Numpy Library for beginner

Qusay AL-Btoush

github.com/qusaybtoush
https://www.linkedin.com/in/qusayal-btoush
https://www.kaggle.com/qusaybtoush1990

NumPy

For Beginner

NumPy is a popular Python library for numerical and scientific computing. It provides support for working with large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays. NumPy is an essential tool for data scientists, engineers, and researchers working with numerical data in Python.

- concat me if you have any questions : Qusay AL-Btoush

- https://github.com/qusaybtoush
- https://www.linkedin.com/in/qusayal-btoush/
- https://www.kaggle.com/qusaybtoush1990

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

In [61]:
```
# create array
x= np.array([1,2,3,4,5])
y= np.array([4,5,8,9,10])
print (x)
print (y)
```

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

In [7]:
```
#create array number 1 / just choice how many the rows and columns
one = np.ones((3,4)) #3 rows, 4 column
print(one)
```

```
[[1. 1. 1. 1.]
 [1. 1. 1. 1.]
 [1. 1. 1. 1.]]
```

In [9]:
```
#create array number 0 / just choice how many the rows and columns
ze = np.zeros((5,3)) #5 rows, 3 column
print(ze)
```

```
[[0. 0. 0.]
 [0. 0. 0.]
 [0. 0. 0.]
 [0. 0. 0.]
 [0. 0. 0.]]
```

In [11]:
```
#create array with any number you want  / just choice how many the rows and columns and
fu = np.full((3,6),8) # 3 rows, 6 columns  and the number 8
print(fu)
```

```
[[8 8 8 8 8 8]
 [8 8 8 8 8 8]
 [8 8 8 8 8 8]]
```

In [12]:
```
#create array with range and step
st = np.arange(5,31,5) # start from 5 to 31 and the step 5
print(st)
```

```
[ 5 10 15 20 25 30]
```

In [21]:
```
#create array with range and how many element you want
li = np.linspace(1,5,6) # start from 1 to 5 and 6 elements
print(li)
```

```
[1.  1.8 2.6 3.4 4.2 5. ]
```

In [24]:
```
# creates a 4*4 identity matrix with ones on the main diagonal and zeros elsewhere
np.eye(4)
```

Out[24]:
```
array([[1., 0., 0., 0.],
       [0., 1., 0., 0.],
       [0., 0., 1., 0.],
       [0., 0., 0., 1.]])
```

In [31]:
```
# create random array 0 to 1
np.random.rand(2,2) #array 2 rows, 2 column and the element from 0 to 1
```

Out[31]:
```
array([[0.07635816, 0.99177734],
       [0.23383263, 0.93246568]])
```

In [37]:
```
# create random array int number
np.random.randint(5,size=(4,4)) # 5 the max number in the array from 0 to 5 you can cho
```

Out[37]:
```
array([[4, 0, 1, 0],
       [2, 4, 0, 2],
       [1, 1, 2, 4],
       [2, 0, 3, 1]])
```

In [39]:
```
# create array 2 dimension
arr= np.array([[1,3,5,9],[10,15,6,8]])
print (arr)
```

```
[[ 1  3  5  9]
 [10 15  6  8]]
```

In [45]:
```
#show the dimension array name  arr
np.ndim(arr)  # using function ndim
```

Out[45]:
```
2
```

In [72]:
```
#show the unique element
np.unique(arr)
```

Out[72]:
```
array([ 1,  3,  5,  6,  8,  9, 10, 15])
```

In [46]:
```
#show the shape array name  arr
np.shape(arr) #using function shape , we have 2 rows and 4 columns
```

Out[46]:
```
(2, 4)
```

In [48]:
```
#show the size array name  arr
np.size(arr)  #using function size, we have 8 element
```

Out[48]:
```
8
```

In [52]:
```
#show ths data type, using function dtype
arr.dtype
```

Out[52]:
```
dtype('int32')
```

In [58]:
```
# we can create with any kind of element in the array, like bool data type
bol= np.array([True, False,False, False, True])
bol.dtype
```

Out[58]:
```
dtype('bool')
```

In [54]:
```
# you can flip the array, using function flip
np.flip(arr)
```

Out[54]:
```
array([[ 8,  6, 15, 10],
       [ 9,  5,  3,  1]])
```

In [87]:
```
#show the array arr
arr
```

Out[87]:
```
array([[ 1,  3,  5,  9],
       [10, 15,  6,  8]])
```

In [88]:
```
#show the index [0]= fisrt rows 1= second rows
arr[1]
```

Out[88]:
```
array([10, 15,  6,  8])
```

In [92]:
```
#show the second column [:,1] this mean all rows, 1 first column
arr[:,1]
```

Out[92]:
```
array([ 3, 15])
```

In [95]:
```
#set the index [ row 1 and column 2] * the index here in numpy start from 0
arr[1,2]
```

Out[95]:
```
6
```

In [96]:
```
#you can do transpose
np.transpose(arr)
```

Out[96]:
```
array([[ 1, 10],
       [ 3, 15],
       [ 5,  6],
       [ 9,  8]])
```

In [101…
```
# using function resize you can resize the array
np.resize(arr,(5,5))
```

Out[101…
```
array([[ 1,  3,  5,  9, 10],
       [15,  6,  8,  1,  3],
       [ 5,  9, 10, 15,  6],
       [ 8,  1,  3,  5,  9],
       [10, 15,  6,  8,  1]])
```

In [102…
```
# using function append you can add new element
np.append(arr,(1,5,6))
```

Out[102…
```
array([ 1,  3,  5,  9, 10, 15,  6,  8,  1,  5,  6])
```

In [105…
```
#you can delete element , based on the index
np.delete(arr,[1,1])
```

Out[105…
```
array([ 1,  5,  9, 10, 15,  6,  8])
```

In [62]:
```
# we defined 2 array at the first x and y
print (x)
print (y)
```

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

In [65]:
```
#Array Mathematics, you can do any operations,
# Subtraction, Addition, Division, Multiplication

print (x + y )  #Addition
print ( x - y )  #Subtraction
print (x / y  ) #Division
print (x * y )  #Multiplication
```

```
[ 5  7 11 13 15]
[-3 -3 -5 -5 -5]
[0.25       0.4        0.375      0.44444444 0.5       ]
[ 4 10 24 36 50]
```

In [67]:
```
# you can do maths operations, like sin, sqrt, cos, log ...
print (np.sin(x) )
print (np.sqrt(x) )
print (np.cos(y) )
print (np.log(y) )
```

```
[ 0.84147098  0.90929743  0.14112001 -0.7568025  -0.95892427]
[1.         1.41421356 1.73205081 2.         2.23606798]
[-0.65364362  0.28366219 -0.14550003 -0.91113026 -0.83907153]
[1.38629436 1.60943791 2.07944154 2.19722458 2.30258509]
```

In [70]:

```
# as well you do statistic function sum, min, max, mean, median, and sort the array
print (x)

print ("The Sum :",np.sum(x) )
print ("The Minimum : ",np.min(x) )
print ("The Maximum ",np.max(x) )

print ("The Average :",np.mean(x) )
print ("The Median : " , np.median(x) )
print ("The Sort the array : " , np.sort(x) )
```

```
[1 2 3 4 5]
The Sum : 15
The Minimum :  1
The Maximum  5
The Average : 3.0
The Median :  3.0
The Sort the array :  [1 2 3 4 5]
```

- concat me if you have any questions : Qusay AL-Btoush
- https://github.com/qusaybtoush
- https://www.linkedin.com/in/qusayal-btoush/
- https://www.kaggle.com/qusaybtoush1990