Import numpy library.

```
import numpy as np
```

```
list = [1,2,3,4,5]
print(list)
```

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

```
x=np.array(list)
```

Create 1D array with 5 elements.

```
x=np.array([1,2,45,4,3])
print(x)
```

```
[ 1  2 45  4  3]
```

Create 1D array with 5 elements random value. (by default values between 0 to 1)

```
x=np.random.rand(5)
print(x)
```

```
[0.55319637 0.65697631 0.46315282 0.81382293 0.034081  ]
```

Generate random value from range 5 to 15 with 1D array size 10.

```
x=np.random.randint(5,16,5)
print(x)
```

```
[14 14 11 11  9]
```

Create 2D array - 4 by 3 size.

```
y=np.array([[1,2,3],[4,5,6],[7,8,9],[10,11,12]])
print(y)
```

```
[[ 1  2  3]
 [ 4  5  6]
 [ 7  8  9]
 [10 11 12]]
```

Create 2D array with size 3 by 3. Elements values are random number(by default values between 0 to 1)

```
y=np.random.rand(3,3)
print(y)
```

```
[[0.55131192 0.84550054 0.2345885 ]
 [0.93192845 0.0100533  0.80713761]
 [0.1174903  0.64244841 0.57135036]]
```

Generate random value from range 5 to 15 with 2D array size 3 by 3.

```
n=np.random.randint(5,16,(3,3))
print(n)
```

```
[[15 15 12]
 [11 15  8]
 [ 6 15 14]]
```

Create 3D array with size 2 * 3 * 2.

```
y=np.array([[[1,2,3],[4,5,6]],[[4,3,3],[7,8,2]]])
print(y)
```

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

 [[4 3 3]
  [7 8 2]]]
```

Initialize 5 by 5 array with all values are zeros.

```
a=np.zeros((5,5),dtype=int)
print(a)
```

```
[[0 0 0 0 0]
 [0 0 0 0 0]
 [0 0 0 0 0]
 [0 0 0 0 0]
 [0 0 0 0 0]]
```

Initialize 5 by 5 array with all values are ones.

```
a=np.ones((5,5),dtype=int)
print(a)
```

```
[[1 1 1 1 1]
 [1 1 1 1 1]
 [1 1 1 1 1]
 [1 1 1 1 1]
 [1 1 1 1 1]]
```

Initialize 5 by 5 array with all values are particular values(consider 4).

```
a=np.full((5,5),4)
print(a)
```

```
[[4 4 4 4 4]
 [4 4 4 4 4]
 [4 4 4 4 4]
 [4 4 4 4 4]
 [4 4 4 4 4]]
```

Create indentity matrix with all diagonal values are 1 and rest all values are zeros.

```
a=np.eye(5,5,dtype=int)
print(a)
```

```
[[1 0 0 0 0]
 [0 1 0 0 0]
 [0 0 1 0 0]
 [0 0 0 1 0]
 [0 0 0 0 1]]
```

ATTRIBUTES

Find out shape of array.

```
print(a.shape)
print(x.shape)
```

```
(5, 5)
(5,)
```

Find dimension of array.

```
print(a.ndim)
print(x.ndim)
```

```
2
1
```

Find no. of elements of from above any array.

```
print(a.size)
print(x.size)
```

```
25
5
```

Find type of array.

```
a.dtype
x.dtype
```

```
dtype('int64')
```

Find maximum value from array.

```
b = np.max(n)
print(b)
```

```
15
```

Find minimum value from array.

```
b = np.min(n)
print(b)
```

```
6
```

Find average (mean) values from array.

```
#minmum value location
b = np.argmin(n)
print(b)

#maximum value location
b = np.argmax(n)
print(b)
print(n)
```

```
6
0
[[15 15 12]
 [11 15  8]
 [ 6 15 14]]
```

Find location of maximum value from array.

```
#maximum value location
b = np.argmax(n)
print(b)


print(n)
```

```
0
[[15 15 12]
 [11 15  8]
 [ 6 15 14]]
```

Create two 2D array of size 3 by 3 and perform four basic mathematical operations

```
a = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
b = np.array([[9, 8, 7], [6, 5, 4], [3, 2, 1]])


print(a)
print()
print(b)
print()

print(a+b)
print()

print(a-b)
print()

print(a/b)
print()

print(a*b)
print()
```

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

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

[[10 10 10]
 [10 10 10]
 [10 10 10]]

[[-8 -6 -4]
 [-2  0  2]
 [ 4  6  8]]

[[0.11111111 0.25       0.42857143]
 [0.66666667 1.         1.5       ]
 [2.33333333 4.         9.        ]]

[[ 9 16 21]
 [24 25 24]
 [21 16  9]]
```

Print index 2 and -2 value for 1D array. (indexing).

```
x=np.array([14,15,11,8,8])
print(x)
print(x[2]) #11
print(x[-2]) #8
```

```
[14 15 11  8  8]
11
8
```

Perform indexing operations with 2D array.

```
print(n)

print(n[1,2])  #   n[1,2]  where 1 is row and 2 is column
print(n[0,0])
```

```
[[15 15 12]
 [11 15  8]
 [ 6 15 14]]
8
15
```

Print value of index 2 to 5 (slicing)

```
x=np.array([14,15,11,8,8,10,12])
print(x[2:6])
```

```
[11  8  8 10]
```

Print from index 5 to all

```
x=np.array([14,15,11,8,8,10,12,13,14])
print(x[5:])
```

```
[10 12 13 14]
```

Reassign value for particular index.

```
x=np.array([14,15,11,8,8,10,12,13,14])
print(x)
x[2]=20
print(x)
```

```
[14 15 11  8  8 10 12 13 14]
[14 15 20  8  8 10 12 13 14]
```

Extract elements from index 1 to 7 with a step of 2.

```
x = np.array([14, 15, 11, 8, 8, 10, 12, 13, 14])
print(x[1:8:2])
```

```
[15  8 10 13]
```

Print Reverse the array.

```
x = np.array([14, 15, 11, 8, 8, 10, 12, 13, 14])
print(x[::-1])
```

```
[14 13 12 10  8  8 11 15 14]
```