day2-630

February 14, 2024

[11]:
```
#To convert list to array, print array and length of array
import numpy
list_1=[1,2,3,4,5,6,7,8,9,10]
print(type(list_1)) #to print type of list_1
array_1=numpy.array(list_1) #To convert list to array
print(array_1) #To print array
print(type(array_1)) #To print array_1 type
print(len(array_1)) #To print array length
```

```
<class 'list'>
[ 1  2  3  4  5  6  7  8  9 10]
<class 'numpy.ndarray'>
10
```

[5]:
```
#To print diagonal matrix method 1
numpy.eye(5)
```

[5]:
```
array([[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.]])
```

[6]:
```
#To print diagonal matrix method 2
numpy.identity(5)
```

[6]:
```
array([[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.]])
```

[8]:
```
#To print m X n diagonal matrix
numpy.eye(3,4)
```

[8]:
```
array([[1., 0., 0., 0.],
       [0., 1., 0., 0.],
       [0., 0., 1., 0.]])
```

[26]:
```
#To reshape an existing array
list_1=range(25)
array_1=numpy.array(list_1)
array_1=array_1.reshape(5,5)
print(array_1)
```

```
[[ 0  1  2  3  4]
 [ 5  6  7  8  9]
 [10 11 12 13 14]
 [15 16 17 18 19]
 [20 21 22 23 24]]
```

[27]:
```
#Slicing of array
print(array_1[3,1:7])
```

```
[16 17 18 19]
```

[28]:
```
#To print dimensions of matrix
array_1.shape
```

[28]:
```
(5, 5)
```

[35]:
```
#To print sub-matrix
array_1[1:3,1:3]
```

[35]:
```
array([[ 6,  7],
       [11, 12]])
```

[22]:
```
#To print complex matrix
numpy.identity(5,dtype=complex)
```

[22]:
```
array([[1.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j],
       [0.+0.j, 1.+0.j, 0.+0.j, 0.+0.j, 0.+0.j],
       [0.+0.j, 0.+0.j, 1.+0.j, 0.+0.j, 0.+0.j],
       [0.+0.j, 0.+0.j, 0.+0.j, 1.+0.j, 0.+0.j],
       [0.+0.j, 0.+0.j, 0.+0.j, 0.+0.j, 1.+0.j]])
```

[34]:
```
#To print sub-matrix
array_1[-4:-2,-4:-2]
```

[34]:
```
array([[ 6,  7],
       [11, 12]])
```

[37]:
```
#To find mean
list_1=[1,2,3,4,5,6,7,8]
array_1=numpy.array(list_1)
print(array_1.mean())
```

```
4.5
```

[39]:
```
#To find median
print(numpy.median(array_1))
```

```
4.5
```

[55]:
```
#To find sum
print(sum(array_1))
```

```
[100 105 110 115 120 125 130 135 140 145]
```

[56]:
```
#To find variance
numpy.var(array_1)
```

[56]: 208.25

[58]:
```
#To find variance
a=range(9)
ar=numpy.array(a)
numpy.var(ar,axis=0)
```

[58]: 6.666666666666667

[61]:
```
#To find variance
a=range(9)
ar=numpy.array(a)
numpy.var(ar,axis=0)
```

[61]: 6.666666666666667

[43]:
```
#To find standard deviation
numpy.std(array_1)
```

[43]: 2.29128784747792

[46]:
```
#To reshape an existing array
list_1=range(50)
array_1=numpy.array(list_1)
array_1=array_1.reshape(5,10)
print(array_1)
```

```
[[ 0  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]]
```

[71]:
```
numpy.linspace(1,20,5,retstep=True)
```

[71]:
```
(array([ 1.  ,  5.75, 10.5 , 15.25, 20.  ]), 4.75)
```

[75]:
```
numpy.random.randint(10)
```

[75]:
```
5
```

[84]:
```
numpy.random.rand(4)
```

[84]:
```
array([0.58833606, 0.98541566, 0.08610926, 0.46742413])
```

[ ]:
```

```