Aditi Rao EXTC B 118A2088

Numpy Exercise Part A:

1. Import NumPy as np¶
2. Create an array of 10 zeros Expected ouput: array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0.])
3. Create an array of 10 ones Expected ouput: array([1., 1., 1., 1., 1., 1., 1., 1., 1., 1.])
4. Create an array of 10 fives Expected ouput: array([5., 5., 5., 5., 5., 5., 5., 5., 5., 5.])
5. Create an array of the integers from 10 to 50 Expected ouput: array([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])
6. Create an array of all the even integers from 10 to 50 Expected ouput: array([10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42,44, 46, 48, 50])
7. Create a 3x3 matrix with values ranging from 0 to 8 Expected ouput: array([[0, 1, 2], [3, 4, 5], [6, 7, 8]])
8. Create a 3x3 identity matrix Expected ouput: array([[1., 0., 0.], [0., 1., 0.], [0., 0., 1.]])
9. Use NumPy to generate a random number between 0 and 1 Expected ouput: array([0.42829726])
10. Use NumPy to generate an array of 25 random numbers sampled from a standard normal distribution Expected ouput: array([1.32031013, 1.6798602 , -0.42985892, -1.53116655, 0.85753232,
11. 87339938, 0.35668636, -1.47491157, 0.15349697, 0.99530727,

-0.94865451, -1.69174783, 1.57525349, -0.70615234, 0.10991879, -0.49478947, 1.08279872, 0.76488333, -2.3039931 , 0.35401124, -0.45454399, -0.64754649, -0.29391671, 0.02339861, 0.38272124])

11. Create an array of 20 linearly spaced points between 0 and 1: Expected ouput: array([0. , 0.05263158, 0.10526316, 0.15789474, 0.21052632,
12. 26315789, 0.31578947, 0.36842105, 0.42105263, 0.47368421,
13. 52631579, 0.57894737, 0.63157895, 0.68421053, 0.73684211,
14. 78947368, 0.84210526, 0.89473684, 0.94736842, 1.])

Numpy Indexing and Selection Consider the following 2darray

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

12. Now do the following a. Get the sum of all the values in y Expected ouput: 222 b. Get the standard deviation of the values in y Expected ouput: 4.232808366400098

c. Get the sum of all the columns in y Expected ouput: array([51, 54, 57, 60])

d. What will be the output for the following line y[1:3,0:2]

```
 Expected ouput:  array([[17, 18],
    [22, 23]])
```

```
import numpy as np
```

```
np.zeros(10)
```

```
array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0.])
```

```
np.ones(10)
```

```
array([1., 1., 1., 1., 1., 1., 1., 1., 1., 1.])
```

```
np.ones(10)*5
```

```
array([5., 5., 5., 5., 5., 5., 5., 5., 5., 5.])
```

```
np.arange(10,51)
```

```
array([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])
```

```
np.arange(10,51,2)
```

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

```
np.arange(9).reshape(3,3)
```

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

```
np.eye(3)
```

```
array([[1., 0., 0.],
```

```
            [0., 1., 0.],
            [0., 0., 1.]])
```

```
np.random.rand(1)
```

```
     array([0.09293975])
```

```
np.random.normal(0,1,25)
```

```
     array([-0.3052661 , -0.16142307,  0.86033155,  0.08627231,  1.08330077,
             0.13963604, -0.34056118,  0.14023259, -1.50125256,  0.08702996,
            -2.19542342,  0.09721533, -0.20004484,  1.92362625, -1.16232805,
            -0.92299554,  0.949434  ,  0.2540811 , -0.87976717, -0.10766334,
             0.040681  ,  0.46670653, -1.24970903, -0.91916977,  0.19316143])
```

```
np.linspace(0,1,20)
```

```
     array([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.        ])
```

```
y = np.array([[12, 13, 14, 15],
    [17, 18, 19, 20],
    [22, 23, 24, 25]])
```

```
np.sum(y)
```

```
     222
```

```
np.std(y)
```

```
     4.232808366400098
```

```
sum(y)
```

```
     array([51, 54, 57, 60])
```

```
y[1:3,0:2]
```

```
     array([[17, 18],
            [22, 23]])
```

✓ 0s completed at 12:57 PM