

In [1]:

```
# Sorting Arrays
import numpy as np
```

In [2]:

#Sorting means putting elements in a ordered sequence.

```
a1 = np.array([6, 3, 2, 0, 4, 1, 5])
np.sort(a1)
```

Out[2]:

```
array([0, 1, 2, 3, 4, 5, 6])
```

In []:

In [3]:

```
a1 = np.array(['b', 'd', 'e', 'g', 'c', 'a', 'f'])
np.sort(a1)
```

Out[3]:

```
array(['a', 'b', 'c', 'd', 'e', 'f', 'g'], dtype='<U1')
```

In []:

In [4]:

```
a1 = np.array([True,False,False,True,True,True,False])
np.sort(a1)
```

Out[4]:

```
array([False, False, False,  True,  True,  True,  True])
```

In []:

In [5]:

```
# Sorting a 2-D Array
```

```
a1 = np.array([[3, 2, 4], [5, 0, 1]])  
np.sort(a1)
```

Out[5]:

```
array([[2, 3, 4],  
       [0, 1, 5]])
```

In []:

In []:

```
# Filtering Arrays
```

```
# A boolean index list is a list of booleans corresponding to indexes in the array.
```

```
# If the value at an index is True that element is contained in the filtered array,
```

```
# if the value at that index is False that element is excluded from the filtered array.
```

In [6]:

```
a2 = np.array([10, 20, 30, 40])
```

```
x = [True, False, True, False]
```

```
aa = a2[x]
```

```
aa
```

Out[6]:

```
array([10, 30])
```

In []:

In [7]:

```
a3 = np.array([10, 20, 30, 40])
```

```
res = a3 > 12
```

```
aa = a3[res]
```

```
aa
```

Out[7]:

```
array([20, 30, 40])
```

In []:

In [8]:

```
a3 = np.array([11, 20, 31, 40])  
res = a3 % 2 == 0  
aa = a3[res]  
aa
```

Out[8]:

```
array([20, 40])
```

In []:

In []:

In []:

```
# Statistical Functions
```

In [9]:

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

In [10]:

```
a
```

Out[10]:

```
array([[3, 7, 5],  
       [8, 4, 3],  
       [2, 4, 9]])
```

In [11]:

```
np.min(a)
```

Out[11]:

```
2
```

In [12]:

```
# 0 = check column wise  
np.amin(a,0)
```

Out[12]:

```
array([2, 4, 3])
```

In [13]:

```
# 1 = check row wise  
np.amin(a,1)
```

Out[13]:

```
array([3, 3, 2])
```

In []:

In [14]:

```
a
```

Out[14]:

```
array([[3, 7, 5],  
       [8, 4, 3],  
       [2, 4, 9]])
```

In [15]:

```
np.max(a)
```

Out[15]:

```
9
```

In [16]:

```
# 0 = check column wise  
np.amax(a,0)
```

Out[16]:

```
array([8, 7, 9])
```

In [17]:

```
# 1 = check row wise  
np.amax(a,1)
```

Out[17]:

```
array([7, 8, 9])
```

In []:

In []:

```
# numpy.ptp() function returns the range (maximum-minimum) of values along with axis
```

In [18]:

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

In [19]:

```
b
```

Out[19]:

```
array([[3, 7, 5],  
       [8, 4, 3],  
       [2, 4, 9]])
```

In [20]:

```
# subtraction of max value with min value from column wise  
np.ptp(b, axis = 0)
```

Out[20]:

```
array([6, 3, 6])
```

In [21]:

```
# subtraction of max value with min value from row wise  
np.ptp(b, axis = 1)
```

Out[21]:

```
array([4, 5, 7])
```

In []:

In []: