numpy is a library

mostly deals with arrays and mathmatical calculations

we import this library using

import numpy as np

Arrays

data type must be same

length of the subarrays must be same

dont superated by commas " , "

eg: [1 3 4 5]

we can perform many mathmatical operations

Importin library

```
1 import numpy as np
2
```

creating Array

1-D **Array**

```
1 import numpy as np
2 arr = np.array([1,2,3,4,5])
3 print(arr)
```

```
[1 2 3 4 5]
```

2-D Array

Note : length must be same

```
1 import numpy as np
2 arr = np.array([[1,2,3,4,5],[6,7,8,9,10]])
3 print(arr)
```

```
[[ 1  2  3  4  5]
 [ 6  7  8  9 10]]
```

3-D Array

```
1 import numpy as np
2 arr = np.array([[1,2,3,4,5],[6,7,8,9,10],[12,23,4,56,9]])
3 print(arr)
```

```
[[ 1  2  3  4  5]
 [ 6  7  8  9 10]
 [12 23  4 56  9]]
```

Slicing the Array

arr[start : stop : step]

```
import numpy as np
arr = np.array([1,2,3,4,5])
print(arr[:2])
```

```
[1 2]
```

print(arr[: 2 , : 2])

```
import numpy as np
arr = np.array([[1,2,3,4,5],[6,7,8,9,10]])
print(arr[:2,:2])
```

```
[[1 2]
 [6 7]]
```

Double-click (or enter) to edit

```
import numpy as np
arr = np.array([[[1,2,3],[4,5,7]],[[8,9,67],[23,87,9]]])
print(arr[:1,1:2,2:3])
print(np.ndim(arr))

# # # print(arr[:2,:3,:4]) Wrong because length must be same
# print(arr[:2,1:3])
# # print(arr)

```

```
[[[7]]]
3
```

ATRIUTES

np.shape : to know number of rows and columns

np.size : to get total number of elements

np.ndim : to get dmention of the Array

np.dtype : to get data type of an Array

```
import numpy as np
arr = np.array([[1,2,3,4,5],[234,87,9,79,90]])
print(np.shape(arr))
print(np.size(arr))
print(np.dtype)
print(np.ndim(arr))
```

```
(2, 5)
10
<class 'numpy.dtype'>
2
```