What is Numpy

Numpy is a Python library that supports multi-dimensional arrays and matrix. It also provides many basic and high-level mathematical functions that can be applied on these multi-dimensional arrays and matrices with less code footprint.

Numpy Datatypes:

1. Integers(uint8,uint16,unint32,uint64)
2. Float(float16,float32...)
3. Boolean(True:1,False:0)

Numpy Array

In [7]:
```
import numpy as np

a  = [5,9,12]

a = np.array(a)
```

In [131]:
```
print(np.__version__)
```

```
1.19.5
```

In [9]:
```
np.array([10,13,17])

```

Out[9]:
```
array([10, 13, 17])
```

arange

In [15]:
```
1 np.arange(0,10,1)
```

Out[15]:
```
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

linspace

In [21]:
```
1 np.linspace(1,10,10)
```

Out[21]:
```
array([ 1.,  2.,  3.,  4.,  5.,  6.,  7.,  8.,  9., 10.])
```

In [23]:
```
1 (10-1)/(10-1)
```

Out[23]:
```
1.0
```

In [25]:
```
1 np.linspace(1,15,7)
```

Out[25]:
```
array([ 1.        ,  3.33333333,  5.66666667,  8.        , 10.33333333,
       12.66666667, 15.        ])
```

In [24]:
```
1 14/6
```

Out[24]:
```
2.3333333333333335
```

nd Numpy arrays

In [51]:
```
1 a = np.arange(0,12,1)
```

In [52]:
```
1 a
```

Out[52]:
```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11])
```

In [57]:
```
1 a = a.reshape(3,2,2)
```

In [58]:
```
print('dims : ', a.ndim )
print('Size : ', a.size )
print('Shape : ', a.shape )

```

```
dims :  3
Size :  12
Shape :  (3, 2, 2)
```

In [70]:
```
a
```

Out[70]:
```
array([[[ 0,  1],
        [ 2,  3]],

       [[ 4,  5],
        [ 6,  7]],

       [[ 8,  9],
        [10, 11]]])
```

In [67]:
```
b = np.array([2,3,5,3,2,3])
b = b.reshape(3,2)
b.shape
```

Out[67]:
```
(3, 2)
```

In [69]:
```
b.reshape(3,2)
```

Out[69]:
```
array([[2, 3],
       [5, 3],
       [2, 3]])
```

Indexing arrays

In [75]:
```
a  = [5,9,12]

a = np.array(a)
```

In [78]:
```
a[-1]
```

Out[78]: 12

In [79]:
```
b = np.array([2,3,5,3,2,3])
b = b.reshape(3,2)
b.shape
```

Out[79]: (3, 2)

In [85]:
```
b
```

Out[85]:
```
array([[2, 3],
       [5, 3],
       [2, 3]])
```

In [89]:
```
b[1,-1]
```

Out[89]: 3

In [91]:
```
a = np.arange(0,12,1)
a = a.reshape(3,2,2)
a
```

Out[91]:
```
array([[[ 0,  1],
        [ 2,  3]],

       [[ 4,  5],
        [ 6,  7]],

       [[ 8,  9],
        [10, 11]]])
```

In [95]:
```
a[1,1,0]
```

Out[95]: 6

In [99]:
```
a  = [5,9,12]

a = np.array(a)

b  = np.array([10,10,10])
```

In [97]:
```
lst1 = [1,2,3]
lst2 = [3,5,6]
```

In [98]:
```
lst1 + lst2
```

Out[98]: [1, 2, 3, 3, 5, 6]

In [104]:
```
a+b # addition of corresponding a and b values
```

Out[104]: array([15, 19, 22])

In [105]:
```
a + 4 # addition of 4 to every element in a
```

Out[105]: array([9, 13, 16])

In [106]:
```
np.sum(b)
```

Out[106]: 30

In [108]:
```
np.mean(a)
np.mean(b)
```

Out[108]: 10.0

Random Generator

In [110]:
```
from numpy import random
```

In [118]:
```
random.randint(0,100,size = (10))
```

Out[118]:
```
array([44, 18, 62, 70, 62, 96, 97, 18, 12, 55])
```

In [117]:
```
random.choice([3,27,9,33,93],size = (2,5))
```

Out[117]:
```
array([[ 9, 27,  3, 27, 27],
       [33, 33,  3, 33,  3]])
```

In [124]:
```
random.rand(3,4)
```

Out[124]:
```
array([[5.06214119e-02, 9.26912562e-01, 1.45387284e-01, 8.58833704e-01],
       [1.41037953e-01, 1.06430124e-01, 9.45585599e-01, 9.26321340e-01],
       [2.17229717e-01, 2.61338745e-01, 7.97841146e-04, 8.71253871e-01]])
```

Generating standard Arrays: arrays with 0 and 1

In [127]:
```
np.zeros((2,2))
```

Out[127]:
```
array([[0., 0.],
       [0., 0.]])
```

In [129]:
```
a = np.ones((2,2))
```

In [130]:
```
a
```

Out[130]:
```
array([[1., 1.],
       [1., 1.]])
```

In [ ]:
```

```