**Experiment – 2.1(a)**

**Student Name: Milan Sharma UID: 23MAI10003**

**Branch: ME – CSE - AIML Section/Group: MAI – 1 (A)**

**Semester: 1st Date of Performance: 04 Oct 2023**

**Subject Name: Python Programming Subject Code: 23 CSH 623**

1. **Aim of the Experiment :**

Write a python program to create a list and demonstrate different inbuilt list methods .

1. **Objective of the Experiment :**

To create a list and demonstrate different inbuilt list methods .

1. **Algorithm/ Steps for Experiment**

1.         Python programming, a list is created by placing all the items (elements) inside square brackets[], separated by commas.

2.         It can have any numberof items and they may be of different types (integer, float, string etc.).

**Code for Experiment :**

my\_list = []

my\_list1 = [1, 2, 3]

my\_list2 = [1, "Hello", 3.4]

my\_list3 = ["mouse", [8, 4, 6], ['a']]

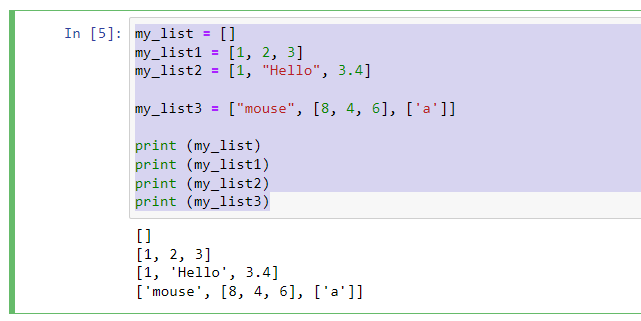
print (my\_list)

print (my\_list1)

print (my\_list2)

print (my\_list3)

**Result/Output :**

****

**Accessing elements from the List**

In order to access the list items refer to the index number.

Use the index operator [ ] to access an item in a list. The index must be an integer. Nested list are accessed using nested indexing.

**Code for Experiment :**

List = ["Geeks", "For", "Geeks"]

print("Accessing a element from the list")

print(List[0])

print(List[2])

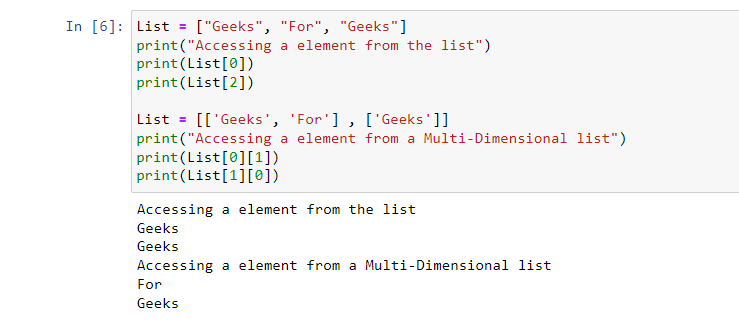
List = [['Geeks', 'For'] , ['Geeks']]

print("Accessing a element from a Multi-Dimensional list")

print(List[0][1])

print(List[1][0])

**Result/Output :**

****

**Inbuilt Methods for List**

**Using insert () method**

append() method only works for addition of elements at the end of the List, for the addition of elements at the desired position, the insert() method is used. Unlike append() which takes only one argument, the insert() method requires two arguments(position, value).

**Code for Experiment :**

List = [1,2,3,4]

print("Initial List: ")

print(List)

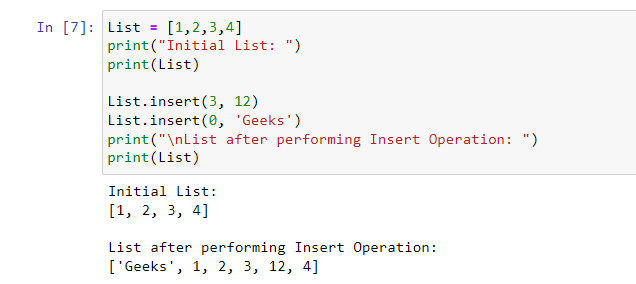
List.insert(3, 12)

List.insert(0, 'Geeks')

print("\nList after performing Insert Operation: ")

print(List)

**Result/Output :**



**Using extend() method**

# Other than append() and insert() methods, there’s one more method for Addition of elements, extend(), this method is used to add multiple elements at the same time at the end of the list.

Note – append() and extend() methods can only add elements at the end.

**Code for Experiment :**

List = [1,2,3,4]

print("Initial List: ")

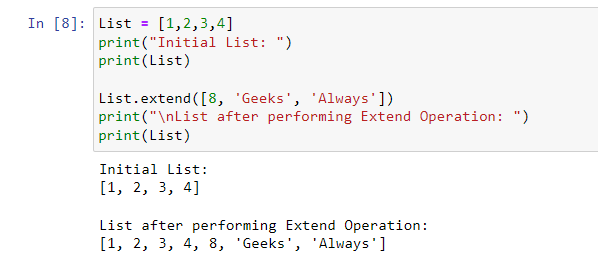
print(List)

List.extend([8, 'Geeks', 'Always'])

print("\nList after performing Extend Operation: ")

print(List)

**Result/Output :**



**Len() method**

**pop() method**

Pop() function can also be used to remove and return an element from the set, but by default it removes only the last element of the set, to remove element from a specific position of the List, index of the element is passed as an argument to the pop() method.

**Code for Experiment :**

List1 = []

print(len(List1))

List2 = [10, 20, 14]

print(len(List2))

List = [1,2,3,4,5]

List.pop()

print("\nList after popping an element: ")

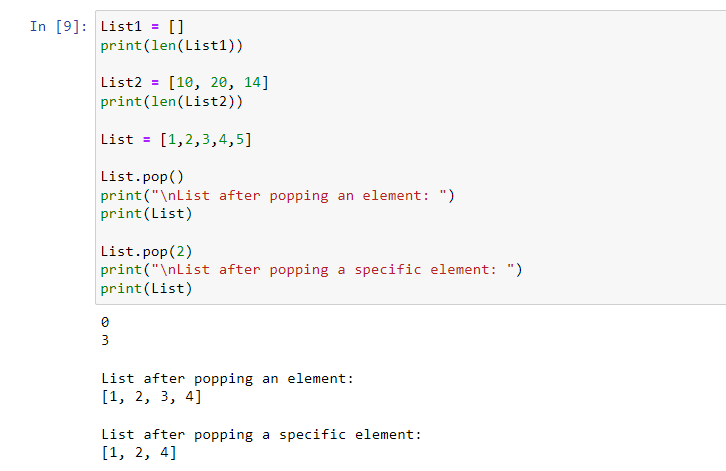
print(List)

List.pop(2)

print("\nList after popping a specific element: ")

print(List)

**Result/Output :**

****

**Removing Elements from the List Using remove() method, iterator method**

**Code for Experiment :**

List = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]

print("Initial List: ")

print(List)

List.remove(5)

List.remove(6)

print("\nList after Removal of two elements: ")

print(List)

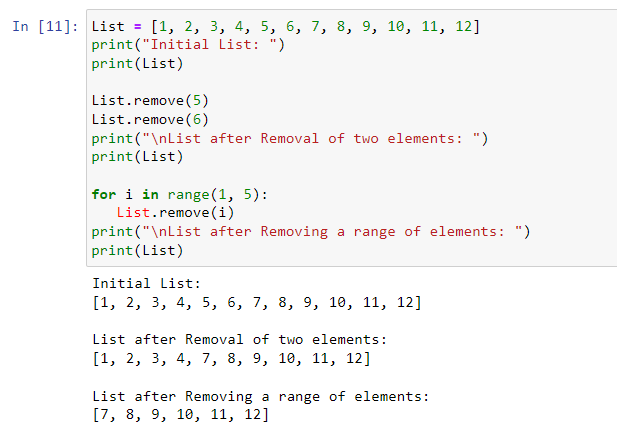
for i in range(1, 5):

List.remove(i)

print("\nList after Removing a range of elements: ")

print(List)

**Result/Output :**



**Learning outcomes (What I have learnt):**

1. Learnt different operations performed on list.
2. Learnt rhe different inbuild function that are performed on list
3. Learnt functions like len(), pop() etc.