color_key, 'corresponds to ', d[color_key])

"""
Red corresponds to   1
Green corresponds to   2
Blue corresponds to   3
"""
```

SET B List

```
# 1. Write a Python program to remove duplicates from a list.

list1 = [1, 2, 3, 1, 2, 4, 5, 4 ,6, 2]
print("List Before removing duplicates :\n", list1)
list2 = [] #Temporary List

for i in list1:
    if i not in list2:
        list2.append(i)

list1 = list2

print("List After removing duplicates :\n", list1)

"""
List Before removing duplicates :
 [1, 2, 3, 1, 2, 4, 5, 4, 6, 2]
List After removing duplicates :
 [1, 2, 3, 4, 5, 6]
"""


# 2. Write a Python program to check a list is empty or not.

def Enquiry(lis1):
    if len(lis1) == 0:
        return 0
    else:
        return 1

# Driver Code
lis1 = []
if Enquiry(lis1):
    print ("The list is not empty")
else:
    print("Empty List")
```

SET B Tuples

```
#1. Write a Python program to convert a list to a tuple.

def convert(list):
    return tuple(list)

list = [1, 2, 3, 4]
print(convert(list))

"""
(1, 2, 3, 4)
"""

# 2. Write a Python program to remove an item from a tuple.

tuple1 = [(1,2), (3.78, 9.56), ("Python", "Study hard")]
tuple1.pop(2)
print(tuple1)

"""
[(1, 2), (3.78, 9.56)]
"""


# 3. Write a Python program to slice a tuple.
numTuple = (11, 22, 33, 44, 55, 66, 77, 88, 99, 100)
print("Tuple Items = ", numTuple)

slice1 = numTuple[2:6]
print("Tuple Items from 3 to 5 = ", slice1)

"""
Tuple Items =   (11, 22, 33, 44, 55, 66, 77, 88, 99, 100)
Tuple Items from 3 to 5 =   (33, 44, 55, 66)
"""


# 4. Write a Python program to find the length of a tuple.

tuple1 = (10, 20, 30, 40, 50)
print("Tuple Items = ", tuple1)

print("Tuple Length = ", len(tuple1))

"""
Tuple Items =   (10, 20, 30, 40, 50)
Tuple Length =   5
"""
```

SET B Sets

```
# 1. Write a Python program to check if a set is a subset of another set.

A = {1, 2, 3}
B = {1, 2, 3, 4, 5}
C = {1, 2, 4, 5}

print("A is SubSet B :", A.issubset(B))

print("B is SubSet A :", B.issubset(A))

print("A is SubSet C :", A.issubset(C))

print("C is SubSet B :", C.issubset(B))


"""
A is SubSet B : True
B is SubSet A : False
A is SubSet C : False
C is SubSet B : True
"""

# 2. Write a Python program to find maximum and the minimum value in a set.

setn = {5, 10, 3, 15, 2, 20}
print("Original set elements:")
print(setn)
print(type(setn))

print("\nMaximum value of the said set:")
print(max(setn))

print("\nMinimum value of the said set:")
print(min(setn))

"""
Original set elements:
{2, 3, 20, 5, 10, 15}
<class 'set'>

Maximum value of the said set:
20

Minimum value of the said set:
2
"""

# 3. Write a Python program to find the length of a set.
```

```
setn = {5, 10, 3, 15, 2, 20}
print("\nOriginal set elements:")
print(setn)
print(type(setn))
print("Length of the set:")
print(len(setn))

setn = {5, 5, 5, 5, 5, 5}
print("\nOriginal set elements:")
print(setn)
print("Length of the set:")
print(len(setn))

setn = {5, 5, 5, 5, 5, 5, 7}
print("\nOriginal set elements:")
print(setn)
print("Length of the set:")
print(len(setn))


"""
Original set elements:
{2, 3, 20, 5, 10, 15}
<class 'set'>
Length of the set:
6

Original set elements:
{5}
Length of the set:
1

Original set elements:
{5, 7}
Length of the set:
2
"""
```

SET B Dictionary

```
# 1. Write a Python script to generate and print a dictionary that contains a number (between 1 and n) in the form (x, x*x).
n=int(input("Input a number :"))
d = dict()

for x in range(1,n+1):
    d[x]=x*x

print("A number (between 1 and n) in the form (x, x*x) :\n ",d)
```

```
"""
Input a number :10
A number (between 1 and n) in the form (x, x*x) :
  {1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100}
"""

# 2. Write a Python script to merge two Python dictionaries.

d1 = {'a': 100, 'b': 200}
print("Dictionary 1:",d1)
d2 = {'x': 300, 'y': 200}
print("\nDictionary 2:",d2)
d = d1.copy()
d.update(d2)
print("\nMerged Dictionary :\n",d)

"""
Dictionary 1: {'a': 100, 'b': 200}

Dictionary 2: {'x': 300, 'y': 200}

Merged Dictionary :
 {'a': 100, 'b': 200, 'x': 300, 'y': 200}
"""

# 3. Write a Python program to get a dictionary from an object's fields.
class dictObj(object):
    def __init__(self):
        self.x = 'red'
        self.y = 'Yellow'
        self.z = 'Green'
    def do_nothing(self):
        pass
test = dictObj()
print(test.__dict__)
```