SOMAIYA VIDYAVIHAR UNIVERSITY

K J Somaiya Institute of Management

Department of Data Science and Technology

Practical No: 01

Subject: Python Programming Lab

MCA / Sem I / Python Programming [Course Code : 217P09L102]

ROLL No: 09 **DATE:08/09/2023**

FULL NAME: Atharv Ankush Desai

Aim:	**Understanding Python Lists**
Topics Covered:	Creation, Access, Modification, Slicing, Iteration, Sorting, Comprehensions, Methods for list
Problem Statement:	1. Create a Python program that performs basic list operations. - Create an empty list. - Add three different types of elements to the list (e.g., integers, strings, and floats). - Print the list. - Remove one element from the list. - Print the modified list. 2. WAPP for list indexing and slicing in Python. - Create a list of numbers from 1 to 10. - Print the first and last elements of the list. - Print a slice of the list containing elements from index 3 to 7. - Modify the list to replace the elements from index 5 to 9 with even numbers (e.g., 10, 12, 14, ...). - Print the modified list. 3. WAPP to count occurrences of all the elements present in the list.

	- Provide a list. - Print all the elements of the list with their counts. - Print the list. 4. WAPP for List Sorting and Reversing list - Create a list of unsorted numbers. - Sort the list in ascending order. - Reverse the sorted list. - Print the reversed list. 5. WAPP to demonstrate list concatenation in Python. - Provide two lists of names (e.g., boys names and girls names). - concatenate the two lists **to create a combined list of names.** **- Print the combined list.**
Theory:	list(): Creates a new empty list or converts an iterable. append(): Adds an element to the end of the list. reverse(): Reverses the order of elements in the list. copy(): Creates a shallow copy of the list. sort(): Sorts the list in ascending order (in-place). sorted(): Returns a sorted copy of the list. count(): Counts the occurrences of an element in the list. [:]: For Splicing any given range [:::x]: For the iteration direction

Code:	1.
	Code :
	
	`# list with 3 diffrent types`
	`myList = [2,69.420,'LOL','KYS']`
	
	`# print List`
	`print(f'myList => {myList}')`
	
	`# Removing one ele from list`
	`myList.remove(2)`
	`print(f'myList after Removing => {myList}')`
	
	2.
	Code :
	`# list from 1 to 10`
	`myList = [ele for ele in range(1,11)]`

```
print(f'myList => {myList}')

# prin last ele of list
print(f'myList last ele => {myList[-1]}')

# spliced list from indx 3 - 7
print(f'myList 3-7 indx => {myList[3:7]}')

# modify all the ele from indx 5 - 9 with even numbers
myList[5:9] = [i for i in range(10,15,2)]
print(f'myList Modified => {myList}')
```

3.

Code :

```
import random, math
random.seed(420)

myList = [math.ceil(random.random() * 10) for i in range(1,21)]
myList_items = {* myList}

print(f"myList => {myList}")
for ele in myList_items:
    print(f"{ele} occurs {myList.count(ele)}")
```

4.

Code :

```
import random, math
random.seed(420)
myList = [math.ceil(random.random() * 100) for i in range(1,51)]
print(f'myList => {myList}')

# asc sort
myList.sort()
myList = [*{*myList}]
print(f'myList after sorting ascending => {myList}')

# reversing
myList = myList[::-1]
print(f'myList after sorting ascending and reversing => {myList}')
```

5.

Code :

```
boysList = ['Shubavrat','Swaraj','Atharv','Pratham']
girlsList = ['Girl1','Girl2','Girl3','Girl4']
```

```
# .append() or .extend() works 👍
combList = boysList + girlsList
print(f'Combined list => {combList}')
```

Screenshot of Output:

1.

```
myList => [2, 69.42, 'LOL', 'KYS']
myList  after Removing => [69.42, 'LOL', 'KYS']
```

2.

```
myList => [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
myList last ele => 10
myList 3-7 indx => [4, 5, 6, 7]
myList Modified => [1, 2, 3, 4, 5, 10, 12, 14, 10]
```

3.

```
myList => [1, 8, 3, 1, 7, 7, 1, 4, 10, 1, 8, 2, 4, 5, 2, 10, 3, 2, 2, 3]
1 occurs 4
2 occurs 4
3 occurs 3
4 occurs 2
5 occurs 1
7 occurs 2
8 occurs 2
10 occurs 2
```

4.

```
myList => [3, 79, 28, 10, 67, 66, 4, 33, 100, 2, 78, 13, 34, 44, 13, 92, 28, 15, 11, 27, 19, 67, 25, 1,
myList after sorting ascending => [1, 2, 3, 4, 5, 10, 11, 13, 15, 19, 20, 25, 27, 28, 30, 33, 34, 43, 44
myList after sorting ascending and reversing => [100, 98, 97, 95, 92, 91, 81, 80, 79, 78, 75, 72, 69, 68
```

5.

```
Combined list => ['Shubavrat', 'Swaraj', 'Atharv', 'Pratham', 'Girl1', 'Girl2', 'Girl3', 'Girl4']
```

+ Code + Markdown

Observations:

1. List is mutable
2. List can be added within another list or any other data-type

Conclusion:

List is a mutable dynamic data-type in python which stores various data-types within class<list> obj, allowing element manipulation, sorting, copying, and counting, making them essential for various programming tasks.

Subject-In-Charge:

Sign: _____

Prof. Mayura Nagar