numpytask1lb

June 2, 2024

[2]:
```
import numpy as np
```

[3]:
```
#Qus1
a_np=np.array([1,2,3,4,5,6,7,8,9,10])
b_np=np.array([11,12,13,14,15,16,17,18,19,20])
print('1:',a_np)
print('2:',b_np)
```

```
1: [ 1  2  3  4  5  6  7  8  9 10]
2: [11 12 13 14 15 16 17 18 19 20]
```

[4]:
```
#Qus2
a_np=np.array([[1,2,3,4,5,6,7,8,9,10]])
b_np=np.array([[11,12,13,14,15,16,17,18,19,20]])
print('1:',a_np)
print('2:',b_np)
```

```
1: [[ 1  2  3  4  5  6  7  8  9 10]]
2: [[11 12 13 14 15 16 17 18 19 20]]
```

[5]:
```
#Qus3
a_np=np.array([[[1,2,3],[4,5,6],[7,8,9]]])
b_np=np.array([[[1,2,3],[4,5,6],[7,8,9]]])
print('1:',a_np)
print('2:',b_np)
```

```
1: [[[1 2 3]
  [4 5 6]
  [7 8 9]]]
2: [[[1 2 3]
  [4 5 6]
  [7 8 9]]]
```

[6]:
```
#Qus4
a_np=np.array([1,2,3,4,5])
b_np=np.array([[1,2,3,4,5],[6,7,8,9,0]])
c_np=np.array([[[1,2,3],[4,5,6],[7,8,9]]])
print(a_np.ndim)
print('------------------')
print(b_np.ndim)
print('--------------------')
print(c_np.ndim)
```

```
1
------------------
2
--------------------
3
```

[7]:

```
a_np=np.array([1,2,3,4,5])
b_np=np.array([[1,2,3,4,5],[6,7,8,9,0]])
c_np=np.array([[[1,2,3],[4,5,6],[7,8,9]]])
print(a_np.shape)
print('------------------')
print(b_np.shape)
print('--------------------')
print(c_np.shape)
```

```
(5,)
------------------
(2, 5)
--------------------
(1, 3, 3)
```

[8]:

```
a_np=np.array([1,2,3,4,5])
b_np=np.array([[1,2,3,4,5],[6,7,8,9,0]])
c_np=np.array([[[1,2,3],[4,5,6],[7,8,9]]])
print(a_np.size)
print('------------------')
print(b_np.size)
print('--------------------')
print(c_np.size)
```

```
5
------------------
10
--------------------
9
```

[9]:

```
a_np=np.array([1,2,3,4,5])
b_np=np.array([[1,2,3,4,5],[6,7,8,9,0]])
c_np=np.array([[[1,2,3],[4,5,6],[7,8,9]]])
print(a_np.dtype)
print('------------------')
print(b_np.dtype)
print('--------------------')
print(c_np.dtype)
```

```
int32
------------------
int32
--------------------
int32
```

[22]:

```
#shape manipulation
#ravel
a_np=np.array([10,20,30,40,50,60,70,80,90,100])
print('1d:',a_np)
print(a_np.ravel())
print('-------------------------')
b_np=np.array([[10,20,30,40,50],[60,70,80,90,100]])
print('2d:',b_np)
print(b_np.ravel())
print('------------------------')
c_np=np.array([[[10,20,30],[40,50,60],[70,80,90]]])
print('3d:',c_np)
print(c_np.ravel())
print('------------------------')
```

```
1d: [ 10  20  30  40  50  60  70  80  90 100]
[ 10  20  30  40  50  60  70  80  90 100]
-------------------------
2d: [[ 10  20  30  40  50]
 [ 60  70  80  90 100]]
[ 10  20  30  40  50  60  70  80  90 100]
------------------------
3d: [[[10 20 30]
  [40 50 60]
  [70 80 90]]]
[10 20 30 40 50 60 70 80 90]
------------------------
```

[42]:

```
#reshape
a_np=np.array([[10,20,30,40],[60,70,80,90],[110,120,130,140]])

print(a_np.reshape(1,12))
print('----------------------------------------')

print(a_np.reshape(2,6))
print('--------------------------------------')

print(a_np.reshape(3,4))
print('---------------------------------------')

print(a_np.reshape(4,3))
print('-------------------------------------')

print(a_np.reshape(6,2))
print('------------------------------------')

print(a_np.reshape(12,1))
```

```
[[ 10  20  30  40  60  70  80  90 110 120 130 140]]
----------------------------------------
[[ 10  20  30  40  60  70]
 [ 80  90 110 120 130 140]]
--------------------------------------
[[ 10  20  30  40]
 [ 60  70  80  90]
 [110 120 130 140]]
---------------------------------------
[[ 10  20  30]
 [ 40  60  70]
 [ 80  90 110]
 [120 130 140]]
-------------------------------------
[[ 10  20]
 [ 30  40]
 [ 60  70]
 [ 80  90]
 [110 120]
 [130 140]]
------------------------------------
[[ 10]
 [ 20]
 [ 30]
 [ 40]
 [ 60]
 [ 70]
 [ 80]
 [ 90]
 [110]
 [120]
 [130]
 [140]]
```

[46]:

```
#resize
a_np=np.array([[10,20,30,40],[50,60,70,80]])
```

```
a_np.resize(1,8)
print(a_np)
print('---------------')
a_np.resize(2,4)
print(a_np)
print('---------------')
a_np.resize(4,2)
print(a_np)
print('----------------')
a_np.resize(8,1)
print(a_np)
```

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

[54]:

```
#horizontal splitting
a_np=np.array([[10,20,30,40,50,60],[70,80,90,100,110,120]])
(np.hsplit(a_np,2))
```

[54]:

```
[array([[10, 20, 30],
        [70, 80, 90]]),
 array([[ 40,  50,  60],
        [100, 110, 120]])]
```

[55]:

```
#hstack
a_np=np.array([10,20,30,40,50,60])
b_np=np.array([70,80,90,100,110,120])
np.hstack([a_np,b_np])
```

[55]:

```
array([ 10,  20,  30,  40,  50,  60,  70,  80,  90, 100, 110, 120])
```

[56]:
```
a_np*b_np
```

[56]:
```
array([ 700, 1600, 2700, 4000, 5500, 7200])
```

[65]:
```
#apply all broadcasting funtions in created 1d arrays
#tile
t=np.tile(np.arange(0,60,5),(1,))
t
```

[65]:
```
array([ 0,  5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55])
```

[67]:
```
#arrange
a=np.arange(0,60,5)
a
```

[67]:
```
array([ 0,  5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55])
```

[74]:
```
#common array funtions
#ones
a=np.ones((5,5))
a
```

[74]:
```
array([[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.]])
```

[75]:
```
#zeros
a=np.zeros((5,5))
a
```

[75]:
```
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.]])
```

[76]:
```
#eye
a=np.eye(3)
a
```

[76]:
```
array([[1., 0., 0.],
       [0., 1., 0.],
       [0., 0., 1.]])
```

[79]:
```
#diag
a=np.diag(np.arange(0,60,5))
a
```

[79]:
```
array([[ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  5,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0, 10,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0, 15,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0, 20,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0, 25,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0, 30,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0, 35,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  0, 40,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  0,  0, 45,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0, 50,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0, 55]])
```

[80]:
```
#create two 2d arrays and perform dot matrix multilpication using zeros and one
a_np=np.array([[0,1,1],[1,0,1],[1,1,0]])
b_np=np.array([[1,0,1],[0,1,0],[1,1,1]])
np.dot(a_np,b_np)
```

[80]:
```
array([[1, 2, 1],
       [2, 1, 2],
       [1, 1, 1]])
```

[ ]:
```

```