P S PAVAN KUMAR

21BCB7139 ASSIGNMENT-1

Import NumPy as np

```
In [2]: import numpy as np
```

Create an array of 10 zeros

```
In [3]: a=np.zeros(10)
```

```
In [4]: a
```

```
Out[4]: array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0.])
```

Create an array of 10 ones

```
In [5]: b=np.ones(10)
```

```
In [6]: b
```

```
Out[6]: array([1., 1., 1., 1., 1., 1., 1., 1., 1., 1.])
```

Create an array of 10 fives

```
In [8]: c=np.full(10,5)
```

```
In [9]: c
```

```
Out[9]: array([5, 5, 5, 5, 5, 5, 5, 5, 5, 5])
```

Create an array of the integers from 10 to 50

```
In [11]: arr=np.arange(10,51)
         print(arr)
```

```
[10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50]
```

Create an array of all the even integers from 10 to 50

```
In [12]: arr=np.arange(10,51,2)
         print(arr)
```

```
[10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50]
```

Create a 3x3 matrix with values ranging from 0 to 8

```
In [13]: x=np.array([[[0,1,2],[3,4,5],[6,7,8]]])
```

```
In [14]: x
```

```
Out[14]: array([[[0, 1, 2],
                 [3, 4, 5],
                 [6, 7, 8]]])
```

Create a 3x3 identity matrix

```
In [15]: w=np.eye(3)
```

```
In [16]: w
```

```
Out[16]: array([[1., 0., 0.],
                [0., 1., 0.],
                [0., 0., 1.]])
```

Use NumPy to generate a random number between 0 and 1

```
In [17]: j=np.random.rand()
         print(j)
```

```
0.35723041231668873
```

Use NumPy to generate an array of 25 random numbers sampled from a standard normal distribution

```
In [18]: d=np.random.randn(25)
         print(d)
```

```
[ 1.25672915  1.58980986 -0.76339614 -0.40612831 -1.60040518  2.06121143
  0.71280324 -0.51572333  2.29829254 -1.00007662 -0.63890038 -0.58691598
  0.50198629  0.17710587 -1.96003092  0.92998831 -0.05962203 -0.1014562
 -0.94893058 -2.01991079 -0.17672986  0.01520591 -0.04165433  0.85243855
 -0.62033224]
```

Create the following matrix:

```
In [19]: e=np.arange(0.01,1.01,0.01).reshape(10,10)
         print(e)
```

```
[[0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08 0.09 0.1 ]
 [0.11 0.12 0.13 0.14 0.15 0.16 0.17 0.18 0.19 0.2 ]
 [0.21 0.22 0.23 0.24 0.25 0.26 0.27 0.28 0.29 0.3 ]
 [0.31 0.32 0.33 0.34 0.35 0.36 0.37 0.38 0.39 0.4 ]
 [0.41 0.42 0.43 0.44 0.45 0.46 0.47 0.48 0.49 0.5 ]
 [0.51 0.52 0.53 0.54 0.55 0.56 0.57 0.58 0.59 0.6 ]
 [0.61 0.62 0.63 0.64 0.65 0.66 0.67 0.68 0.69 0.7 ]
 [0.71 0.72 0.73 0.74 0.75 0.76 0.77 0.78 0.79 0.8 ]
 [0.81 0.82 0.83 0.84 0.85 0.86 0.87 0.88 0.89 0.9 ]
 [0.91 0.92 0.93 0.94 0.95 0.96 0.97 0.98 0.99 1.  ]]
```

Create an array of 20 linearly spaced points between 0 and 1:

```
In [20]: k=np.linspace(0,1,20)
         print(k)
```

```
[0.         0.05263158 0.10526316 0.15789474 0.21052632 0.26315789
 0.31578947 0.36842105 0.42105263 0.47368421 0.52631579 0.57894737
 0.63157895 0.68421053 0.73684211 0.78947368 0.84210526 0.89473684
 0.94736842 1.        ]
```

Numpy Indexing and Selection

```
In [21]: mat = np.arange(1,26).reshape(5,5)
         mat
```

```
Out[21]: array([[ 1,  2,  3,  4,  5],
                [ 6,  7,  8,  9, 10],
                [11, 12, 13, 14, 15],
                [16, 17, 18, 19, 20],
                [21, 22, 23, 24, 25]])
```

```
In [22]: mat[2:,1:5]
```

```
Out[22]: array([[12, 13, 14, 15],
                [17, 18, 19, 20],
                [22, 23, 24, 25]])
```

```
In [23]: mat[3,4]
```

```
Out[23]: 20
```

```
In [24]: mat[0:3 ,1:2]
```

```
Out[24]: array([[ 2],
                [ 7],
                [12]])
```

```
In [25]:  mat[-1,:]
```

```
Out[25]: array([21, 22, 23, 24, 25])
```

```
In [26]: mat[-2:,:]
```

```
Out[26]: array([[16, 17, 18, 19, 20],
                [21, 22, 23, 24, 25]])
```

Get the sum of all the values in mat

```
In [27]: mat.sum()
```

```
Out[27]: 325
```

Get the standard deviation of the values in mat

```
In [28]: o=np.std(mat)
```

```
In [29]: o
```

```
Out[29]: 7.211102550927978
```

Get the sum of all the columns in mat

```
In [30]: c=np.sum(mat,axis=0)
         print(c.tolist())
```

```
[55, 60, 65, 70, 75]
```

```
In [31]: u=np.array([55,60,65,70,75])
```

```
In [32]: u
```

```
Out[32]: array([55, 60, 65, 70, 75])
```

```
In [33]: u.sum()
```

```
Out[33]: 325
```