

## NUMPY ARRAYS INTRO

In [177]:

```
import numpy as np
```

In [178]:

```
list = [1,2,3,4,5]  
list
```

Out[178]:

```
[1, 2, 3, 4, 5]
```

In [179]:

```
np.array(list)
```

Out[179]:

```
array([1, 2, 3, 4, 5])
```

In [180]:

```
matrix = [[1,2,3,0],[4,5,6,10],[7,8,9,11]]  
matrix
```

Out[180]:

```
[[1, 2, 3, 0], [4, 5, 6, 10], [7, 8, 9, 11]]
```

In [181]:

```
np.array(matrix)
```

Out[181]:

```
array([[ 1,  2,  3,  0],  
       [ 4,  5,  6, 10],  
       [ 7,  8,  9, 11]])
```

In [182]:

```
np.arange(0,10)
```

Out[182]:

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

In [183]:

```
np.arange(0,11,2)
```

Out[183]:

```
array([ 0,  2,  4,  6,  8, 10])
```

In [184]:

```
np.zeros(3)
```

Out[184]:

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

In [185]:

```
np.zeros((5,5))
```

Out[185]:

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

In [186]:

```
np.ones(3)
```

Out[186]:

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

In [187]:

```
np.linspace(0,10,3)
```

Out[187]:

```
array([ 0.,  5., 10.])
```

In [188]:

```
np.eye(4)
```

Out[188]:

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

In [189]:

```
np.random.rand(2)
```

Out[189]:

```
array([0.66765914, 0.38644837])
```

In [190]:

```
np.random.rand(5,5)
```

Out[190]:

```
array([[0.36545681, 0.30145784, 0.45342776, 0.16849891, 0.23200717],
       [0.8817242 , 0.11158517, 0.22216057, 0.84003629, 0.13289205],
       [0.40952877, 0.46401861, 0.31544002, 0.91253952, 0.41878136],
       [0.11838112, 0.91410628, 0.266818 , 0.79350955, 0.21671133],
       [0.90285242, 0.60677348, 0.79074439, 0.96584837, 0.40924201]])
```

In [191]:

```
np.random.randint(1,100)
```

Out[191]:

In [192]:

```
np.random.randint(1,100,10)
```

Out[192]:

```
array([ 1, 39, 77, 95, 28, 85, 88, 14, 49, 97])
```

In [193]:

```
arr = np.arange(25)  
ranarr = np.random.randint(0,50,10)
```

In [194]:

```
arr
```

Out[194]:

```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,  
       17, 18, 19, 20, 21, 22, 23, 24])
```

In [195]:

```
ranarr
```

Out[195]:

```
array([ 7, 28,  6, 30, 45, 15, 20, 29, 22, 23])
```

In [196]:

```
arr.reshape(5,5)
```

Out[196]:

```
array([[ 0,  1,  2,  3,  4],  
       [ 5,  6,  7,  8,  9],  
       [10, 11, 12, 13, 14],  
       [15, 16, 17, 18, 19],  
       [20, 21, 22, 23, 24]])
```

In [197]:

```
ranarr
```

Out[197]:

```
array([ 7, 28,  6, 30, 45, 15, 20, 29, 22, 23])
```

In [198]:

```
ranarr.max()
```

Out[198]:

```
45
```

In [199]:

```
ranarr.argmax()
```

Out[199]:

```
4
```

In [200]:

```
ranarr.min()
```

Out[200]:

6

In [201]:

```
ranarr.argmin()
```

Out[201]:

2

In [202]:

```
arr.shape
```

Out[202]:

(25,)

In [203]:

```
arr.reshape(1,25)
```

Out[203]:

```
array([[ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15,
        16, 17, 18, 19, 20, 21, 22, 23, 24]])
```

In [204]:

```
arr.reshape(1,25).shape
```

Out[204]:

(1, 25)

In [205]:

```
arr.reshape(25,1)
```

Out[205]:

```
array([[ 0],
       [ 1],
       [ 2],
       [ 3],
       [ 4],
       [ 5],
       [ 6],
       [ 7],
       [ 8],
       [ 9],
      [10],
      [11],
      [12],
      [13],
      [14],
      [15],
      [16],
      [17],
      [18],
      [19],
      [20],
      [21],
      [22],
```

```
[23],  
[24]])
```

In [206]:

```
arr.reshape(25,1).shape
```

Out[206]:

```
(25, 1)
```

In [207]:

```
arr.dtype
```

Out[207]:

```
dtype('int32')
```

## INDEXING AND SELECTING

In [208]:

```
arr = np.arange(0,11)
```

In [209]:

```
arr
```

Out[209]:

```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10])
```

In [210]:

```
arr[8]
```

Out[210]:

```
8
```

In [211]:

```
arr[1:5]
```

Out[211]:

```
array([1, 2, 3, 4])
```

In [212]:

```
arr[0:5]
```

Out[212]:

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

In [213]:

```
arr[0:5]=100
```

In [214]:

```
arr
```

Out[214]:

```
array([100, 100, 100, 100, 100, 5, 6, 7, 8, 9, 10])
```

In [215]:

```
arr = np.arange(0,11)
```

In [216]:

```
arr
```

Out[216]:

```
array([ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10])
```

In [217]:

```
slice = arr[0:6]
```

In [218]:

```
slice
```

Out[218]:

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

In [219]:

```
slice[:]=99
```

In [220]:

```
slice
```

Out[220]:

```
array([99, 99, 99, 99, 99, 99])
```

In [221]:

```
arr
```

Out[221]:

```
array([99, 99, 99, 99, 99, 99, 6, 7, 8, 9, 10])
```

In [222]:

```
arr_copy = arr.copy()
```

In [223]:

```
arr_copy
```

Out[223]:

```
array([99, 99, 99, 99, 99, 99, 6, 7, 8, 9, 10])
```

In [224]:

```
arr_2d = np.array([[5,10,15],[20,25,30],[35,40,45]])
```

In [225]:

```
arr_2d
```

Out[225]:

```
array([[ 5, 10, 15],
       [20, 25, 30],
       [35, 40, 45]])
```

In [226]:

```
arr_2d[1]
```

Out[226]:

```
array([20, 25, 30])
```

In [227]:

```
arr_2d[1][0]
```

Out[227]:

```
20
```

In [228]:

```
arr_2d[1,0]
```

Out[228]:

```
20
```

In [229]:

```
arr_2d[:2,1:]
```

Out[229]:

```
array([[10, 15],
       [25, 30]])
```

In [230]:

```
arr_2d[2]
```

Out[230]:

```
array([35, 40, 45])
```

In [231]:

```
arr_2d[2,:]
```

Out[231]:

```
array([35, 40, 45])
```

In [232]:

```
arr2d = np.zeros((10,10))
```

In [233]:

```
arr_length = arr2d.shape[1]
```

In [234]:

```
for i in range(arr_length):
    arr2d[i] = i

arr2d
```

Out[234]:

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

In [235]:

```
arr2d[[2,4,6,8]]
```

Out[235]:

```
array([[2., 2., 2., 2., 2., 2., 2., 2., 2., 2.],
       [4., 4., 4., 4., 4., 4., 4., 4., 4., 4.],
       [6., 6., 6., 6., 6., 6., 6., 6., 6., 6.],
       [8., 8., 8., 8., 8., 8., 8., 8., 8., 8.]])
```

In [236]:

```
arr2d[[6,4,2,7]]
```

Out[236]:

```
array([[6., 6., 6., 6., 6., 6., 6., 6., 6., 6.],
       [4., 4., 4., 4., 4., 4., 4., 4., 4., 4.],
       [2., 2., 2., 2., 2., 2., 2., 2., 2., 2.],
       [7., 7., 7., 7., 7., 7., 7., 7., 7., 7.]])
```

In [237]:

```
arr = np.arange(1,11)
arr
```

Out[237]:

```
array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10])
```

In [238]:

```
arr > 4
```

Out[238]:

```
array([False, False, False, False,  True,  True,  True,  True,  True,
        True])
```

In [239]:

```
bool_arr = arr>4
```

In [240]:

```
bool_arr
```

Out[240]:



```
array([False, False, False, False,  True,  True,  True,  True,  True,
       True])
```

In [241]:

```
arr[bool_arr]
```

Out[241]:

```
array([ 5,  6,  7,  8,  9, 10])
```

In [242]:

```
arr[arr>2]
```

Out[242]:

```
array([ 3,  4,  5,  6,  7,  8,  9, 10])
```

In [243]:

```
x = 2
arr[arr>x]
```

Out[243]:

```
array([ 3,  4,  5,  6,  7,  8,  9, 10])
```

## OPERATIONS

In [244]:

```
arr = np.arange(0,10)
```

In [245]:

```
arr
```

Out[245]:

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

In [246]:

```
arr + arr
```

Out[246]:

```
array([ 0,  2,  4,  6,  8, 10, 12, 14, 16, 18])
```

In [247]:

```
arr * arr
```

Out[247]:

```
array([ 0,  1,  4,  9, 16, 25, 36, 49, 64, 81])
```

In [248]:

```
arr - arr
```

Out[248]:

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

In [249]:

```
arr/arr
```

```
C:\Users\sharuk\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: RuntimeWarning: invalid value encountered in true_divide
  """Entry point for launching an IPython kernel.
```

Out[249]:

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

In [250]:

```
1/arr
```

```
C:\Users\sharuk\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: RuntimeWarning: divide by zero encountered in true_divide
  """Entry point for launching an IPython kernel.
```

Out[250]:

```
array([      inf,  1.,          0.5,          0.33333333,  0.25,
         0.2,          0.16666667,  0.14285714,  0.125,          0.11111111])
```

In [251]:

```
arr**3
```

Out[251]:

```
array([  0,    1,    8,   27,   64,  125,  216,  343,  512,  729], dtype=int32)
```

In [252]:

```
np.sqrt(arr)
```

Out[252]:

```
array([0.,          1.,          1.41421356,  1.73205081,  2.,
        2.23606798,  2.44948974,  2.64575131,  2.82842712,  3.
        ])
```

In [ ]: