# 1.Manipulate using a list

In [ ]:

```
# 1)Add new elements at end of the list
list=['hi',2,'hello']  #list defined
list.append(4) #element added at last
list0=[1,2,5,8]
list.extend(list0) #multiple elements can be added at the end using extend
list
```

In [4]:

```
#2)Reverse the list
list.reverse()
list
```

Out[4]:

```
[4, 'hello', 2, 'hi']
```

In [5]:

```
#3)Display the same list multiplr times
list*10
```

Out[5]:

```
[4,
 'hello',
 2,
 'hi',
 4,
 'hello',
 2,
 'hi',
 4,
 'hello',
 2,
 'hi',
 4,
 'hello',
 2,
 'hi',
 4,
 'hello',
 2,
 'hi',
 4,
 'hello',
 2,
 'hi',
 4,
 'hello',
 2,
 'hi',
 4,
 'hello',
 2,
 'hi',
 4,
 'hello',
 2,
 'hi',
 4,
 'hello',
 2,
 'hi']
```

In [8]:

```
#4)concatenation of two lists

list1=[1,'ok','python']
list1+list
```

Out[8]:

```
[1, 'ok', 'python', 4, 'hello', 2, 'hi']
```

In [15]:

```
#5)SORTING in ascending order
list2=[1,2,5,8,88,0,6,-7]
list2.sort() #list sorted in ascending order
list3=['praveen','pavan','Mphasis']
list3.sort()
list3
```

Out[15]:

```
['Mphasis', 'pavan', 'praveen']
```

# 2. Write a python program to do in the tuples

In [19]:

```
#1 Manipulate using tuples
#2 add new elements to the end of the tuple

tup=(1,'hi','hello')
tup1=(2,'Mphasis')
tup+tup1    #new element added at the end of the tuple
```

Out[19]:

```
(1, 'hi', 'hello', 2, 'Mphasis')
```

In [20]:

```
#3
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[20], line 1
----> 1 tup.revese()

AttributeError: 'tuple' object has no attribute 'revese'
```

In [22]:

```
#4 to display elements of the tuple many time
tup1*2
```

Out[22]:

```
(2, 'Mphasis', 2, 'Mphasis')
```

In [23]:

```
#5 To concatenate two tuples
tup3=tup+tup1
tup3
```

Out[23]:

```
(1, 'hi', 'hello', 2, 'Mphasis')
```

# 3.Write a python program to implement the folloeing using list

In [25]:

```
#1.Create a list with integers
lst=[1,2,3,4,5,6,7,8,9,1,0]
lst
```

Out[25]:

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 0]
```

In [27]:

```
#2.How to display last number in the list
lst.pop()
```

Out[27]:

```
0
```

In [29]:

```
#3.Command to displaying the values from th elist [0:4]
lst[0:4]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[29], line 2
      1 #3Command to displaying the values from th elist [0:4]
----> 2 lst.index[0]

TypeError: 'builtin_function_or_method' object is not subscriptable
```

In [33]:

```
#4.Command to displaying the values from th elist [2:]
lst[2:]
```

Out[33]:

```
[3, 4, 5, 6, 7, 8, 9, 1]
```

In [34]:

```
#5.Command to displaying the values from th elist [:6]
lst[:6]
```

Out[34]:

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

In [32]:

```
lst
```

Out[32]:

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 1]
```

# 4.Write a python Program

In [35]:

```
tuple1=(10,50,20,40,30)
tuple1
```

Out[35]:

```
(10, 50, 20, 40, 30)
```

In [36]:

```
#1. To display elements 10 and 50 from tuple1
tuple1[:2]
```

Out[36]:

```
(10, 50)
```

In [37]:

```
#2.to display length of tuple1
len(tuple1)
```

Out[37]:

```
5
```

In [38]:

```
#3.T0 find minimum elememnt from tuple1
min(tuple1)
```

Out[38]:

```
10
```

In [39]:

```
max(tuple1)
```

Out[39]:

```
50
```

In [40]:

```
#4.To add all elements in the tuple1
tuple1+tuple1
```

Out[40]:

```
(10, 50, 20, 40, 30, 10, 50, 20, 40, 30)
```

```
In [43]: #5.To display the same tuple1 multiple times
         tuple1*2
```

```
Out[43]: (10, 50, 20, 40, 30, 10, 50, 20, 40, 30)
```

# 5.Write a Python Program

```
In [44]: #1.To calculate the length of a string
         str='Welcome to Python'
         len(str)
```

```
Out[44]: 17
```

```
In [47]: #2.To reverse words in string
         str[::-1]
```

```
Out[47]: 'nohtyP ot emocleW'
```

```
In [48]: #3.To display the same string multiple times
         str*2
```

```
Out[48]: 'Welcome to PythonWelcome to Python'
```

```
In [50]: #4.to concatenate two stings
         s1='hi'
         s2=' hello'
         s1+s2
```

```
Out[50]: 'hi hello'
```

```
In [52]: #5. Display India from South India
         str1='South India'
         str1[6:]
```

```
Out[52]: 'India'
```

# 6.Perform the following

```
In [79]: #1.Create the Dictionary
         Dictionary={'name':'Praveen','Age':35,'ID':6365}
         Dictionary
```

```
Out[79]: {'name': 'Praveen', 'Age': 35, 'ID': 6365}
```

```
In [57]: #2. Accessing the values and keys in Dictionary
         Dictionary.get('Age')
```

```
Out[57]: 35
```

```
In [76]: #3.Updating dictionary usng function
         Dictionary.update({'Phone':817542333})
```

```
Dictionary
```

Out[76]:

```
{'name': 'Praveen', 'Age': 35, 'ID': 6365, 'Phone': 817542333}
```

In [80]:

```
#4.Clear and delete the dictionary values
Dictionary.pop('Age');
Dictionary
del Dictionary
Dictionary
#Dictionary.clear()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[80], line 5
      3 Dictionary
      4 del Dictionary
----> 5 Dictionary

NameError: name 'Dictionary' is not defined
```

# 7.Pythom progrm to insert a number to any position in a list

In [84]:

```
list.insert(3,2)
list
```

Out[84]:

```
['hi', 2, 'hello', 2, '2', '2', 4, 1, 2, 5, 8]
```

# 8.Pythom progrm to delete an element from a list by index

In [85]:

```
list.remove(2)
list
```

Out[85]:

```
['hi', 'hello', 2, '2', '2', 4, 1, 2, 5, 8]
```

# 9.Write a program to display a number from 1 to 100.

In [92]:

```
for i in range (1,101) :
    print(i)
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
```

```
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

# 10.Write a python program to find sum of all items in a tuple

In [110…

```
tuple=(1,2,3,4,5)
n=len(tuple)
sum=0;
for i in tuple:
    sum=sum+i

sum
```

Out[110]:

```
15
```

# 11. Create a dictionary containing three lambda functions square,cube,sqrt

In [15]:

```
import math

dict={'square':'lambda x:x**2','cube':'lambda x:x**3','st':'lambda x:math.sqrt(x)'}
print('enter the value for square')
x=int(input())
print('enter the value for cube')
y=int(input())
print('enter the value for square root')
z=int(input())
print((cube(y))+(square(x))+(st(z)))
```

```
enter the value for square
2
enter the value for cube
2
enter the value for square root
4
14.0
```

# 12.A list of word is given.Find the words from the list that have their second character in uppercase.

In [128…

```
ls=['hEllo','Dear','hOw','ARe','You']
n=len(ls)
for i in range(0,n):
    if(ls[i][1].isupper()):
        print(ls[i])
```

```
hEllo
hOw
ARe
```

# 13. A dictionary of names and their weights on earth is given. Find how much they will weigh on the moon. (Use map and lambda functions)

Formula: wMoon = (wEarth * GMoon) / GEarth i) #Weight of people in kg WeightOnEarth = {'John':45, 'Shelly':65, 'Marry':35}

ii)# Gravitational force on the Moon: 1.622 m/s2 GMoon 1.622

iii)# Gravitational force on the Earth: 9.81 m/s2 GEarth 9.81

In [20]:

```
WeightOnEarth = {'John': 45, 'Shelly': 65, 'Marry': 35}
GMoon = 1.622
GEarth = 9.81
calculate_weight_on_moon = lambda weight_on_earth: (weight_on_earth * GMoon) / GEarth
WeightOnMoon = dict(map(lambda person: (person, calculate_weight_on_moon(WeightOnEarth
for person, weight_moon in WeightOnMoon.items():
    print(f"{person}'s weight on the Moon: {weight_moon} kg")
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[20], line 5
      3 GEarth = 9.81
      4 calculate_weight_on_moon = lambda weight_on_earth: (weight_on_earth * GMoon)
/ GEarth
----> 5 WeightOnMoon = dict(map(lambda person: (person, calculate_weight_on_moon(Weig
htOnEarth[person])), WeightOnEarth))
      6 for person, weight_moon in WeightOnMoon.items():
      7     print(f"{person}'s weight on the Moon: {weight_moon} kg")

TypeError: 'dict' object is not callable
```

In [ ]: