

In [2]:

```
# Converting normal data into array data  
import numpy as np
```

In [3]:

```
l1 = [1,2,3,4,5]  
v1 = np.array(l1)  
v1
```

Out[3]:

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

In [4]:

```
v2 = np.asarray(l1)  
v2
```

Out[4]:

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

In [5]:

```
v3 = np.asarray(l1, dtype=float)  
v3
```

Out[5]:

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

In [6]:

```
# tuple  
t1 = (1,2,3,4)  
v4 = np.asarray(t1)  
v4
```

Out[6]:

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

In [7]:

```
# tuple inside list  
  
l2 = [(1,2,3),(4,5,6)]  
v5 = np.asarray(l2)  
v5
```

Out[7]:

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

In [ ]:

In [10]:

```
s1 = 'disney world'
for x in s1:
    print(x,end=' ')
```

d i s n e y w o r l d

In [ ]:

In [12]:

```
s = b'disney world'
v6 = np.frombuffer(s, dtype='S1')
v6
```

Out[12]:

```
array([b'd', b'i', b's', b'n', b'e', b'y', b' ', b'w', b'o', b'r', b'l',
       b'd'], dtype='|S1')
```

In [13]:

```
s = b'disney world'
v6 = np.frombuffer(s, dtype='S1', count=7)
v6
```

Out[13]:

```
array([b'd', b'i', b's', b'n', b'e', b'y', b' '], dtype='|S1')
```

In [14]:

```
s = b'disney world'
v6 = np.frombuffer(s, dtype='S1', count=5, offset=7)
v6
```

Out[14]:

```
array([b'w', b'o', b'r', b'l', b'd'], dtype='|S1')
```

In [ ]:

In [15]:

```
# range
r1 = range(5)
r1
```

Out[15]:

```
range(0, 5)
```

In [17]:

```
for x in range(5):
    print(x, end=" ")
```

```
0 1 2 3 4
```

In [18]:

```
l1 = []
for x in range(5):
    l1.append(x)
l1
```

Out[18]:

```
[0, 1, 2, 3, 4]
```

In [19]:

```
l2 = np.array(l1)
l2
```

Out[19]:

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

In [ ]:

In [22]:

```
d1 = range(10)
res = iter(d1)
res
```

Out[22]:

```
<range_iterator at 0x281e3631ed0>
```

In [23]:

```
# iterator to get value from object
x = np.fromiter(res, dtype=int)
x
```

Out[23]:

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

In [ ]:

In [24]:

```
# Numerical Ranges
z1 = np.arange(5)
z1
```

Out[24]:

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

In [25]:

```
z1 = np.arange(5, dtype=float)
z1
```

Out[25]:

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

In [27]:

```
z1 = np.arange(5,50,5, dtype=float)
z1
```

Out[27]:

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

In [28]:

```
z1 = np.arange(5,50,5, dtype=int)
z1
```

Out[28]:

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

In [ ]:

In [ ]:

```
# linspace
```

In [29]:

```
z2 = np.linspace(5,10)
z2
```

Out[29]:

```
array([ 5.          ,  5.10204082,  5.20408163,  5.30612245,  5.40816327,
        5.51020408,  5.6122449  ,  5.71428571,  5.81632653,  5.91836735,
        6.02040816,  6.12244898,  6.2244898  ,  6.32653061,  6.42857143,
        6.53061224,  6.63265306,  6.73469388,  6.83673469,  6.93877551,
        7.04081633,  7.14285714,  7.24489796,  7.34693878,  7.44897959,
        7.55102041,  7.65306122,  7.75510204,  7.85714286,  7.95918367,
        8.06122449,  8.16326531,  8.26530612,  8.36734694,  8.46938776,
        8.57142857,  8.67346939,  8.7755102  ,  8.87755102,  8.97959184,
        9.08163265,  9.18367347,  9.28571429,  9.3877551  ,  9.48979592,
        9.59183673,  9.69387755,  9.79591837,  9.89795918, 10.          ])
```

In [30]:

```
z3 = np.linspace(5,10,5)
z3
```

Out[30]:

```
array([ 5. ,  6.25,  7.5 ,  8.75, 10.  ])
```

In [31]:

```
z4 = np.linspace(5,10,5, endpoint=False)
z4
```

Out[31]:

```
array([5., 6., 7., 8., 9.]
```

In [ ]:

In [32]:

```
x = np.linspace(1,2,5)
x
```

Out[32]:

```
array([1. , 1.25, 1.5 , 1.75, 2.  ])
```

In [33]:

```
x1 = np.linspace(1,5,10)
x1
```

Out[33]:

```
array([1.          , 1.44444444, 1.88888889, 2.33333333, 2.77777778,
        3.22222222, 3.66666667, 4.11111111, 4.55555556, 5.          ])
```

In [36]:

```
x1 = np.linspace(1,2,3, retstep=True)
x1
```

Out[36]:

```
(array([1. , 1.5, 2. ]), 0.5)
```

In [39]:

```
x1 = np.linspace(1,2,3, retstep=False)
x1
```

Out[39]:

```
array([1. , 1.5, 2. ])
```

In [40]:

```
x1 = np.linspace(1,2,3, retstep=False, endpoint=False)
x1
```

Out[40]:

```
array([1.          , 1.33333333, 1.66666667])
```

In [ ]:

In [ ]:

In [ ]:

```
# Indexing => Slicing
```

In [41]:

```
# 1st
y1 = np.arange(10)
res = slice(2,9,2)
y1[res]
```

Out[41]:

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

In [ ]:

In [43]:

```
# 2nd  
y2 = np.arange(10)  
res2 = y2[2:9:2]  
res2
```

Out[43]:

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

In [ ]:

In [44]:

```
y2 = np.arange(10)  
res3 = y2[6]  
res3
```

Out[44]:

```
6
```

In [45]:

```
y2 = np.arange(10)  
res3 = y2[3:]  
res3
```

Out[45]:

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

In [ ]:

In [46]:

```
y2 = np.arange(10)  
res3 = y2[:11]  
res3
```

Out[46]:

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

In [ ]:

In [47]:

```
z1 = np.array([[1,2,3],[4,5,6],[7,8,9]])  
z1
```

Out[47]:

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

In [48]:

```
z1.ndim
```

Out[48]:

```
2
```

In [52]:

```
z2 = np.array([  
                [[1,2,3]],  
                [[4,5,6]],  
                [[7,8,9]]  
              ])  
z2
```

Out[52]:

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

In [53]:

```
z2.ndim
```

Out[53]:

```
3
```

In [55]:

```
z2[0:]
```

Out[55]:

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



In [56]:

```
z2[1:]
```

Out[56]:

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

In [57]:

```
z2[2:]
```

Out[57]:

```
array([[7, 8, 9]])
```

In [59]:

```
z2
```

Out[59]:

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

In [61]:

```
z2[:2]
```

Out[61]:

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

In [65]:

```
z2
```

Out[65]:

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

In [66]:

```
z2[...,0]
```

Out[66]:

```
array([1,  
      4,  
      7])
```

In [67]:

```
z2[...,1]
```

Out[67]:

```
array([[2],  
       [5],  
       [8]])
```

In [68]:

```
z2[...,2]
```

Out[68]:

```
array([[3],  
       [6],  
       [9]])
```

In [69]:

```
z2[...,1:]
```

Out[69]:

```
array([[2, 3],  
       [5, 6],  
       [8, 9]])
```