In [1]:

```
import numpy as np
```

Boolean

In [4]:

```
X = np.arange(25).reshape(5, 5)
```

In [6]:

```
print(X)
```

```
[[ 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]]
```

In [4]:

```
print(X[X > 10])
```

```
[11 12 13 14 15 16 17 18 19 20 21 22 23 24]
```

In [5]:

```
print(X[(X > 10) & (X < 17)])
```

```
[11 12 13 14 15 16]
```

In [6]:

```
X[(X > 10) & (X < 17)] = -1
print(X)
```

```
[[ 0  1  2  3  4]
 [ 5  6  7  8  9]
 [10 -1 -1 -1 -1]
 [-1 -1 17 18 19]
 [20 21 22 23 24]]
```

SET

In [7]:

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

In [8]:

```
print('The elements that are both in x and y:', np.intersect1d(x,y))
```

```
The elements that are both in x and y: [2 4]
```

In [9]:

```
print('The elements that are in x that are not in y:', np.setdiff1d(x,y
))
```

```
The elements that are in x that are not in y: [1 3 5]
```

In [10]:

```
print('All the elements of x and y:',np.union1d(x,y))
```

```
All the elements of x and y: [1 2 3 4 5 7 8]
```

Sort

In [9]:

```
print('Sorted x (out of place):', np.sort(y))
print(y)
```

```
Sorted x (out of place): [2 4 6 7 8]
[6 7 2 8 4]
```

In [11]:

```
# We sort x but only keep the unique elements in x
print(np.sort(np.unique(y)))
```

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

In [13]:

```
y.sort()
print(y)
```

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

rank 2 --------sort()----------------------

In [17]:
```
X = np.random.randint(1,11,size=(5,5))
print(X)
```

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

In [19]:
```
#Column wise sorting
print(np.sort(X, axis = 0))
```

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