

'''

Experiment 1.2 : Impementing data type like list, set, tuple and dictionary

Name: Khan Arshad Abdulla
Roll No: 20CO24
2021-2022

THEORY:

a)List:-

A list in Python is used to store the sequence of various types of data. Python lists are mutable type its mean we can modify its element after it created. However, Python consists of six data-types that are capable to store the sequences, but the most common and reliable type is the list. A list can be defined as a collection of values or items of different types. The items in the list are separated with the comma (,) and enclosed with the square brackets [].

b)Tuple:-

Python Tuple is used to store the sequence of immutable Python objects. The tuple is similar to lists since the value of the items stored in the list can be changed, whereas the tuple is immutable, and the value of the items stored in the tuple cannot be changed. A tuple can be written as the collection of comma-separated (,) values enclosed with the small () brackets. The parentheses are optional but it is good practice to use

c)Set:-

A Python set is the collection of the unordered items. Each element in the set must be unique, immutable, and the sets remove the duplicate elements. Sets are mutable which means we can modify it after its creation. Unlike other collections in Python, there is no index attached to the elements of the set, i.e., we cannot directly access any element of the set by the index. However, we can print them all together, or we can get the list of elements by looping through the set. The set can be created by enclosing the comma-separated immutable items with the curly braces {}. Python also provides the set() method, which can be used to create the set by the passed sequence.

d)Dictionary:-

Python Dictionary is used to store the data in a key-value pair format. The dictionary is the data type in Python, which can simulate the real-life data arrangement where some specific value exists for some particular key. It is the mutable data-structure.

The dictionary is defined into element Keys and values.
 Keys must be a single element
 Value can be any type such as list, tuple, integer, etc.
 In other words, we can say that a dictionary is the collection of key-value pairs where the value can be any Python object. In contrast, the keys are the immutable Python object, i.e., Numbers, string, or tuple.
 The dictionary can be created by using multiple key-value pairs enclosed with the curly brackets {}, and each key is separated from its value by the colon (:).

```
'''
l=[]                # an empty list
l1 = list()        # an empty list.
l= [10,20,30,]
ch = 'y'
while ch=='y':
    l1.append(int(input("Enter a number:")))
    ch= input('Do you want to add more? (y/n):')
print(l1)

print('Reverse is:\t',l1[::-1])
print('Sorted list:\t',sorted(l1))

l2 = [x*2 for x in l1] # list comprehension
print('Double list:\t',l2)

l3 = [x**2 for x in l1 if x%2==1] # list comprehension for getting
squares of odd vlaues elements from l1.
print("Square of odd values",l3)
##### set #####
s=set()          #an empty set
sd={}            #its an empty dictionary
s={1,2,3,1,1,2,4,5,8,9,7} # duplicate values will be removed.
print("Set 1 is",s)
s1={3,6,7,5,6,8}
print("Set 2 is",s1)
print("Union of Set1 and Set2 is",s.union(s1))
print("Intersection of Set1 and Set2 is",s.intersection(s1))
print("Is Set1 is subset of Set2? :",s.issubset(s1))
print("Help on Set class :",dir(set))

#### Tuple ####
print("\n\t\t\t\t\t##### Tuple #####")
t=()
t1=tuple()
t=(1,2,3,'AIKTC',9.92,'A','A')
print("Tuple is",t)
print("Occurance of 2 is ",t.count('AIKTC'))
print("Occurance of A" ,t.count('A'))
if 'K' in t[3]:
    print("yes")
else:
    print("no")
print("Help on tuple class",dir(tuple))
'''
```

```

#### dictionary ####
print("\n\t\t#### Dictionary ####")
d={1:''}
d1=dict()
d.update({1:'FE'})
d.update({2:'SE'})
print('Dictionary is',d)
l=d.items()
for i in l:
    print(i)
print("Values of given key is",d.get("1"))
print("Keys of dictionary are",d.keys())
print("help dictionary class",dir(dict))

```

'''

OUTPUT:

```

Enter a number:5
Do you want to add more? (y/n):y
Enter a number:15
Do you want to add more? (y/n):y
Enter a number:10
Do you want to add more? (y/n):y
Enter a number:20
Do you want to add more? (y/n):n
[5, 15, 10, 20]
Reverse is:      [20, 10, 15, 5]
Sorted list:     [5, 10, 15, 20]
Double list:     [10, 30, 20, 40]
Square of odd values [25, 225]
Set 1 is {1, 2, 3, 4, 5, 7, 8, 9}
Set 2 is {3, 5, 6, 7, 8}
Union of Set1 and Set2 is {1, 2, 3, 4, 5, 6, 7, 8, 9}
Intersection of Set1 and Set2 is {8, 3, 5, 7}
Is Set1 is subset of Set2? : False
Help on Set class : ['__and__', '__class__', '__class_getitem__',
 '__contains__', '__delattr__',
 '__dir__', '__doc__', '__eq__', '__format__', '__ge__',
 '__getattr__', '__gt__', '__hash__',
 '__iand__', '__init__', '__init_subclass__', '__ior__', '__isub__',
 '__iter__', '__ixor__',
 '__le__', '__len__', '__lt__', '__ne__', '__new__', '__or__', '__rand__',
 '__reduce__',
 '__reduce_ex__', '__repr__', '__ror__', '__rsub__', '__rxor__',
 '__setattr__', '__sizeof__',
 '__str__', '__sub__', '__subclasshook__', '__xor__', 'add', 'clear',
 'copy', 'difference',
 'difference_update', 'discard', 'intersection', 'intersection_update',
 'isdisjoint', 'issubset',
 'issuperset', 'pop', 'remove', 'symmetric_difference',
 'symmetric_difference_update', 'union', 'update']

```

Tuple

```

Tuple is (1, 2, 3, 'AIKTC', 9.92, 'A', 'A')
Occurance of 2 is 1
Occurance of A 2

```

```

yes
Help on tuple class ['__add__', '__class__', '__class_getitem__',
 '__contains__', '__delattr__',
 '__dir__', '__doc__', '__eq__', '__format__', '__ge__',
 '__getattr__', '__getitem__',
 '__getnewargs__', '__gt__', '__hash__', '__init__', '__init_subclass__',
 '__iter__', '__le__',
 '__len__', '__lt__', '__mul__', '__ne__', '__new__', '__reduce__',
 '__reduce_ex__', '__repr__',
 '__rmul__', '__setattr__', '__sizeof__', '__str__', '__subclasshook__',
 'count', 'index']

```

```

##### Dictionary #####
Dictionary is {1: 'FE', 2: 'SE'}
(1, 'FE')
(2, 'SE')
Values of given key is None
Keys of dictionary are dict_keys([1, 2])
help dictionary class ['__class__', '__class_getitem__', '__contains__',
 '__delattr__',
 '__delitem__', '__dir__', '__doc__', '__eq__', '__format__', '__ge__',
 '__getattr__',
 '__getitem__', '__gt__', '__hash__', '__init__', '__init_subclass__',
 '__ior__', '__iter__',
 '__le__', '__len__', '__lt__', '__ne__', '__new__', '__or__',
 '__reduce__', '__reduce_ex__',
 '__repr__', '__reversed__', '__ror__', '__setattr__', '__setitem__',
 '__sizeof__', '__str__',
 '__subclasshook__', 'clear', 'copy', 'fromkeys', 'get', 'items', 'keys',
 'pop', 'popitem',
 'setdefault', 'update', 'values']

```

CONCLUSION:

In this particular experiment we have successfully implemented data type like list, set, tuple, and dictionary. list in Python is used to store the sequence of various types of data and Python Tuple is used to store the sequence of immutable Python objects, A Python set is the collection of the unordered items. Each element in the set must be unique, immutable, and the sets remove the duplicate elements, while Python Dictionary is used to store the data in a key-value pair format.