Task.4: use various data types, list. Tuple. and Dictionary in python Programmins

Aim:-

To use various data type, list, Tuples and Dictonary in Phython Programmins.

Q) You are working on a Python project that requires you to manage and manipulate a list of numbers your task is to create a python program that demonstrates the following list operations;

Algorithm:-

1. Start
2. For adding elements to a list first create a list with name "list and assign the values within [] brackets, in order to add a new value use the function append()
3. For removing a specific element use "pop index(value)" or remove (item name)".
4. For Sorting the elements use "sorted (list)" function.
5. For finding minimum value use "min list)" and for maximum use "max (list)".
6. For Sum use function "sum (list)" and for avrange use the formula sum(list)len(list)
7. Print the output,
8. End.

Program:-

```
#Add Element: Add elements to the list.
    list = [10,20]
      a=30
list append(a)
Print(list)
# Remove Elements: Remove specific elements
                                  from list
```

output:

```
[10, 20, 30]
 [10, 30]
  [30]
 [5, 8, 9, 15, 30, 89]
The minimum value is: 5
 The maximum value is: 89.
   The sum is : 156
  The average : 26.0.
```

✓

list remov (10) # by itemname

Print(list)

sort Elements: sort the list in assending and descending order.

l=[5,8,9,15,30,89]

Print(sorted)(l)

Find Minimum and maximum: find the minimum and maximum elements in the list.

Print("The minimum value is:", min(l))

Print ("The maximum value is: "max(l))

Calculate sum and Average

Print("The sum is;" sum(l))

Print ("The average is:" ((sum(l)/len(l)))

b) you're tasked with Creating a python Program that ~~shows~~ cases operated on tuples. Tuples are immutable sequences similar to lists but with the key difference that they cannot be changed after Cretain, your Program should illustrate the following tuple operation.

Algorithm:-

1) Start

2) To create a tuple use "tuple-name=(values)".

3) To access the elements of a tuple either we the index values(tuple-name (index-value)) or tuple slicing (tuple-name[start:end])

4) To concatenate tuples use the operator '+' (tuple1 + tuple2)

5) Try to modify the tuple element by assigning the values directly like; tuple(index] = new-value, will result in an error as it immutable.

6) Print the output.

7) End.

output:

```
(10,'hello', 3.14,'world')
 10
hello
 3.14
world
('hello', 3.14)
 (10,'hello',3.14)

 (10,'hello', 3.14,'world', 5,0,5)
```

✓

Program:

```
# Create a tuple: Define a tuple with elements
  of different dat types (10, 'hello', 3.14, 'world')
  tuple = (10, 'hello', 3.14, 'world')
  Print(tuple)
 # Access Elements Access individual elements
   and slices of the tuple.
   for i in tuple:
    Print(i)
   Print(tuple[1:3])
   Print(tuple[::-1])
  # Concatenate tuples: Combine two tuples to
   Create a new tuple.
   t2 = (5,0.5)
    t3 = tuple + t2
    Print(t3)
  # Immutable nature: Attempt to modify element
     of the tuple and handle the resulting error.
    tuple(2) = "Pi" # Error.
```

(C) (B) You are tasked with creating a python that showcases operation on dictionaries - Dictionaries in python are unordered collections of items. Each item is pair consisting of a key and value.

Algorithm:

1. Start the Program
2. Define a dictionary with key pairs of different data types.
3. Retrive values from the dictionary using their corresponding keys.
4. modify dictionary.
5. Iterative Over Dictionary
6. Stop the Program.

output

```
{'name': Alice', 'age': 30', 'city': 'New York'}
Alice
30
{name:

{name:

key:name
key:age
dict-items([('name'; Jame'), ('age', 30)])
```

Program:-

```
# Create a dictionary, Define a dictionary with
  key-value,
  Pairs of different data type ({name: "Alice" age: 30,
  City "New York"})
  dictionary = {"name": "Alice", "age: 30, "city": newyork"
  Print (dictionary)
# Acess values: Acess values using keys.
  Print (dictionary [name])
  Print (dictionary ['age'])
# modify dictionary: update values, add new key value
  pairs, and remove existing pairs.
  dictionary ['name'] = "James"
  Print (dictionary)
# Iterate over Dictionary: use loops to iterate
  over keys or the values.
  for k in dictionary:
    Print ("key", k)
  Print (dictionary.items())
```

VEL TECH	
EX No.	4
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	20
SIGN WITH DATE	13/8

Result:-

Thus, various data types, list, tuples and Dictionary in python programming was used and verified successfully.