

## Task-4: Use Various data types, List, Tuples and Dictionary in python programming

Aim:

To use various data types, List, Tuples and Dictionary in python programming

a. 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.

Add Elements: Add elements to the list

Remove Elements: Remove specific elements from the list.

Sort Elements: Sort the list in ascending and descending order.

Find Minimum and Maximum: Find the minimum and maximum elements in the list

Calculate Sum and Average: Calculate the sum and average of the elements in the list.

### ALGORITHM

1. start

2. For adding elements to a list first create a list and assign the values with [] brackets 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 "sorted(list)" function.
6. For sum use function "sum(list)" and for average use the formula "sum(list) / len(list)"
7. Print the output
8. End

### PROGRAM

# Add Elements : Add elements to the list

list = [10, 20]

a = 30

list.append(a)

print(list)

# Remove Elements : Remove specific elements from the list.

list.remove(10) # by element

print(list)

# Sort Elements : Sort the list in ascending 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)))

out put

[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 is 26.0

o/h

5. You are tasked with creating a python program that show cases operations on tuples. Tuples are immutable sequences, similar to lists but with the key difference that they cannot be changed after creation - Your program should illustrate the following tuple operations:

1. Create a Tuple: Define a tuple with elements of different data types (10, 'hello', 3.14, 'world')
2. Access Elements: Access individual elements and slices of the tuple
3. Concatenate Tuples: Combine two tuples to create a new tuple.
4. Immutable Nature: Attempt to modify elements of the tuple and handle the resulting error.

#### ALGORITHM

1. start
2. To create a tuple use "tuple - name = (values)".
3. To access the elements of a tuple slicing (tuple - name [start : end]).
4. To concatenate tuples use the operator "+" (tuple1 "+" tuple2)
5. Try to modify the tuple elements by assigning the values directly like tuple [index] = new - value, will result in an error as it is immutable
6. End

## PROGRAM

# create a Tuple : Define a tuple with elements of different data 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 e in tuple:

```
print(i)
```

```
print (tuple[1:3])
```

```
print (tuple[: -1])
```

# concatenate Tuples : Combine two tuples to create a new tuple

```
t2 = (5, 10, 5)
```

```
t3 = tuple + t2
```

```
print (t3)
```

# Immutable Nature : Attempt to modify elements of the tuple and handle the resulting error

```
tuple[3] = "PI" # ERROR
```

out put

C10 , 'hello' , 3.14 , 'world')

to

hello

3.14

world

('hello' , 3.14)

C10 , 'hello' , 3.14)



c) You are tasked with creating a python program that shows operations on dictionaries. Dictionaries in python are unordered collection of items. Each item is a pair consisting of a key and a value. Your program should illustrate the following dictionary operations:

1. create a Dictionary :- Define a dictionary with key-value pairs of different data types. `{'name': 'Alice', 'age': 30, 'city': 'New York'}`

2. Access values :- Access values using keys

3. Modify Dictionary :- Update values, add new key-value pairs and remove existing pairs.

4. Iterate over Dictionary :- use loops to iterate over keys or values.

### Algorithm

1. start the program

2. Define a dictionary with key-value pairs of different data types

3. Retrieve values from the dictionary using their corresponding keys

4. modify Dictionary

5. Iterate over Dictionary

6. stop the program.

## PROGRAM

# Create a Dictionary : Define a dictionary with key - values pairs of different data types. ('name': 'Alice', 'age': 30, 'city': 'New York')

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

```
print (dictionary)
```

# Access Values : Access 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)
```

```
dictionary.pop ('city')
```

```
print (dictionary)
```

# Iterate Over Dictionary : Use loops to iterate over keys or values for k in dictionary.

```
print ("KEY:", k)
```

```
print (dictionary)
```

VELTECH	
Roll No.	45
PERFORMANCE (5)	5
RESULT ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	25
GN WITH DATE	3/3/24

Result:-

Thus, various data types, List, Tuples and Dictionary in python programming was used and verified successfully.



out put

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

Alice

30

```
{'name': 'James', 'age': 30, 'City': 'New  
York'}
```

```
{'name': 'James', 'age': 30}
```

KEY: name

KEY: age

dict - items [(('name', 'James'), ('age', 30))]