

```
In [1]: ► import numpy as np
np.random.seed() #on each run, these 5 numbers will be changed.
a=np.random.randint(1,10,5) #return any 5 numbers between 1 to 10
print(a)
print('-----')
np.random.seed(5) #on each run, these 5 numbers wont be changed.
a=np.random.randint(1,10,5)
print(a)
```

```
[3 9 8 6 4]
```

```
-----
```

```
[4 7 7 1 9]
```

```
In [2]: ► # matrix=[]
# row=int(input("enter number of rows : "))
# col=int(input("enter number of columns : "))
# for i in range (row):
#     a=[]
#     for j in range (col):
#         val = int(input("enter values : "))
#         a.append(val)
#     matrix.append(a)
# arr=np.array(matrix)
# for i in range (row):
#     for j in range (col):
#         print(arr[i,j],end=" ")
#     print()
# print(type(arr))
# print('-----')
# print(arr.ndim)
# print('-----')
# print(arr.size)
```

```
In [3]: ► a=np.array([[10,20,30],[40,50,60],[70,80,90]]) #converting list into array and assigning it to 'a'.
print(a)
print()
print(a[:2]) #row 0 and 1(2-1) will be printed.
print()
print(a[2,1:3]) #row 0 and 1(2-1) will be printed and column 1 and 2(3-1) will be printed.
print()
np.random.seed(1)
a=np.random.randint(1,50,30).reshape(6,5) # return any 30 num and convert it into 2D array (6R,5C)
print(a)
print()
print(a[2:,2:]) #print row 2 to upto limit and print column 2 to upto limit.
print()
#slicing : extract 38,19,21 and 29,30,15
print(a[3:5,2:])
```

```
[[10 20 30]
 [40 50 60]
 [70 80 90]]
```

```
[[10 20 30]
 [40 50 60]]
```

```
[[20 30]
 [50 60]]
```

```
[[38 44 13  9 10]
 [12  6 16  1 17]
 [ 2 13  8 46  7]
 [26 21 38 19 21]
 [12 43 29 30 15]
 [ 5 24 24 42 31]]
```

```
[[ 8 46  7]
 [38 19 21]
 [29 30 15]
 [24 42 31]]
```

```
[[38 19 21]
 [29 30 15]]
```

```
In [4]: ► a=np.array([10,20,30,40,50,60,70,80,90])
b=a[3:6].copy() #copy value from position 3 to 5(6-1)
print(b)
print('-----')
b[:]=0 #assign 0 from starting position to end position if nothing is specified.
print(b)
print('-----')
```

```
[40 50 60]
```

```
[0 0 0]
```

```
In [5]: ► a=np.arange(1,6) # make a list of values from 1 to 5(6-1).
b=np.arange(6,11) # make a list of values from 6 to 10(11-1).
print('a = ',a)
print('-----')
print('b = ',b)
a=a+2 # add 2 in each element of list a & a will be modified.
print('-----')
print('a = ',a)
print('-----')
print(a+2) # add 2 in each element of list a and original a wont changed.
print('-----')
print(a/0) # divide each element of list a by 2 and it will give infinity.
print('-----')
print(a+b) # add list a and list b.Original list wont be changed.
print('-----')
print(a-b) # subtract list b from list b.Original list wont be changed.
print('-----')
print('a = ',a)
print('b = ',b)
```

```
a = [1 2 3 4 5]
-----
b = [ 6  7  8  9 10]
-----
a = [3 4 5 6 7]
-----
[5 6 7 8 9]
-----
[inf inf inf inf inf]
-----
[ 9 11 13 15 17]
-----
[-3 -3 -3 -3 -3]
-----
a = [3 4 5 6 7]
b = [ 6  7  8  9 10]
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_5140\1791051099.py:12: RuntimeWarning: divide by zero encountered in divide
 print(a/0) # divide each element of list a by 2 and it will give infinity.

```
In [6]: ► a=np.arange(1,5).reshape(2,2) # make list of values from 1 to 4 and make it into 2D array.
print(a)
print('-----')
b=np.arange(5,9).reshape(2,2)# make list of values from 5 to 8 and make it into 2D array.
print('-----')
print(b)
print('-----')
print(a+b) #add array a and array b. Original arrays wont be changed.
print('ORIGINAL ARRAYS WONT GET CHANGED')
print(a)
print(b)
print('-----')
print(a-b) #subtract array b from array a. Original arrays wont be changed.
print('ORIGINAL ARRAYS WONT GET CHANGED')
print(a)
print(b)
print('-----')
print(a/b) #divide array a by array b. Original arrays wont be changed.
print('ORIGINAL ARRAYS WONT GET CHANGED')
print(a)
print(b)
print('-----')
print(a**b) #exponential. Original arrays wont be changed.
print('ORIGINAL ARRAYS WONT GET CHANGED')
print(a)
print(b)
print('-----')
print(a*b) #multiply array a and array b. Original arrays wont be changed.
print('ORIGINAL ARRAYS WONT GET CHANGED')
print(a)
print(b)
print('-----')
print(a.dot(b)) #matrix multiplication of array a and array b. Original arrays wont be changed.
print('ORIGINAL ARRAYS WONT GET CHANGED')
print(a)
print(b)
```

```
[[1 2]
 [3 4]]
-----
[[5 6]
 [7 8]]
-----
[[ 6 8]
 [10 12]]
ORIGINAL ARRAYS WONT GET CHANGED
[[1 2]
 [3 4]]
[[5 6]
 [7 8]]
-----
[[-4 -4]
 [-4 -4]]
ORIGINAL ARRAYS WONT GET CHANGED
[[1 2]
 [3 4]]
[[5 6]
 [7 8]]
-----
[[0.2      0.33333333]
 [0.42857143 0.5      ]]
ORIGINAL ARRAYS WONT GET CHANGED
[[1 2]
 [3 4]]
[[5 6]
 [7 8]]
-----
[[ 1 64]
 [ 2187 65536]]
ORIGINAL ARRAYS WONT GET CHANGED
[[1 2]
 [3 4]]
[[5 6]
 [7 8]]
-----
[[ 5 12]
 [21 32]]
ORIGINAL ARRAYS WONT GET CHANGED
[[1 2]
 [3 4]]
[[5 6]
 [7 8]]
-----
[[19 22]
 [43 50]]
ORIGINAL ARRAYS WONT GET CHANGED
[[1 2]
 [3 4]]
[[5 6]
 [7 8]]
```

In []: ►