Numpy - 19BCP138

--> Importing Library

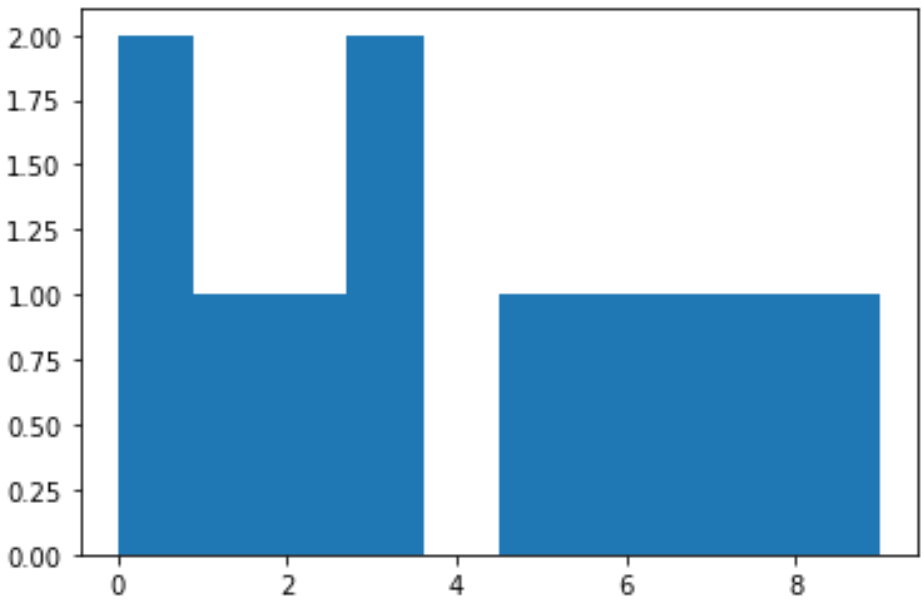
```
from matplotlib import pyplot as plt
import numpy as np
```

--> Compute the histogram of a set of numbers stored in a numpy array

```
a = np.array([3,6,7,3,0,9,2,1,8,0,5])

# To plot Histogram
plt.hist(a)
```

```
(array([2., 1., 1., 2., 0., 1., 1., 1., 1., 1.]),
 array([0. , 0.9, 1.8, 2.7, 3.6, 4.5, 5.4, 6.3, 7.2, 8.1, 9. ]),
 <a list of 10 Patch objects>)
```



--> Count the number of words in a string

```
string = "My Name is Vedant Patel."
print ("The original string is : " +  string)

# To count words in string
res = len(string.split())
print ("The number of words in string are : " +  str(res))
```

```
The original string is : My Name is Vedant Patel.
The number of words in string are : 5
```

--> Find the Euclidean Distance between two 1D arrays

```
point1 = np.array((4, 5, 8))
point2 = np.array((9, 3, 6))

# Calculating Euclidean Distance using linalg.norm()
dist = np.linalg.norm(point1 - point2)
print(dist)
```

```
5.744562646538029
```

--> Calculate the sum of diagonal elements

```
arr = np.array([[1,2,3], [4,5,6], [7,8,9]])
print("The Numpy Array is:")
print(arr)

trace = np.trace(arr)
print("\nThe Sum of Diagonal Elements are:", trace)
```

```
The Numpy Array is:
[[1 2 3]
 [4 5 6]
 [7 8 9]]

The Sum of Diagonal Elements are: 15
```

--> Return the indices of elements where the given condition is satisfied

```
c = np.arange(11)

# Condition
res = c[c%2==0]
print("The Array is:", c)

print("The Sorted Arrray is:", res)
```

```
The Array is: [ 0  1  2  3  4  5  6  7  8  9 10]
The Sorted Arrray is: [ 0  2  4  6  8 10]
```

--> Get row numbers of numpy array having elements larger than some xvalue

```
d = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15])
print(d)

e = np.where(d > 10)
print("Elements which are larger than 10:", d[e])
```

```
[ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15]
Elements which are larger than 10: [11 12 13 14 15]
```

--> Finding the k smallest values of a numpy array

```
arr1 = np.array([2,45,56,23,20,1,4,5,6,7,8])
print("The Original Array is:" , arr1)

k = 5
p = np.sort(arr1)

print("The", k, "Smallest value of the numpy array is:", p[:k])
```

```
The Original Array is: [ 2 45 56 23 20  1  4  5  6  7  8]
The 5 Smallest value of the numpy array is: [1 2 4 5 6]
```

--> Compute the 25th, 50th, 75th percentile of a numpy array

```
arr2 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
print("The Array is : ", arr2)
print("25th percentile of the Array : ",
      np.percentile(arr2, 25))
print("50th percentile of the Array : ",
      np.percentile(arr2, 50))
print("75th percentile of the Array : ",
      np.percentile(arr2, 75))
```

```
The Array is :  [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
25th percentile of the Array :  3.75
50th percentile of the Array :  6.5
75th percentile of the Array :  9.25
```

--> Finding the inverse of a matrix. Also check how to compute the inverse if matrix

```
A = np.array([[1, 5, 3], [4, 5, 6], [7, 8, 9]])
print(np.linalg.inv(A))

B = np.array([[5,7], [9,4]])
print("\nThe Array is:")
print(B)

print("\nThe Inverse of the array is:")
np.linalg.det(B)
print(np.linalg.inv(B))
```

```
[[-0.16666667 -1.16666667  0.83333333]
 [ 0.33333333 -0.66666667  0.33333333]
 [-0.16666667  1.5        -0.83333333]]

The Array is:
[[5 7]
 [9 4]]

The Inverse of the array is:
[[-0.09302326  0.1627907 ]
 [ 0.20930233 -0.11627907]]
```