```
import numpy as np
a= [0,'one',2,'three',4]
an = np.array(a)
print(a)
print(an) #observe no commas in numpy array
```

```
[0, 'one', 2, 'three', 4]
['0' 'one' '2' 'three' '4']
```

If even numbers typecasted to string along with other strings

```
an[0]
```

```
'0'
```

```
an=np.array([1,2,3,4,5]) ## elements must be enclosed by []
an
```

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

```
arr=np.arange(5) # arange not arrange
arr
```

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

3d array image visualisation

```
arr=np.array([0,2,4,6,8]) #np.arr wrong
arr
```

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

```
type(arr)
```

```
numpy.ndarray
```

```
arr.dtype
```

```
dtype('int64')
```

```
print(arr,type(arr))
```

```
[0 2 4 6 8] <class 'numpy.ndarray'>
```

```
arr=np.arange(2,11,2)
arr
```

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

```
arr.size
```

```
5
```

```
arr[0]=3
arr
```

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

```
arr.ndim
```

```
1
```

```
arr.shape #no of columns first if 1d
```

```
(5,)
```

```
arr2 = np.array([[0,2,4,6,8],[10,12,14,16,18],[20,22,24,26,28]]) #use ( must)
print(arr2)
print(arr2.ndim)
print(arr2.shape)
print(arr2.size)
```

```
[[ 0  2  4  6  8]
 [10 12 14 16 18]
 [20 22 24 26 28]]
2
(3, 5)
15
```

```
arr3=np.array([[[0,2,4,6,8],
                [10,12,14,16,18]],
               [[0,4,16,36,64],
                [100,144,196,256,324]]
               ])
```

```
print(arr3)
print(arr3.ndim)
print(arr3.shape)
print(arr3.size)
```

```
[[[  0   2   4   6   8]
  [ 10  12  14  16  18]]

 [[  0   4  16  36  64]
  [100 144 196 256 324]]]
3
(2, 2, 5)
20
```

```
arr3[0,1,2:4]
```

```
array([14, 16])
```

```
arr3[:,0,2:4] #4,6 , 16,36
```

```
array([[ 4,  6],
       [16, 36]])
```

```
#get
arr3[:,0:2,4]
#or
arr3[:,:,4]
```

```
array([[  8,  18],
       [ 64, 324]])
```

```
#get
arr3[:,:,2:4]
```

```
array([[[  4,   6],
        [ 14,  16]],

       [[ 16,  36],
        [196, 256]]])
```

✓ 0s completed at 1:48 PM