Accessing array element using index of 1D arrays

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

In [ ]:
```
x=np.arange(1,17)
print("Orignal = \n",x)
x[1]=13
print("Changed = \n",x)
```

```
Orignal =
 [ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16]
Changed =
 [ 1 13  3  4  5  6  7  8  9 10 11 12 13 14 15 16]
```

Accessing array element using index of 2D arrays

In [ ]:
```
x=np.arange(1,10).reshape(3,3)
print("Orignal = \n",x)
x[(1,0)]=13
print("Changed = \n",x)
```

```
Orignal =
 [[1 2 3]
 [4 5 6]
 [7 8 9]]
Changed =
 [[ 1  2  3]
 [13  5  6]
 [ 7  8  9]]
```

Deleteing array elements using delete function of 1-D Arrays

In [ ]:
```
# ********* 1-D Arrays********
x=np.arange(1,10)
print(f"{x}")
x=np.delete(x,[0])
print(f"{x}")
x=np.delete(x,[0,3])
print(f"{x}")
# ********* 2-D Arrays***
```

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

Deleteing array elements using delete function of 2-D Arrays

In [ ]:
```
# x=np.delete(<array-variable>,<row.column>,axis=0/1)
# axis=0 means row
# axis=1 means column
x=np.arange(9).reshape(3,3)
print(x)
x=np.delete(x,1,axis=0)
print(x)
```

```
[[0 1 2]
 [3 4 5]
 [6 7 8]]
[[0 1 2]
 [6 7 8]]
```

Appending elements to an array

In [ ]:
```
# append(array,values,axis=None)
x=np.arange(9).reshape(3,3)
print(x)
x=np.append(x,[[11,22,33]],0)
print(x)
x=np.append(x,[[11],[22],[33],[44]],1)
print(x)
```

```
[[0 1 2]
 [3 4 5]
 [6 7 8]]
[[ 0  1  2]
 [ 3  4  5]
 [ 6  7  8]
 [11 22 33]]
[[ 0  1  2 11]
 [ 3  4  5 22]
 [ 6  7  8 33]
 [11 22 33 44]]
```

Inserting elements to an array

In [ ]:
```
# insert(array, pos, values, axis=None):

x=np.array([1,2,3,4,5])
print(x)
x=np.insert(x,3,111)
print(x)
```

```
[1 2 3 4 5]
[  1   2   3 111   4   5]
[[1 2 3]
 [4 5 6]
 [7 8 9]]
```

In [ ]:
```
y=np.arange(1,10).reshape((3,3))
print(y)
y=np.insert(y,1,[0,0,0],axis=0)
print(y)
```

```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
[[1 2 3]
 [0 0 0]
 [4 5 6]
 [7 8 9]]
```

In [ ]:
```
y=np.arange(1,10).reshape((3,3))
print(y)
y=np.insert(y,1,[[0,1,2]],axis=1)
print(y)
```

```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
[[1 0 2 3]
 [4 1 5 6]
 [7 2 8 9]]
```

Stack arrays on each other (Vertical Stack)

In [ ]:
```
x=np.array([[1,2,3],[4,5,6]])
print(x)
y=np.array([7,8,9])
z=np.vstack((x,y))
print(z)
```

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

Stack arrays on each other (Horizontal Stack)

In [ ]:
```
x=np.array([[1,2,3],[4,5,6]])
print(x)
y=np.array([[7],[8]])
z=np.hstack((x,y))
print(z)
```

```
[[1 2 3]
 [4 5 6]]
[[1 2 3 7]
 [4 5 6 8]]
```