Python NumPy Array Conctinatoin and Split

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

In [2]:
```
arr1 = np.arange(1,17).reshape(4,4)
print(arr1)
```

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

In [3]:
```
arr2 = np.arange(17,33).reshape(4,4)
print(arr2)
```

```
[[17 18 19 20]
 [21 22 23 24]
 [25 26 27 28]
 [29 30 31 32]]
```

In [4]:
```
list1 = [2,4, 6,8]
list2 = [10,12,14,16]
list1 + list2
#In list + operator perform as a concetenation but in numpy array isn't work
```

Out[4]:
```
[2, 4, 6, 8, 10, 12, 14, 16]
```

In [5]:
```
arr1 + arr2
```

Out[5]:
```
array([[18, 20, 22, 24],
       [26, 28, 30, 32],
       [34, 36, 38, 40],
       [42, 44, 46, 48]])
```

In [6]:
```
print(arr1)
print(arr2)
```

```
[[ 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]]
```

In [7]:
```
np.concatenate((arr1,arr2))
#column wise concatenation
```

Out[7]:
```
array([[ 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]])
```

In [8]:
```
np.concatenate((arr1,arr2), axis = 1)
#row wise concetatoin
```

Out[8]:
```
array([[ 1,  2,  3,  4, 17, 18, 19, 20],
       [ 5,  6,  7,  8, 21, 22, 23, 24],
       [ 9, 10, 11, 12, 25, 26, 27, 28],
       [13, 14, 15, 16, 29, 30, 31, 32]])
```

In [9]:
```
#funcaton for concetatoin row wise
np.vstack((arr1,arr2))
```

Out[9]:
```
array([[ 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]])
```

In [10]:
```
np.hstack((arr1,arr2))
```

Out[10]:
```
array([[ 1,  2,  3,  4, 17, 18, 19, 20],
       [ 5,  6,  7,  8, 21, 22, 23, 24],
       [ 9, 10, 11, 12, 25, 26, 27, 28],
       [13, 14, 15, 16, 29, 30, 31, 32]])
```

In [11]:
```
np.hstack((arr1,arr2,arr1))
```

Out[11]:
```
array([[ 1,  2,  3,  4, 17, 18, 19, 20,  1,  2,  3,  4],
       [ 5,  6,  7,  8, 21, 22, 23, 24,  5,  6,  7,  8],
       [ 9, 10, 11, 12, 25, 26, 27, 28,  9, 10, 11, 12],
       [13, 14, 15, 16, 29, 30, 31, 32, 13, 14, 15, 16]])
```

Split functoin

In [12]:
```
#divide the array in different part
np.split(arr1,2)
#the result of this functon always store array of splited  in the list data type start from
# 0 index
```

Out[12]:
```
[array([[1, 2, 3, 4],
        [5, 6, 7, 8]]), array([[ 9, 10, 11, 12],
        [13, 14, 15, 16]])]
```

In [13]:
```
list1 = np.split(arr1,2)
print(list1)
type(list1)
```

```
[array([[1, 2, 3, 4],
       [5, 6, 7, 8]]), array([[ 9, 10, 11, 12],
       [13, 14, 15, 16]])]
```

Out[13]:
```
list
```

In [14]:
```
list1[0] # splited first part of array store in list number 0 index
```

Out[14]:
```
array([[1, 2, 3, 4],
       [5, 6, 7, 8]])
```

In [15]:
```
list1[1] #second part of array store in list index number 1
```

Out[15]:
```
array([[ 9, 10, 11, 12],
       [13, 14, 15, 16]])
```

In [16]:
```
type(list1[0])
```

Out[16]:
```
numpy.ndarray
```

In [17]:
```
arr1
```

Out[17]:
```
array([[ 1,  2,  3,  4],
       [ 5,  6,  7,  8],
       [ 9, 10, 11, 12],
       [13, 14, 15, 16]])
```

In [18]:
```
#row wise split
np.split(arr1,2, axis = 1)
# np.split(arr1,4, axis = 1)
```

Out[18]:
```
[array([[ 1,  2],
        [ 5,  6],
        [ 9, 10],
        [13, 14]]), array([[ 3,  4],
        [ 7,  8],
        [11, 12],
        [15, 16]])]
```

In [19]:
```
d1 = np.array([4,7,1,3,9]) #index wise split
```

In [20]:
```
np.split(d1,[1,3])
```

Out[20]:
```
[array([4]), array([7, 1]), array([3, 9])]
```

In [21]:
```
d2 = np.array([12,45,16,17,20,30])
np.split(d2,[2,4])
```

Out[21]:
```
[array([12, 45]), array([16, 17]), array([20, 30])]
```