

# Practical 2

Aim : Create and initialize lists, tuples, sets, and dictionaries in python also perform operations like indexing, slicing, adding, removing, and sorting elements.

```
In [44]: l = [10, 20, 30, 40, 50]
         print('Original list:', l)
```

Original list: [10, 20, 30, 40, 50]

```
In [45]: print('First element:', l[0])
```

First element: 10

```
In [46]: print('Elements from index 1 to 3:', l[1:4])
```

Elements from index 1 to 3: [20, 30, 40]

```
In [47]: l.append(60)
         print('After appending 60:', l)
```

After appending 60: [10, 20, 30, 40, 50, 60]

```
In [48]: l.insert(2, 25)
         print('After inserting 25 at index 2:', l)
```

After inserting 25 at index 2: [10, 20, 25, 30, 40, 50, 60]

```
In [49]: l.remove(30)
         print('After removing 30:', l)
```

After removing 30: [10, 20, 25, 40, 50, 60]

```
In [50]: popped = l.pop(3)
         print('After popping element at index 3:', l)
```

After popping element at index 3: [10, 20, 25, 50, 60]

```
In [51]: l.sort()
         print('After sorting:', l)
```

After sorting: [10, 20, 25, 50, 60]

```
In [52]: l.reverse()
         print('After reversing:', l)
```

After reversing: [60, 50, 25, 20, 10]

```
In [53]: t = (5, 15, 25, 35, 45)
         print('Original tuple:', t)
```

Original tuple: (5, 15, 25, 35, 45)

```
In [54]: print('Second element:', t[1])
```

Second element: 15

```
In [55]: print('First three elements:', t[:3])
```

First three elements: (5, 15, 25)

```
In [56]: temp_list = list(t)
temp_list.append(55)
t = tuple(temp_list)
print('After appending 55:', t)
```

After appending 55: (5, 15, 25, 35, 45, 55)

```
In [57]: s = {100, 200, 300, 400}
print('Original set:', s)
```

Original set: {200, 100, 400, 300}

```
In [58]: s.add(500)
print('After adding 500:', s)
```

After adding 500: {100, 200, 300, 400, 500}

```
In [59]: s.remove(200)
print('After removing 200:', s)
```

After removing 200: {100, 300, 400, 500}

```
In [60]: s.discard(600)
print('After discarding 600 (non-existent):', s)
```

After discarding 600 (non-existent): {100, 300, 400, 500}

```
In [61]: set1 = {1, 2, 3, 4}
set2 = {3, 4, 5, 6}
union_set = set1 | set2
print('Union of set1 and set2:', union_set)
```

Union of set1 and set2: {1, 2, 3, 4, 5, 6}

```
In [62]: intersection_set = set1 & set2
print('Intersection of set1 and set2:', intersection_set)
```

Intersection of set1 and set2: {3, 4}

```
In [63]: difference_set = set1 - set2
print('Difference of set1 and set2:', difference_set)
```

Difference of set1 and set2: {1, 2}

```
In [64]: my_dict = {'a': 1, 'b': 2, 'c': 3}
print('Original dictionary:', my_dict)
```

Original dictionary: {'a': 1, 'b': 2, 'c': 3}

```
In [65]: my_dict['d'] = 4
print('After adding key d:', my_dict)
```

After adding key d: {'a': 1, 'b': 2, 'c': 3, 'd': 4}

```
In [66]: del my_dict['b']  
print('After removing key b:', my_dict)
```

After removing key b: {'a': 1, 'c': 3, 'd': 4}

```
In [67]: d = {'name': 'Alice', 'age': 25, 'city': 'New York'}  
print('Original dictionary:', d)
```

Original dictionary: {'name': 'Alice', 'age': 25, 'city': 'New York'}

```
In [68]: print('Name:', d['name'])
```

Name: Alice

```
In [69]: d['age'] = 26  
print('After updating age:', d)
```

After updating age: {'name': 'Alice', 'age': 26, 'city': 'New York'}

```
In [70]: d['profession'] = 'Engineer'  
print('After adding profession:', d)
```

After adding profession: {'name': 'Alice', 'age': 26, 'city': 'New York', 'profession': 'Engineer'}

```
In [71]: del d['city']  
print('After deleting city:', d)
```

After deleting city: {'name': 'Alice', 'age': 26, 'profession': 'Engineer'}

```
In [72]: removed_value = d.pop('age')  
print('After popping age:', d)
```

After popping age: {'name': 'Alice', 'profession': 'Engineer'}

```
In [73]: sorted_keys = sorted(d.keys())  
print('Sorted keys:', sorted_keys)
```

Sorted keys: ['name', 'profession']