**NUMPY IN PYTHON**

Install the numpy package (as below)

A screenshot of a computer

Description automatically generated

Numpy is a great library to work with homogeneous numeric data, which uses integer based indexing. Its not the best choice for handing big data type.

**Code:**

from numpy import \*  
  
allArray= array([10,20,30,40,50,'a','Jazy'])  
intArray = array([10,20,30,40,50],int)  
charArray = array(['j','j','a','k'])  
stringArray = array(['C#','Java','Python'], dtype=str)  
  
*#ARRAY OUTPUT*print("================ NORMAL ARRAY =============")  
print(allArray)  
print()  
*#DEFINE ONLY INTEGER ARRAY*print("================ INTEGER ARRAY =============")  
print(intArray)  
print()  
  
*#CHAR ARRAY OUTPUT*print("================ CHAR ARRAY =============")  
print(charArray)  
print()  
  
*#STRING ARRAY OUTPUT*print("================ STRING ARRAY =============")  
print(stringArray)  
print()  
  
*#LINESPACE*print("================ LINESPACE =============")  
linespaceArray = linspace(1,50,25)  
print(linespaceArray)  
print()  
  
*#LOGSPACE*print("================ LOGSPACE =============")  
print(logspace(1,10))  
print()  
  
*#ARANGE*print("================ ARANGE =============")  
print(arange(1,5)) *# SIMILAR TO RANGE*print(arange(1,10,2)) *# STEP 2*print(arange(15,5,-2)) *# REVERSE*print()  
  
*#ZEROS*print("================ ZEROS =============")  
print(zeros(10))  
print()  
  
*#ONES*print("================ ONES =============")  
print(ones(10))  
print()

**Code (Maths Functions):**

from numpy import \*  
  
array1 = array([15,17,19,21,25])  
array2 = array([22,24,26,28,30])  
  
*#ADDING / SUB 2 ARRAYS*arraysum = (array1+array2) *# NO OF VALUES HAS TO MATCH*arraydiff = (array1-array2)  
  
print("=============== ADDING TWO ARRAYS ===========")  
print(arraysum)  
print()  
  
print("=============== DIFFERENCE TWO ARRAYS ===========")  
print(arraydiff)  
print()  
  
print("=============== SUBSTRACT 5 FROM ARRAY 1 ===========")  
print(array1 - 5)  
print()  
  
print("=============== SIN VALUES ===========")  
print(sin(array1))  
print(sin(array2))  
print  
  
print("=============== COS VALUES ===========")  
print(cos(array1))  
print(cos(array2))  
print()  
  
print("=============== LOG VALUES ===========")  
print(log(array1))  
print(log(array2))  
print()  
  
print("=============== MEAN VALUES ===========")  
print(mean(array1))  
print(mean(array2))  
print()  
  
print("=============== ABS VALUES ===========")  
print(abs(array1))  
print(abs(array2))  
print()  
  
print("=============== MIN and MAX VALUES ===========")  
print(min(array1))  
print(max(array1))  
print(min(array2))  
print(max(array2))  
print()

**Code (Array Copy):**

from numpy import \*  
  
array1 = arange(1,30,5)  
*#SHALLAOW COPY*array2 = array1.view()  
*# DEEP COPY*array3 = array1.copy()  
  
print("Initial Arrays")  
print("===============")  
print("Array 1 : ",array1)  
print()  
print("Array 2 : ",array2)  
print()  
  
*# ADD A NUMBER IN ANY POSISTION IN array2*array2[5] = 29  
print("After adding the value 30 in position 5")  
print("=======================================")  
print("Array 1 : ",array1)  
print()  
print("Array 2 : ",array2)  
print()  
  
*# ADD A NUMBER IN ANY POSISTION IN array2*array3[5] = 26  
print("After adding the value 30 in position 5")  
print("=======================================")  
print("Array 1 : ",array1)  
print()  
print("Array 2 : ",array2)  
print()  
print("Array 3 : ",array3)  
print()

**Code (Slicing):**

from numpy import \*  
  
array1 = arange(1,30)  
print(array1)  
  
*#SLICING (POSITION STARTS FROM 0 TO 10-1)*print("Array after slicing from position 5 to 10")  
print(array1[5:10])  
  
*#SLICING (POSITION STARTS FROM 0 TO 10-1)*print("Array after slicing from position 5 to 20 with step 5")  
print(array1[5:20:5])  
  
*#SLICING (POSITION STARTS POSTION ONLY)*print("Array after slicing from position 5.....")  
print(array1[5:])

**Code (Multidimensional Array):**

from numpy import \*  
  
*#SINGLE DIMENSION*array1 = arange(1,20)  
  
*# TWO DIMENSION*array2 = array([[10,20,30],[40,50,60]])  
  
*# THREE DIMENSION*array3 = array([[[100,200],[300,400]],[[500,600],[700,800]]])  
  
print("Three Dimensional Array : ", array3.ndim)  
print("Shape of the Three Dimensional Array : ",array3.shape)  
print("Size of the Three Dimensional Array : ",array3.size)  
print()  
print(array3)  
print()  
  
array3.shape = (4,2)  
print("Three Dimensional Array after changing the shape: ", array3.ndim)  
print("Size of the Three Dimensional Array : ",array3.size)  
print("After changing the shape of array : ",array3.shape)  
print()  
print(array3)  
print()  
  
print("========================================================")  
  
print("Two Dimensional Array : ", array2.ndim)  
print("Shape of the Two Dimensional Array : ",array2.shape)  
print("Size of the Two Dimensional Array : ",array2.size)  
print()  
print(array2)  
print()  
  
array2.shape = (3,2)  
print("Two Dimensional Array after changing the shape: ", array2.ndim)  
print("After changing the shape of array : ",array2.shape)  
print("Size of the Two Dimensional Array : ",array2.size)  
print()  
print(array2)  
print()  
print("========================================================")  
  
print("One Dimensional Array : ", array1.ndim)  
print("Shape of the Single Dimensional Array : ",array1.shape)  
print("Size of the Single Dimensional Array : ",array1.size)  
print()  
print(array1)  
print()  
  
print("========================================================")  
*#DATATYPE OF THE ARRAY*print("Data Type of an Array")  
print(array1.dtype)  
print(array2.dtype)  
print(array3.dtype)  
print()  
print("========================================================")  
*# ITEM SIZE OF THE ARRAY*print("Item Size of an Array")  
print(array1.itemsize)  
print(array2.itemsize)  
print(array3.itemsize)  
print("========================================================")  
*# NBYTES OF THE ARRAY*print("NBytes require for an entire Array ")  
print(array1.nbytes)  
print(array2.nbytes)  
print(array3.nbytes)  
print("========================================================")

**Code (Array Funcitons):**

from numpy import \*  
  
*#SINGLE DIMENSION*array1 = arange(1,11)  
print("Single Dimensional Array")  
print(array1)  
print()  
  
*#CONVERT THE SINGLE DIMENSIONAL ARRAY INTO 2 DIMENSIONAL ARRAY*print("Convert Single Dimension into 2 Dimensional")  
array2 = reshape(array1,(2,5))  
print(array2)  
print()  
  
*#IN THE ABOVE ARRAY THE VALUE IS THERE FROM 1-10 WHICH HAS 10, SO EQUALLY DONE.  
# IF WE CONVERT THIS INTO 3,3 : ERROR WILL BE : ValueError: cannot reshape array of size 10 into shape (3,3)  
#array3 = reshape(array1,(3,3))  
#print(array3)  
#print()  
  
#CONVERT THE SINGLE DIMENSIONAL ARRAY INTO 3 DIMENSIONAL ARRAY*print("Convert Single Dimension into 3 Dimensional")  
array1 = arange(12)  
array4 = reshape(array1,(2,2,3))  
print(array4)  
print()  
  
*#CONVERT THE MULTI DIMENSIONAL ARRAY INTO SINGLE DIMENSIONAL ARRAY*print("Convert Three Dimension into Single Dimensional")  
print(array4.flatten())  
print("Convert Double Dimension into Single Dimensional")  
print(array2.flatten())  
print()  
  
*#EYE FUNCTION WILL PRINT ALL THE DIAGONAL VALUE WITH 1*print("Three Dimensional")  
print(eye(3))  
print("Two Dimensional")  
print(eye(2))  
print()  
  
*#USING ONES AND ZERO IN TWO AND THREE DIMENSIONAL ARRAYS  
#BY DEFAULT ITS FLOAT TYPE*print(ones((2,3),int))  
print(zeros((2,3),int))

**QUIZ**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated