**NUMPY TUTORIAL**

Numpy is very popular library of python used to handle multidimensional arrays. When data is arranges in the form of matrices, many dimensions of array NUMPY helps to manage it.

**pip install numpy**

**TERMINOLOGIES**

* **DIMENSIONS** of data refers to numbers of indices needed to access a specific element.
* **DATA TYPE** of array elements which kind of data they hold.
* **SIZE** of data how many elements are present there.
* **SHAPE** of array tells us how many elements are there by columns and rows.
* **Concatenate** will add the index to arrays.
* **Addition** would be add the values on that index.
* **Slicing** means extract out some values from specific position.
* **Reshape** of arrays means change the rows and col positions according to the size of arrays.
* **Arange** will go through from range

**OPEARTIONS OF NUMPY**

### ****Creating Arrays****

### **array = np.array([[1,2,3] , [3,5,6] , [22,11,33]]) #3by3 array**

### a = np.array([1, 2, 3]) # 1D array / 1by3 array

b = np.array ([[1, 2], [3, 4]]) # 2D array

### ****Array Attributes****

a.shape # (3,)

b.shape # (2, 2)

a.ndim # 1 (1D)

b.ndim # 2 (2D)

a.size #3

b.size #4

### ****Array Initialization****

np.zeros((2, 3)) # 2x3 array of zeros

np.ones((2, 2)) # 2x2 array of ones

np.full((2, 2), 7) # 2x2 array filled with 7

np.eye(3) / np.identity(3) # Identity matrix

np.random.rand(2, 3) # Random values in [0,1) (Dimensionality Reduction)

### ****Indexing and Slicing****

arr = np.array([10, 20, 30, 40])

arr[0] # 10

arr[1:3] # [20 30]

### ****Arithmetic Operations****

a = np.array([1, 2, 3])

b = np.array([4, 5, 6])

a + b # [5 7 9]

a \* b # [4 10 18]

a \*\* 2 # [1 4 9]

### ****Matrix Operations****

A = np.array([[1, 2], [3, 4]])

B = np.array([[2, 0], [1, 3]])

A.T # Transpose

### ****Useful Functions****

np.arange(a) # use the range

np.expand (a) # expand the 1D array to 2D array

np.newaxis (a) #convert row to column

axis = 0 , 1, none effect on the array set to be row , column

np.sum(a) # Sum of all elements

np.mean(a) # Average

np.max(a) # Max value

np.sort(a) # Sorted array

np.unique(a) # Unique elements