

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

```
In [2]: print(np.__version__)
```

```
1.24.3
```

```
In [3]: x=10  
print(x,type(x))
```

```
10 <class 'int'>
```

```
In [4]: a=np.array(x)  
print(a,type(a))
```

```
10 <class 'numpy.ndarray'>
```

```
In [5]: a
```

```
Out[5]: array(10)
```

```
In [6]: a.ndim
```

```
Out[6]: 0
```

```
In [9]: a.shape
```

```
Out[9]: ()
```

```
In [10]: x=[10]  
print(x,type(x))
```

```
[10] <class 'list'>
```

```
In [11]: a=np.array(x)  
print(a,type(a))
```

```
[10] <class 'numpy.ndarray'>
```

```
In [12]: a.ndim
```

```
Out[12]: 1
```

```
In [13]: a.shape
```

```
Out[13]: (1,)
```

```
In [16]: r=range(10,25)  
print(r,type(r))
```

```
range(10, 25) <class 'range'>
```

In []:

In []:

```
In [20]: a=np.array(r)
         print(a,type(a),len(a))
```

```
[10 11 12 13 14 15 16 17 18 19 20 21 22 23 24] <class 'numpy.ndarray'> 15
```

```
In [18]: a.ndim
```

```
Out[18]: 1
```

```
In [19]: a.shape
```

```
Out[19]: (15,)
```

```
In [21]: a.shape=(3,5)
         print(a,type(a))
```

```
[[10 11 12 13 14]
 [15 16 17 18 19]
 [20 21 22 23 24]] <class 'numpy.ndarray'>
```

```
In [22]: a.ndim
```

```
Out[22]: 2
```

```
In [23]: a.shape
```

```
Out[23]: (3, 5)
```

```
In [24]: a.shape=(5,3)
```

```
In [25]: a
```

```
Out[25]: array([[10, 11, 12],
                [13, 14, 15],
                [16, 17, 18],
                [19, 20, 21],
                [22, 23, 24]])
```

```
In [26]: a.ndim
```

```
Out[26]: 2
```

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

```
Out[27]: (5, 3)
```

```
In [31]: r=range(10,25)
print(r,type(r))
a=np.array(r,dtype=float)
print(a,type(a),len(a))

range(10, 25) <class 'range'>
[10. 11. 12. 13. 14. 15. 16. 17. 18. 19. 20. 21. 22. 23. 24.] <class 'numpy.ndarray'> 15
```

```
In [32]: a.dtype
```

```
Out[32]: dtype('float64')
```

```
In [33]: r=range(10,25)
print(r,type(r))
a=np.array(r,)
print(a,type(a),len(a))
```

```
range(10, 25) <class 'range'>
[10 11 12 13 14 15 16 17 18 19 20 21 22 23 24] <class 'numpy.ndarray'> 15
```

```
In [34]: a.dtype
```

```
Out[34]: dtype('int32')
```

```
In [35]: lst=[10,20,35,67,12,34]
print(lst,type(lst))
a=np.array(lst)
print(a,type(a))
```

```
[10, 20, 35, 67, 12, 34] <class 'list'>
[10 20 35 67 12 34] <class 'numpy.ndarray'>
```

```
In [36]: print("Dimension=",a.ndim)
print("shape=",a.shape)
print("Data Type=",a.dtype)
```

```
Dimension= 1
shape= (6,)
Data Type= int32
```

```
In [37]: a.shape=(3,2)
```

```
In [39]: print(a)
print("Dimension=",a.ndim)
print("shape=",a.shape)
print("Data Type=",a.dtype)
```

```
[[10 20]
 [35 67]
 [12 34]]
Dimension= 2
shape= (3, 2)
Data Type= int32
```

```
In [40]: a.shape=(2,3)
print("Dimension=",a.ndim)
print("shape=",a.shape)
print("Data Type=",a.dtype)
```

```
Dimension= 2
shape= (2, 3)
Data Type= int32
```

```
In [41]: tpl=(10,20,35,67,12,34)
print(tpl,type(tpl))
a=np.array(tpl)
print(a,type(a))
```

```
(10, 20, 35, 67, 12, 34) <class 'tuple'>
[10 20 35 67 12 34] <class 'numpy.ndarray'>
```

```
In [43]: a.shape=(2,3)
print(a)
print("Dimension=",a.ndim)
print("shape=",a.shape)
print("Data Type=",a.dtype)
```

```
[[10 20 35]
 [67 12 34]]
Dimension= 2
shape= (2, 3)
Data Type= int32
```

```
In [44]: s={10,20,35,67,12,34}
print(s,type(s))
a=np.array(s)
print(a,type(a))
```

```
{34, 35, 67, 20, 10, 12} <class 'set'>
[34, 35, 67, 20, 10, 12] <class 'numpy.ndarray'>
```

```
In [47]: print(a.ndim)
print(a.shape)
```

```
0
()
```

```
In [48]: a
```

```
Out[48]: array({34, 35, 67, 20, 10, 12}, dtype=object)
```

```
In [49]: d1={10:"Apple",20:"Mango",30:"Kiwi",40:"Guava"}
print(d1,type(d1))
```

```
{10: 'Apple', 20: 'Mango', 30: 'Kiwi', 40: 'Guava'} <class 'dict'>
```

```
In [50]: a=np.array(d1)
print(a,type(a))
```

```
{10: 'Apple', 20: 'Mango', 30: 'Kiwi', 40: 'Guava'} <class 'numpy.ndarray'>
```

```
In [51]: a
```

```
Out[51]: array({10: 'Apple', 20: 'Mango', 30: 'Kiwi', 40: 'Guava'}, dtype=object)
```

```
In [53]: print(a.ndim)
print(a.shape)
print(a.dtype)
```

```
0
()
object
```

```
In [54]: lst=[10,20,30,40,50,60,70,80,90,15,25,35,55,65,25,75]
print(lst,type(lst))
```

```
[10, 20, 30, 40, 50, 60, 70, 80, 90, 15, 25, 35, 55, 65, 25, 75] <class 'list'>
```

```
In [55]: a=np.array(lst)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)
print("Data Type=",a.dtype)
```

```
[10 20 30 40 50 60 70 80 90 15 25 35 55 65 25 75] <class 'numpy.ndarray'>
Dimension= 1
shape= (16,)
Data Type= int32
```

```
In [56]: a.shape=(4,4)
print(a)
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[[10 20 30 40]
 [50 60 70 80]
 [90 15 25 35]
 [55 65 25 75]]
Dimension= 2
shape= (4, 4)
```

```
In [57]: a.shape=(8,2)
print(a)
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[[10 20]
 [30 40]
 [50 60]
 [70 80]
 [90 15]
 [25 35]
 [55 65]
 [25 75]]
Dimension= 2
shape= (8, 2)
```

```
In [58]: a.shape=(2,4,2)
print(a)
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[[[10 20]
  [30 40]
  [50 60]
  [70 80]]
```

```
[[[90 15]
  [25 35]
  [55 65]
  [25 75]]]
```

```
Dimension= 3
shape= (2, 4, 2)
```

```
In [59]: a.shape=(2,2,2,2)
print(a)
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[[[[[10 20]
  [30 40]]
```

```
[[[50 60]
  [70 80]]]
```

```
[[[90 15]
  [25 35]]
```

```
[[[55 65]
  [25 75]]]]
```

```
Dimension= 4
shape= (2, 2, 2, 2)
```

```
In [61]: a.shape=(16,)
print(a)
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[10 20 30 40 50 60 70 80 90 15 25 35 55 65 25 75]
```

```
Dimension= 1
shape= (16,)
```

```
In [62]: #Function Name---arange()
#Syntax1: numpy.arange(Value)----->0 to Value-1
#Syntax2: numpy.arange(Begin,End)----->Begin to End-1
#Syntax3: numpy.arange(Begin,End,Step)----->Begin to End-1 with equal Interval with Step
```

```
In [63]: lst=[[10,20,30,40],[50,60,70,80],[90,15,25,35],[55,65,25,75]]
print(lst,type(lst))
a=np.array(lst)
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[[10, 20, 30, 40], [50, 60, 70, 80], [90, 15, 25, 35], [55, 65, 25, 75]] <class 'list'>
Dimension= 2
shape= (4, 4)
```

```
In [64]: #Function Name---arange()
#Syntax1: numpy.arange(Value)----->0 to Value-1
#Syntax2: numpy.arange(Begin,End)----->Begin to End-1
#Syntax3: numpy.arange(Begin,End,Step)----->Begin to End-1 with equal Interval with Step
```

```
In [65]: a=np.arange(9)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[0 1 2 3 4 5 6 7 8] <class 'numpy.ndarray'>
Dimension= 1
shape= (9,)
```

```
In [66]: a.shape=(3,3)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[[0 1 2]
 [3 4 5]
 [6 7 8]] <class 'numpy.ndarray'>
Dimension= 2
shape= (3, 3)
```

```
In [67]: a=np.arange(10,35)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[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] <class 'numpy.ndarray'>
Dimension= 1
shape= (25,)
```

```
In [68]: a.shape=(5,5)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[[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]] <class 'numpy.ndarray'>
Dimension= 2
shape= (5, 5)
```

```
In [69]: a=np.arange(10,51,5)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[10 15 20 25 30 35 40 45 50] <class 'numpy.ndarray'>
Dimension= 1
shape= (9,)
```

```
In [70]: a.shape=(3,3)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[10 15 20]
 [25 30 35]
 [40 45 50]] <class 'numpy.ndarray'>
Dimension= 2
shape= (3, 3)
```

```
In [71]: #Function Name: zeros()
#Syntax: varname=numpy.zeros(shape,dtype)
```

```
In [72]: a=np.zeros(6)
print(a,type(a))

[0. 0. 0. 0. 0. 0.] <class 'numpy.ndarray'>
```

```
In [73]: a=np.zeros(6,dtype=int)
print(a,type(a))

[0 0 0 0 0 0] <class 'numpy.ndarray'>
```

```
In [74]: a.shape=(3,2)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[0 0]
 [0 0]
 [0 0]] <class 'numpy.ndarray'>
Dimension= 2
shape= (3, 2)
```

```
In [75]: a=np.zeros((3,3),int)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[0 0 0]
 [0 0 0]
 [0 0 0]] <class 'numpy.ndarray'>
Dimension= 2
shape= (3, 3)
```



```
In [76]: a=np.zeros((4,5),int)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[0 0 0 0 0]
 [0 0 0 0 0]
 [0 0 0 0 0]
 [0 0 0 0 0]] <class 'numpy.ndarray'>
Dimension= 2
shape= (4, 5)
```

```
In [77]: a=np.zeros((2,4,5),int)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[[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]
  [0 0 0 0 0]
  [0 0 0 0 0]
  [0 0 0 0 0]]] <class 'numpy.ndarray'>
Dimension= 3
shape= (2, 4, 5)
```

```
In [78]: #Function Name: ones()
#Syntax: varname=numpy.ones(shape,dtype)
```

```
In [79]: a=np.ones(8)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[1. 1. 1. 1. 1. 1. 1. 1.] <class 'numpy.ndarray'>
Dimension= 1
shape= (8,)
```

```
In [80]: a=np.ones(8,int)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[1 1 1 1 1 1 1 1] <class 'numpy.ndarray'>
Dimension= 1
shape= (8,)
```

```
In [81]: a.shape=(4,2)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[1 1]
 [1 1]
 [1 1]
 [1 1]] <class 'numpy.ndarray'>
Dimension= 2
shape= (4, 2)
```

```
In [82]: a.shape=(2,2,2)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[[1 1]
  [1 1]]

 [[1 1]
  [1 1]]] <class 'numpy.ndarray'>
Dimension= 3
shape= (2, 2, 2)
```

```
In [84]: a=np.ones((4,3),int)
print(a,type(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[1 1 1]
 [1 1 1]
 [1 1 1]
 [1 1 1]] <class 'numpy.ndarray'>
Dimension= 2
shape= (4, 3)
```

```
In [86]: a=np.ones((3,4,3),int)
print(a,type(a),len(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[[1 1 1]
  [1 1 1]
  [1 1 1]
  [1 1 1]]

 [[1 1 1]
  [1 1 1]
  [1 1 1]
  [1 1 1]]

 [[1 1 1]
  [1 1 1]
  [1 1 1]
  [1 1 1]]] <class 'numpy.ndarray'> 3
Dimension= 3
shape= (3, 4, 3)
```

```
In [87]: #Function Name: full()
#Syntax: varname=np.full(shape,fill_value,dtype)
```

```
In [88]: a=np.full(4,fill_value=8)
print(a,type(a),len(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[8 8 8 8] <class 'numpy.ndarray'> 4
Dimension= 1
shape= (4,)
```

```
In [89]: a.shape=(2,2)
print(a,type(a),len(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[[8 8]
 [8 8]] <class 'numpy.ndarray'> 2
Dimension= 2
shape= (2, 2)
```

```
In [90]: a=np.full((4,4),fill_value="KVR")
print(a,type(a),len(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[[ 'KVR' 'KVR' 'KVR' 'KVR' ]
 [ 'KVR' 'KVR' 'KVR' 'KVR' ]
 [ 'KVR' 'KVR' 'KVR' 'KVR' ]
 [ 'KVR' 'KVR' 'KVR' 'KVR' ]] <class 'numpy.ndarray'> 4
Dimension= 2
shape= (4, 4)
```

```
In [91]: a=np.full((4,4),fill_value=4)
print(a,type(a),len(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)
```

```
[[4 4 4 4]
 [4 4 4 4]
 [4 4 4 4]
 [4 4 4 4]] <class 'numpy.ndarray'> 4
Dimension= 2
shape= (4, 4)
```

```
In [92]: a=np.full((2,4,4),fill_value=9)
print(a,type(a),len(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[[9 9 9 9]
  [9 9 9 9]
  [9 9 9 9]
  [9 9 9 9]]

  [[9 9 9 9]
  [9 9 9 9]
  [9 9 9 9]
  [9 9 9 9]]] <class 'numpy.ndarray'> 2
Dimension= 3
shape= (2, 4, 4)
```

```
In [93]: #Function Name: identity()
#Syntax: varname=numpy.identity(N,dtype)
```

```
In [95]: a=np.identity(3,int)
print(a,type(a),len(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[1 0 0]
 [0 1 0]
 [0 0 1]] <class 'numpy.ndarray'> 3
Dimension= 2
shape= (3, 3)
```

```
In [96]: a=np.identity(4,int)
print(a,type(a),len(a))
print("Dimension=",a.ndim)
print("shape=",a.shape)

[[1 0 0 0]
 [0 1 0 0]
 [0 0 1 0]
 [0 0 0 1]] <class 'numpy.ndarray'> 4
Dimension= 2
shape= (4, 4)
```

```
In [97]: print(a[0,0]*a[1,1]*a[2,2]*a[3,3])

1
```

```
In [98]: #Function Name: hstack()
#Syntax: varname=numpy.hstack(ndarrayobj1,ndarrayobj1)
```

```
In [100]: a=np.array([[10,20,30],[40,50,60]])
          print(a,type(a))
          print(a.ndim)
          print(a.shape)

          [[10 20 30]
           [40 50 60]] <class 'numpy.ndarray'>
          2
          (2, 3)
```

```
In [101]: b=np.array([[100,200],[400,500]])
          print(b,type(b))
          print(b.ndim)
          print(b.shape)

          [[100 200]
           [400 500]] <class 'numpy.ndarray'>
          2
          (2, 2)
```

```
In [102]: c=np.hstack((a,b))
          print(c,type(c))
          print(c.ndim)
          print(c.shape)

          [[ 10  20  30 100 200]
           [ 40  50  60 400 500]] <class 'numpy.ndarray'>
          2
          (2, 5)
```

```
In [103]: c=np.hstack((b,a))
          print(c,type(c))
          print(c.ndim)
          print(c.shape)

          [[100 200  10  20  30]
           [400 500  40  50  60]] <class 'numpy.ndarray'>
          2
          (2, 5)
```

```
In [105]: a=np.array([[10,20,30],[40,50,60]])
          print(a,type(a))
          print(a.ndim)
          print(a.shape)

          [[10 20 30]
           [40 50 60]] <class 'numpy.ndarray'>
          2
          (2, 3)
```

```
In [106]: b=np.array([[10,20,30],[40,50,60],[70,80,90]])
print(b,type(b))
print(b.ndim)
print(b.shape)
```

```
[[10 20 30]
 [40 50 60]
 [70 80 90]] <class 'numpy.ndarray'>
2
(3, 3)
```

```
In [107]: c=np.vstack((a,b))
print(c,type(c))
print(c.ndim)
print(c.shape)
```

```
[[10 20 30]
 [40 50 60]
 [10 20 30]
 [40 50 60]
 [70 80 90]] <class 'numpy.ndarray'>
2
(5, 3)
```

```
In [108]: c=np.vstack((b,a))
print(c,type(c))
print(c.ndim)
print(c.shape)
```

```
[[10 20 30]
 [40 50 60]
 [70 80 90]
 [10 20 30]
 [40 50 60]] <class 'numpy.ndarray'>
2
(5, 3)
```

```
In [ ]:
```