

## Question 1:

```
import numpy as np
```

## Question 2:

```
np.zeros(10)  
array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0.])
```

## Question 3:

```
np.ones(10)  
array([1., 1., 1., 1., 1., 1., 1., 1., 1., 1.])
```

## Question 4:

```
np.full(10,5)  
array([5, 5, 5, 5, 5, 5, 5, 5, 5, 5])
```

## Question 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])
```

## Question 6

```
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])
```

## Question 7

```
np.arange(9).reshape(3,3)  
array([[0, 1, 2],  
[3, 4, 5],  
[6, 7, 8]])
```

### Question 8

```
np.eye(3)

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

### Question 9

```
r = np.random.rand()
r

0.30547353322579307
```

### Question 10

```
np.random.randn(25)

array([ 0.6602066 , -0.57023144,  1.06930178, -1.70818613, -
 0.72530373,
        1.84852434, -1.72179758,  1.01206355, -1.59104299,
 1.22603043,
        0.67827645,  0.84222445, -2.07238069,  0.76240608, -
 1.13612558,
        -0.54100959,  0.07984805,  0.23852444,  1.59945522, -
 0.67440838,
        0.59960223, -0.06241337,  0.41562584, -0.31518895,
 0.07199365])
```

### Question 11

```
np.arange(0.01,1.01,0.01).reshape(10,10)

array([[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.  ]])
```

### Question 12

```
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.          ])
```

## NUMPY INDEXING AND SELECTING

```
array = np.arange(1,26).reshape(5,5)
array
```

```
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]])
```

### Question 1

```
subarray = array[2:5, 1:]
subarray

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

### Question 2

```
value_20 = array[3,4]
value_20

20
```

### Question 3

```
subarray = array[0:3,1:2]
subarray

array([[ 2],
       [ 7],
       [12]])
```

### Question 4

```
subarray = array[4:5,0:5]
subarray

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

### Question 5

```
subarray=array[3:5,0:5]
subarray

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

#### Sum of all values in matrix

```
total_sum = np.sum(array)
print(total_sum)

325
```

#### Standard Deviation

```
deviate = np.std(array)
deviate

7.211102550927978
```

#### Sum of columns

```
colsum = np.sum(array,axis =0)
colsum

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