```
list1=["aman","222",777,400.4]
tup=tuple(list1)
print(tup)
```

```
('aman', '222', 777, 400.4)
```

```
info={'Roll no':'CB.EN.U4CSE20244','First Name':'Nith','Last Name':'M','DOB':'22/02/2002'}
for key,value in info.items():
  print(key,':',value)
```

```
Roll no : CB.EN.U4CSE20244
First Name : Nith
Last Name : M
DOB : 22/02/2002
```

+ Code + Text

```
import numpy as np
np.random.seed(0)
arr=np.random.randint(0,50,size=(2,5))
print(arr)
```

```
[[44 47  0  3  3]
 [39  9 19 21 36]]
```

```
x=np.random.uniform(0.0,1.0)
print(x)
print(abs(-1234))
```

```
0.2975346065444723
1234
```

```
x=np.random.randint(20,91)
print(x)
```

```
57
```

```
def retlist(list1):
  return list1[0:3]
list1=[11,12,13,14,15]
newlist=retlist(list1)
print(newlist)
np.random.shuffle(newlist)
print("After Shuffle :",newlist)
print(np.random.choice(list1))
```

```
[11, 12, 13]
After Shuffle : [11, 13, 12]
14
```

#7

```
def check(x):
  if x>0:
    print("x is positive")
  elif x<0:
    print("x is negetive")
  else:
    print("x is zero")
check(5)
check(-5)
check(0)
x=0
print("While loop")
while(x<10):
  x+=1
  if x==5:
    print("Continue when x =5")
    continue
  if x==9:
    print("break when x=9")
    break
  print(x)
print("For loop")

for i in range(10,20):
  if i==15:
    print("Continue when i=15")
    continue
  if i==18:
    print("break when i=18")
    break
  print(i)
```

```
x is positive
x is negetive
x is zero
While loop
1
2
3
4
Continue when x =5
6
7
8
break when x=9
For loop
10
```

```
11
12
13
14
Continue when i=15
16
17
break when i=18
```

```
string="The King is Dead, Long Live The King"
print(string)
print(string[0])
print(string[3:6])
print(string[3:])
print(string.split())
```

```
The King is Dead, Long Live The King
T
 Ki
 King is Dead, Long Live The King
['The', 'King', 'is', 'Dead,', 'Long', 'Live', 'The', 'King']
```

```
list2=[1,4,5,8]
arr=np.array(list2)
print("The first and second element of arr",arr[0],arr[1])
```

```
The first and second element of arr 1 4
```

```
twod=np.array([[1.,2.,3.],[4.,5.,6.]])
print("The 2D array :")
print(twod)
```

```
The 2D array :
[[1. 2. 3.]
 [4. 5. 6.]]
```

```
print("Second Row",twod[1,:])
print("Second Col",twod[:,1])
print("Length of the array :",len(twod))
```

```
Second Row [4. 5. 6.]
Second Col [2. 5.]
Length of the array : 2
```

```
list3=[]
for i in range(10):
  list3.append(np.random.randint(0,100))
oned=np.array(list3)
print("1D array :",oned)
```

```
twod=oned.reshape(5,2)
print("2D array :\n",twod)
print("Transpose :\n",twod.T)
```

```
1D array : [43 58 23 59  2 98 62 35 94 67]
2D array :
 [[43 58]
 [23 59]
 [ 2 98]
 [62 35]
 [94 67]]
Transpose :
 [[43 23  2 62 94]
 [58 59 98 35 67]]
```

```
l1=[1,2,3]
l2=[4,5,6]
l3=[7,8,9]
newarr=np.array([l1,l2,l3])
print(newarr)
```

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

```
arr=np.array([1,2,3])
for i in range(len(arr)):
  arr[i]=9
print(arr)
```

```
[9 9 9]
```

```
arr=np.array([2,4,5])
sum=0
for i in arr:
  sum+=i
print(sum)
```

```
11
```

```
arr=np.array([77,74,75])
prod=1
for i in arr:
  prod*=i
print(prod)
```

```
427350
```

```
arr=np.array([2,1,9,1,2,22,1,1,4,55,6,2,56])
print("Mean : ",np.mean(arr))
print("Varaince : ",np.var(arr))
print("Max :",np.max(arr))
print("Min : ",np.min(arr))
print("Argmin : ",np.argmin(arr))
print("Median :",np.median(arr))
print("Argmax : ",np.argmax(arr))
print("Standard Deviation : ",np.std(arr))
```

```
Mean :  12.461538461538462
Varaince :  367.32544378698225
Max : 56
Min :  1
Argmin :  1
Median : 2.0
Argmax :  12
Standard Deviation :  19.165736192147232
```

```
arr=np.array([1, 1, 7, 5, 5, 5, 4])
uarr=np.array(np.unique(arr))
uarr.sort()
print(uarr)
```

```
[1 4 5 7]
```

```
m1=np.array([1,2,3])
m2=np.array([4,5,6])
print(np.dot(m1,m2))
```

```
32
```

```
a=np.array([[2,12],[2,-5]])
val,vec=np.linalg.eig(a)
print("Eigen value :",val)
print("Eigen Vector :\n",vec)
```

```
Eigen value : [ 4.52079729 -7.52079729]
Eigen Vector :
 [[ 0.97864042 -0.78338465]
 [ 0.20557951  0.6215372 ]]
```

```
a1=np.array([11,22,11,31])
a2=np.array([25,33,31,28])
corrcoef=np.corrcoef(a1,a2)
print(corrcoef)
```

```
[[1.         0.15997968]
 [0.15997968 1.        ]]
```

Colab paid products - Cancel contracts here

✓ 0s completed at 10:22 PM