

Welcome to num py

```
In [1]: import numpy as np
```

## how to make numpy array with help of python list

```
In [2]: myarr = np.array([3,6,7,32], np.int8)
```

```
In [3]: myarr
```

```
Out[3]: array([ 3,  6,  7, 32], dtype=int8)
```

```
In [4]: myarr[0]
```

```
Out[4]: 3
```

```
In [5]: myarr.shape
```

```
Out[5]: (4,)
```

```
In [6]: myarr.dtype
```

```
Out[6]: dtype('int8')
```

## array creation : conversion from other python structures

```
In [7]: listarray = np.array([[1,2,3],[8,9,5],[65,4,8]])
```

```
In [8]: listarray
```

```
Out[8]: array([[ 1,  2,  3],  
               [ 8,  9,  5],  
               [65,  4,  8]])
```

```
In [9]: zeros = np.zeros((2,5))
```

```
In [10]: zeros
```

```
Out[10]: array([[0., 0., 0., 0., 0.],  
               [0., 0., 0., 0., 0.]])
```

```
In [11]: zeros.dtype
```

```
Out[11]: dtype('float64')
```

```
In [12]: rng = np.arange(15)
```

```
In [13]: rng
```

```
Out[13]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14])
```

```
In [14]: linspace = np.linspace(1,5,4)
```

```
In [15]: linspace
```

```
Out[15]: array([1.          , 2.33333333, 3.66666667, 5.          ])
```

```
In [16]: ide = np.identity(45)
```

```
In [17]: ide
```

```
Out[17]: array([[1., 0., 0., ..., 0., 0., 0.],
                [0., 1., 0., ..., 0., 0., 0.],
                [0., 0., 1., ..., 0., 0., 0.],
                ...,
                [0., 0., 0., ..., 1., 0., 0.],
                [0., 0., 0., ..., 0., 1., 0.],
                [0., 0., 0., ..., 0., 0., 1.]])
```

```
In [18]: ide.shape
```

```
Out[18]: (45, 45)
```

```
In [19]: arr = np.arange(99)
```

```
In [20]: arr
```

```
Out[20]: 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, 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,
                51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67,
                68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84,
                85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98])
```

```
In [21]: arr.reshape(3,33)
```

```
Out[21]: 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, 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, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64,
                65],
               [66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81,
                82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97,
                98]])
```

```
In [22]: arr
```

```
Out[22]: 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, 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,
                51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67,
                68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84,
                85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98])
```

```
In [23]: arr = arr.reshape(3,33)
```

```
In [24]: arr
```

```
Out[24]: 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, 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, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64,
                65],
               [66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81,
                82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97,
                98]])
```

```
In [25]: arr.ravel()
```

```
Out[25]: 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, 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,
                51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67,
                68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84,
                85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98])
```

```
In [26]: arr = arr.ravel()
```

```
In [27]: arr.shape
```

```
Out[27]: (99,)
```

## axis -----> axis 1

||| axis 0

## in 1D array only axis0 will be there

```
In [28]: x = [[1,2,3],[4,5,6],[7,1,0]]
```

```
In [29]: ar = np.array(x)
```

```
In [30]: ar
```

```
Out[30]: array([[1, 2, 3],
                [4, 5, 6],
                [7, 1, 0]])
```

\*\* axis=0 means column wise

```
In [31]: ar.sum(axis=0)
```

```
Out[31]: array([12,  8,  9])
```

axis=1 means row wise

```
In [32]: ar.sum(axis=1)
```

```
Out[32]: array([ 6, 15,  8])
```

## array.T means it will transpose the given array

```
In [33]: ar.T
```

```
Out[33]: array([[1, 4, 7],
                [2, 5, 1],
                [3, 6, 0]])
```

```
In [34]: ar.flat
```

```
Out[34]: <numpy.flatiter at 0x175750ef3e0>
```

```
In [35]: for item in ar.flat:
          print(item)
```

```
1
2
3
4
5
6
7
1
0
```

## arr.ndim shows the number of dimension

```
In [36]: ar.ndim
```

```
Out[36]: 2
```

```
In [37]: ar.size
```

```
Out[37]: 9
```

```
In [38]: ar.nbytes
```

```
Out[38]: 36
```

```
In [39]: one = np.array([1,2,4,543,24])
```

## arr.argmax() give index of max element

```
In [40]: one.argmax()
```

```
Out[40]: 3
```

```
In [41]: one.argmin()
```

```
Out[41]: 0
```

```
In [42]: one.argsort()
```

```
Out[42]: array([0, 1, 2, 4, 3], dtype=int64)
```

```
In [43]: ar
```

```
Out[43]: array([[1, 2, 3],  
               [4, 5, 6],  
               [7, 1, 0]])
```

```
In [44]: ar.argmin()
```

```
Out[44]: 8
```

```
In [45]: ar.argmax()
```

```
Out[45]: 6
```

```
In [46]: ar.argmax(axis=0)
```

```
Out[46]: array([2, 1, 1], dtype=int64)
```

```
In [47]: ar.argmax(axis=1)
```

```
Out[47]: array([2, 2, 0], dtype=int64)
```

```
In [48]: ar.argsort(axis=0)
```

```
Out[48]: array([[0, 2, 2],  
               [1, 0, 0],  
               [2, 1, 1]], dtype=int64)
```

```
In [49]: ar
```

```
Out[49]: array([[1, 2, 3],  
               [4, 5, 6],  
               [7, 1, 0]])
```



```
In [50]: ar.argsort(axis=1)
```

```
Out[50]: array([[0, 1, 2],  
               [0, 1, 2],  
               [2, 1, 0]], dtype=int64)
```

```
In [51]: ar2 = np.array([[3,4,2],[4,2,5],[6,6,2]])
```

```
In [52]: ar2
```

```
Out[52]: array([[3, 4, 2],  
               [4, 2, 5],  
               [6, 6, 2]])
```

```
In [53]: ar+ar2
```

```
Out[53]: array([[ 4,  6,  5],  
               [ 8,  7, 11],  
               [13,  7,  2]])
```

```
In [54]: ar*ar2
```

```
Out[54]: array([[ 3,  8,  6],  
               [16, 10, 30],  
               [42,  6,  0]])
```

```
In [55]: np.sqrt(ar)
```

```
Out[55]: array([[1.          , 1.41421356, 1.73205081],  
               [2.          , 2.23606798, 2.44948974],  
               [2.64575131, 1.          , 0.          ]])
```

```
In [56]: ar.sum()
```

```
Out[56]: 29
```

```
In [57]: ar.min()
```

```
Out[57]: 0
```

```
In [58]: ar
```

```
Out[58]: array([[1, 2, 3],  
               [4, 5, 6],  
               [7, 1, 0]])
```

```
In [59]: np.where(ar>5)
```

```
Out[59]: (array([ 6,  7,  8,  9, 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, 51, 52, 53, 54, 55, 56,  
                57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73,  
                74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90,  
                91, 92, 93, 94, 95, 96, 97, 98], dtype=int64),)
```

```
In [60]: np.where(ar>5)
```

```
Out[60]: (array([1, 2], dtype=int64), array([2, 0], dtype=int64))
```

```
In [61]: np.count_nonzero(ar)
```

```
Out[61]: 8
```

```
In [62]: np.nonzero(ar)
```

```
Out[62]: (array([0, 0, 0, 1, 1, 1, 2, 2], dtype=int64),  
         array([0, 1, 2, 0, 1, 2, 0, 1], dtype=int64))
```

```
In [63]: import sys
```

```
In [65]: py_ar = [0,4,55,2]
```

```
In [66]: np_ar = np.array(py_ar)
```

```
In [67]: sys.getsizeof(1)*len(py_ar)
```

```
Out[67]: 112
```

```
In [69]: np_ar.itemsize*np_ar.size
```

```
Out[69]: 16
```

```
In [ ]:
```