numpy Tutorial

In [1]:
```
import numpy as np
```

In [2]:
```
myarr=np.array([[1,2,4,6,7]], np.int64)
```

In [3]:
```
myarr
```

Out[3]:
```
array([[1, 2, 4, 6, 7]], dtype=int64)
```

In [4]:
```
myarr[0,2]
```

Out[4]:
```
4
```

In [5]:
```
myarr.shape
```

Out[5]:
```
(1, 5)
```

In [6]:
```
myarr.dtype
```

Out[6]:
```
dtype('int64')
```

In [7]:
```
myarr[0,2]=43
```

In [8]:
```
myarr
```

Out[8]:
```
array([[ 1,  2, 43,  6,  7]], dtype=int64)
```

In [9]:
```
listarray=np.array([[1,6,7,8,4],[2,6,9,5,3],[2,7,5,4,6]])
```

In [10]:
```
listarray
```

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

In [11]:
```
listarray.dtype
```

Out[11]:
```
dtype('int32')
```

In [12]:
```
listarray.shape
```

Out[12]:
```
(3, 5)
```

In [13]:
```
listarray.size
```

Out[13]:
```
15
```

In [14]:
```
zeros=np.zeros((2,6))
```

In [15]:
```
zeros
```

Out[15]:
```
array([[0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0.]])
```

In [16]:
```
rng=np.arange(15)
```

In [17]:
```
rng
```

Out[17]:
```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14])
```

In [20]:
```
lspace=np.linspace(2,4,10)
```

In [21]:
```
lspace
```

Out[21]:
```
array([2.        , 2.22222222, 2.44444444, 2.66666667, 2.88888889,
       3.11111111, 3.33333333, 3.55555556, 3.77777778, 4.        ])
```

In [24]:
```
emp=np.empty((4,6))
```

In [ ]:
```
emp
```

In [25]:
```
emp
```

Out[25]:
```
array([[4.67296746e-307, 1.69121096e-306, 1.15711786e-306,
        2.22523004e-307, 1.78020440e-306, 7.56587585e-307],
       [1.37961302e-306, 6.23053614e-307, 4.45065276e-308,
        8.90102881e-307, 3.33776697e-307, 9.79054228e-307],
       [9.34598926e-307, 2.22522597e-306, 1.33511969e-306,
        8.45593934e-307, 7.56593017e-307, 8.01097889e-307],
       [2.22522868e-306, 1.33511562e-306, 1.24611402e-306,
        1.60220393e-306, 8.34423493e-308, 2.29179042e-312]])
```

In [26]:
```
emp_like=np.empty_like(lspace)
```

In [27]:
```
emp_like
```

Out[27]:
```
array([2.        , 2.22222222, 2.44444444, 2.66666667, 2.88888889,
       3.11111111, 3.33333333, 3.55555556, 3.77777778, 4.        ])
```

In [28]:
```
ide=np.identity(45)
```

In [29]:
```
ide
```

Out[29]:
```
array([[1., 0., 0., ..., 0., 0., 0.],
       [0., 1., 0., ..., 0., 0., 0.],
       [0., 0., 1., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 1., 0., 0.],
       [0., 0., 0., ..., 0., 1., 0.],
       [0., 0., 0., ..., 0., 0., 1.]])
```

In [33]:
```
arr=np.arange(60)
```

In [34]:
```
arr
```

Out[34]:
```
array([ 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, 50,
       51, 52, 53, 54, 55, 56, 57, 58, 59])
```

In [35]:
```
arr.reshape(6,10)
```

Out[35]:
```
array([[ 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],
       [50, 51, 52, 53, 54, 55, 56, 57, 58, 59]])
```

In [36]:
```
x=[[1,2,3],[4,5,6],[7,8,9]]
```

In [37]:
```
ar=np.array(x)
```

In [38]:
```
ar
```

Out[38]:
```
array([[1, 2, 3],
       [4, 5, 6],
       [7, 8, 9]])
```

In [39]:
```
ar.sum(axis=0)
```

Out[39]:
```
array([12, 15, 18])
```

In [40]:
```
ar.sum(axis=1)
```

Out[40]:
```
array([ 6, 15, 24])
```

In [41]:
```
ar.T
```

Out[41]:
```
array([[1, 4, 7],
       [2, 5, 8],
       [3, 6, 9]])
```

In [42]:
```
ar.flat
```

Out[42]:
```
<numpy.flatiter at 0x1e1c2896b60>
```

In [43]:
```
for item in ar.flat:
    print(item)
```

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

In [44]:
```
ar.ndim
```

Out[44]:
```
2
```

In [45]:
```
one=np.array([1,56,876,98])
```

In [46]:
```
one.argmax()
```

Out[46]:
```
2
```

In [47]:
```
one.argmin()
```

Out[47]:
```
0
```

In [48]:
```
ar.argmax(axis=0)
```

Out[48]:
```
array([2, 2, 2], dtype=int64)
```

In [49]:
```
ar.argsort(axis=0)
```

Out[49]:
```
array([[0, 0, 0],
       [1, 1, 1],
       [2, 2, 2]], dtype=int64)
```

In [50]:
```
ar
```

Out[50]:
```
array([[1, 2, 3],
       [4, 5, 6],
       [7, 8, 9]])
```

In [51]:
```
ar
```

Out[51]:
```
array([[1, 2, 3],
       [4, 5, 6],
       [7, 8, 9]])
```

In [59]:
```
ar.argsort(axis=1)
```

Out[59]:
```
array([[0, 1, 2],
       [0, 1, 2],
       [0, 1, 2]], dtype=int64)
```

In [60]:
```
arr2=np.array([[2,4,5],[4,5,6],[6,7,4]])
```

In [56]:
```
arr2
```

Out[56]:
```
array([[2, 4, 5],
       [4, 5, 6],
       [6, 7, 4]])
```

In [57]:
```
arr+arr2
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Input In [57], in <cell line: 1>()
----> 1 arr+arr2

ValueError: operands could not be broadcast together with shapes (60,) (3,3)
```

In [58]:
```
arr2.reshape(1,9)
```

Out[58]:
```
array([[2, 4, 5, 4, 5, 6, 6, 7, 4]])
```

In [61]:
```
arr2.reshape(3,3)
```

Out[61]:
```
array([[2, 4, 5],
       [4, 5, 6],
       [6, 7, 4]])
```

In [66]:
```
ar*arr2
```

Out[66]:
```
array([[ 2,  8, 15],
       [16, 25, 36],
       [42, 56, 36]])
```

In [67]:
```
np.sqrt(arr2)
```

Out[67]:
```
array([[1.41421356, 2.        , 2.23606798],
       [2.        , 2.23606798, 2.44948974],
       [2.44948974, 2.64575131, 2.        ]])
```

In [68]:
```

```

In [ ]:
```

```