In [10]:

```
car1="BMW"
car2="Volvo"
car3="Audi"
cars=["BMW","Volvo","Audi"]
print(cars)
for x in cars:
 print(x)
for i in range (1,3):
 print(cars[i])
```

```
['BMW', 'Volvo', 'Audi']
BMW
Volvo
Audi
Volvo
Audi
```

In [11]:

```
a=[1,2,3,4,5]
d=[2,2.3,4.5,6.7]
print(max(a))
print(min(a))
print(sorted(a))
len(a)
a.pop()
a.remove(3)
del a[2]
```

```
5
1
[1, 2, 3, 4, 5]
```

In [12]:

```
orderitem=[1,"Sam","Computer",75.50,True]
print(orderitem)
orderitem[2]="Laptop"
print(orderitem)
```

```
[1, 'Sam', 'Computer', 75.5, True]
[1, 'Sam', 'Laptop', 75.5, True]
```

In [15]:

```
orderitem=[1,"Sam","Computer",75.50,True]
orderitem.append('abc')
print(orderitem)
```

```
[1, 'Sam', 'Computer', 75.5, True, 'abc']
```

In [16]:

```
orderitem=[1,"Sam","Computer",75.50,True]
orderitem.extend(['MIT',2020])
print(orderitem)
```

```
[1, 'Sam', 'Computer', 75.5, True, 'MIT', 2020]
```

In [17]:

```
orderitem=[1,"Sam","Computer",75.50,True]
del orderitem[1]
print(orderitem)
```

```
[1, 'Computer', 75.5, True]
```

In [18]:

```
orderitem=[1,"Sam","Computer",75.50,True]
del orderitem[2:3]
print(orderitem)
```

```
[1, 'Sam', 75.5, True]
```

In [20]:

```
orderitem=[1,"Sam","Computer",75.50,True]
orderitem.remove('Computer')
print(orderitem)
```

```
[1, 'Sam', 75.5, True]
```

In [21]:

```
orderitem=[1,"Sam","Computer",75.50,True]
orderitem.pop(1)
print(orderitem)
```

```
[1, 'Computer', 75.5, True]
```

In [22]:

```
orderitem=[1,"Sam","Computer",75.50,True]
orderitem.clear()
print(orderitem)
```

```
[]
```

In [24]:

```
my_list = [1, 2, 3, 10, 30, 10]
print(len(my_list))
print(my_list.index(10))
print(my_list.count(10))
print(sorted(my_list))
my_list.sort(reverse=True)
print(my_list)
```

```
6
3
2
[1, 2, 3, 10, 10, 30]
[30, 10, 10, 3, 2, 1]
```

In [26]:

```
n = int(input("enter the size of array :"))
arr=[]
sum=0
for i in range(n):
  x= int(input("enter element: "))
  arr.append(x)
  sum=sum+arr[i]
print("Entered array is :", arr," and Average of elements is ",(sum/n))
```

```
enter the size of array :2
enter element: 1
enter element: 3
Entered array is : [1, 3]  and Average of elements is  2.0
```

In [37]:

```
phone1="IPHONE"
phone2="SAMSUNG"
phone3="ONEPLUS"
phones=["IPHONE","SAMSUNG","ONEPLUS"]
print(phones)
for x in phones:
 print(x)
for i in range (1,3):
 print(phones[i])
```

```
['IPHONE', 'SAMSUNG', 'ONEPLUS']
IPHONE
SAMSUNG
ONEPLUS
SAMSUNG
ONEPLUS
```

In [32]:

```
phones_list=["IPHONE","SAMSUNG","ONEPLUS"]
phones_list.append('REDMI')
print(phones_list)
```

```
['IPHONE', 'SAMSUNG', 'ONEPLUS', 'REDMI']
```

In [35]:

```
phones_list=["IPHONE","SAMSUNG","ONEPLUS"]
phones_list.extend(['NOTHING',2022])
print(phones_list)
```

```
['IPHONE', 'SAMSUNG', 'ONEPLUS', 'NOTHING', 2022]
```

In [34]:

```
phones_list=["IPHONE","SAMSUNG","ONEPLUS"]
phones_list.clear()
print(phones_list)
```

```
[]
```

Devanshu Surana
1032210755
PC - 12
Batch - C1

20/11/22

Python Programming
Lab Assignment - 4

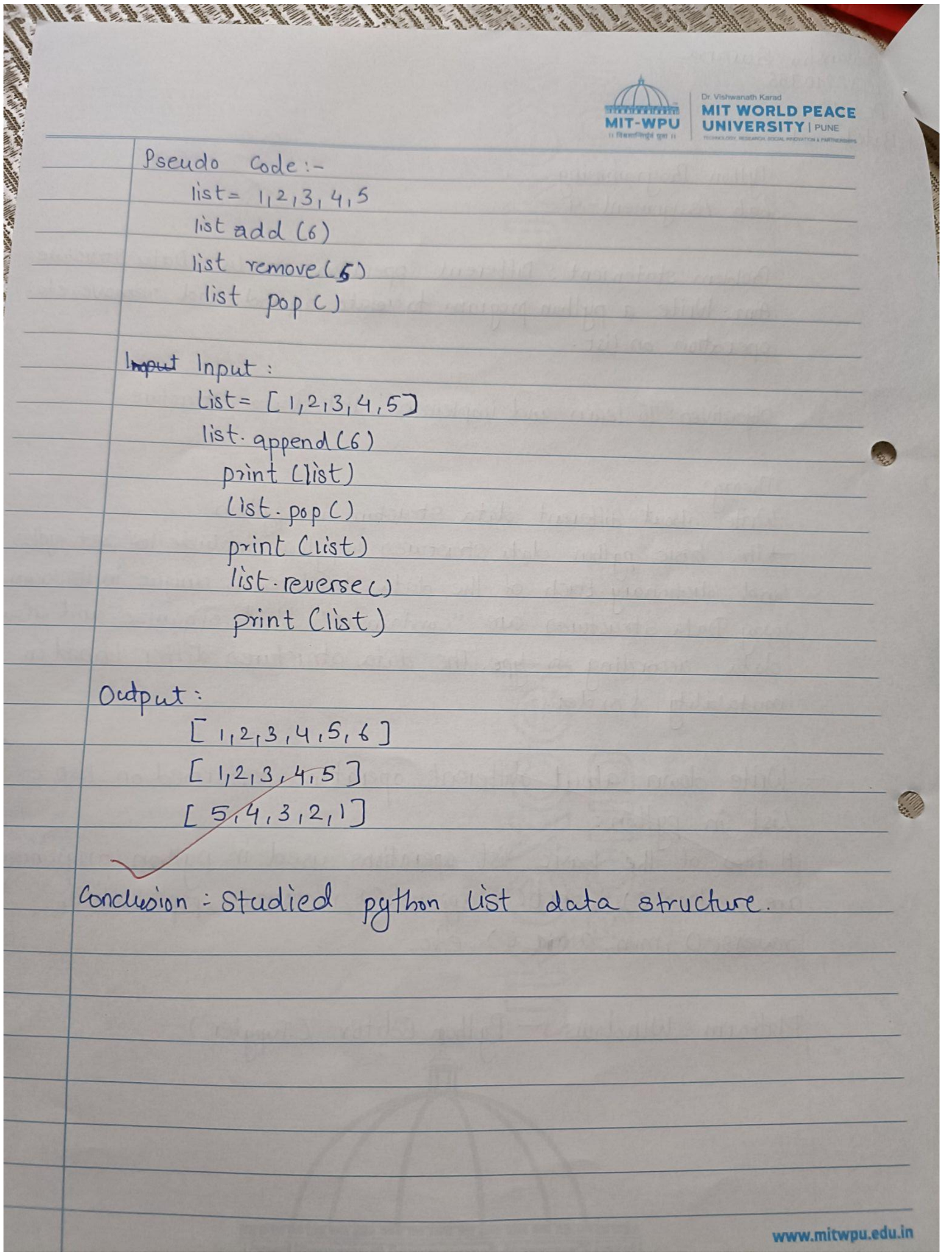
Problem statement : Different operations on list Data Structure
Aim : Write a python program to create, append and remove, etc. operation on list.

Objectives : To learn and implement list Data Structure.

Theory :
Write about different data structure in python.
→ The basic python data structures in python include list, set, tuples and dictionary. Each of the data structures is unique in its own way. Data structures are "containers" that organize and group data according to type. The data structures differ based on mutability & order.

- Write down about different operations performed on List.
List in python :
A few of the basic list operations used in python programming are extend(), insert(), append(), remove(), pop(), slice, reverse(), min, max(), etc.

Platform : Windows - Python Editor (Jupyter)



Pseudo Code:-

```
    list= 1,2,3,4,5
    list add (6)
    list remove(5)
    list pop ()
```

~~Input~~ Input :

```
    List= [1,2,3,4,5]
    list. append(6)
      print (list)
      (list. pop ()
      print (List)
      list. reverse()
      print (list)
```

Output :

```
    [1,2,3,4,5,6]
    [1,2,3,4,5]
    [5,4,3,2,1]
```

Conclusion : Studied python list data structure.

FAQ's

Ans 1) Unlike Strings, lists are mutable. This means we can change an item in a list by acrossing it directly as part of the assignment statement.

For eg: List = [1,2,3]
list.clear()

It will return an empty list.

Ans 2) append() adds a single element to the end of the list while extend() can add multiple individuals elements to the end of the list.

Ans 3) Python list don't store values themselves. They store pointers to values stored else-where in memory. This allows lists to be mutable.

Ans 4) The remove() function removes the first matching value from the list. The pop() function is used to return the removed element from the list.