In [ ]:

```
# numpy
#  numpy = > Numerical Python
#  fundamental python conatins packages for scientific computing  and Prog
```

In [1]:

```
# eye()
import numpy as np
help(np.eye)
```

...

In [2]:

```
v1 =  np.eye(5)
v1
```

Out[2]:

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

In [3]:

```
v2 = np.eye(2,3)
v2
```

Out[3]:

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

In [4]:

```
v3 = np.eye(4,k=-1)
v3
```

Out[4]:

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

In [5]:

```
v4 =  np.eye(5,k=2)
v4
```

Out[5]:

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

In [6]:

```
v5 =  np.eye(5,k=2,dtype=int)
v5
```

Out[6]:

```
array([[0, 0, 1, 0, 0],
       [0, 0, 0, 1, 0],
       [0, 0, 0, 0, 1],
       [0, 0, 0, 0, 0],
       [0, 0, 0, 0, 0]])
```

In [ ]:

```

```

In [7]:

```
# identity
v6 = np.identity(4)
v6
```

Out[7]:

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

In [8]:

```
v7 = np.identity(5, dtype=int)
v7
```

Out[8]:

```
array([[1, 0, 0, 0, 0],
       [0, 1, 0, 0, 0],
       [0, 0, 1, 0, 0],
       [0, 0, 0, 1, 0],
       [0, 0, 0, 0, 1]])
```

In [ ]:

```

```

In [ ]:

```
# Numpy Random  Module
# rand()  => 0 -1
# randn() => - infi  - + infi
# randf()
# randint()
```

In [9]:

```
help(np.random)
```

...

In [10]:

```
help(np.random.rand)
```

...

In [12]:

```
v7 = np.random.rand(5)
v7
```

Out[12]:

```
array([0.03042979, 0.84355375, 0.41053667, 0.59191602, 0.09556419])
```

In [13]:

```
v8 = np.random.rand(4,5)
v8
```

Out[13]:

```
array([[0.95959575, 0.65789464, 0.19428782, 0.13399255, 0.75307587],
       [0.76831324, 0.11985816, 0.76865058, 0.80369408, 0.13060298],
       [0.99772264, 0.88733023, 0.17982918, 0.51160285, 0.50136189],
       [0.57016402, 0.2596785 , 0.32685243, 0.9830847 , 0.66023548]])
```

In [15]:

```
v9 = np.random.randn(5)
v9
```

Out[15]:

```
array([ 0.34054815, -1.09464063,  1.21499168, -0.11927894,  0.432467  ])
```

In [16]:

```
v10 = np.random.randn(4,5)
v10
```

Out[16]:

```
array([[-1.64846449, -1.1482998 ,  0.672524  , -0.91105169, -0.78702243],
       [-0.02276944,  1.48506673, -0.97449367, -0.34528025,  1.89161069],
       [ 0.20301118,  0.79650724,  0.20554235,  0.38607109, -2.687499  ],
       [ 1.57676067, -0.90897321, -1.08119731, -1.43508913,  0.6968368 ]])
```

In [ ]:

```

```

In [ ]:

```
# Import func
```

In [17]:

```
s1 = np.array([1,2,3,4])
s1
```

Out[17]:

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

In [18]:

```
s1.ndim
```

Out[18]:

```
1
```

In [19]:

```
s1.shape
```

Out[19]:

```
(4,)
```

In [21]:

```
s1.size
```

Out[21]:

```
4
```

In [ ]:

```

```

In [27]:

```
s2 = np.array([[[1,2,3]],[[4,5,6]]])
s2
```

Out[27]:

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

       [[4, 5, 6]]])
```

In [28]:

```
s2.ndim
```

Out[28]:

```
3
```

In [29]:

```
s2.shape
```

Out[29]:

```
(2, 1, 3)
```

In [30]:

```
s2.size
```

Out[30]:

```
6
```

In [33]:

```
s3 = np.zeros((2,3,4))
s3
```

. . .

In [32]:

```
s3.shape
```

Out[32]:

```
(2, 3, 4)
```

In [34]:

```
s3.size
```

Out[34]:

```
24
```

In [41]:

```
s3.dtype
```

Out[41]:

```
dtype('float64')
```

In [40]:

```
s3.itemsize
```

Out[40]:

```
8
```

In [ ]:

```

```

In [35]:

```
b = np.array([[1,2],[3,4]])
b
```

Out[35]:

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

In [36]:

```
b.shape
```

Out[36]:

```
(2, 2)
```

In [37]:

```
b.size
```

Out[37]:

```
4
```

In [38]:

```
b.dtype
```

Out[38]:

```
dtype('int32')
```

In [39]:

```
b.itemsize
```

Out[39]:

```
4
```

In [ ]:

```
# float64  : 64/8 = 8byte
# int32    : 32/8 = 4byte
```

In [ ]:

```

```

In [ ]:

```
# Data type
# Numerical Data Type
# 1 Boolean
# 2 Integer
# 3 Unsigned Integer
# 4 Float
# 5 complex
```

In [ ]:

```

```

In [43]:

```
# index
# 2d
c = np.array([[1,2],[3,4],[5,6]])
c
```

Out[43]:

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

In [44]:

```
c[0][0]
```

Out[44]:

```
1
```

In [45]:

```
c[-2][-2]
```

Out[45]:

```
3
```

In [46]:

```
c[1]
```

Out[46]:

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

In [ ]:

```
```

In [48]:

```
# 3d Array

x1 = [[
        [1,2,3,4],
        [5,6,7,8],
        [9,10,11,12]
      ],
      [
        [13,14,15,16],
        [17,18,19,20],
        [21,22,23,24]
      ]
    ]
x1
```

Out[48]:

```
[[[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]],
 [[13, 14, 15, 16], [17, 18, 19, 20], [21, 22, 23, 24]]]
```

In [50]:

```
x2 = np.array(x1)
x2
```

Out[50]:

```
array([[[ 1,  2,  3,  4],
        [ 5,  6,  7,  8],
        [ 9, 10, 11, 12]],

       [[13, 14, 15, 16],
        [17, 18, 19, 20],
        [21, 22, 23, 24]]])
```

In [51]:

```
x2.ndim
```

Out[51]:

```
3
```

In [52]:

```
# x2[matrix][row][column]

x2[0][0][1]
```

Out[52]:

```
2
```

In [53]:

```
x2[1][0][2]
```

Out[53]:

```
15
```

In [54]:

```
x2[-2][-3][-2]
```

Out[54]:

```
3
```

In [55]:

```
x2[0][2][3]
```

Out[55]:

```
12
```

In [56]:

```
x2[0]
```

Out[56]:

```
array([[ 1,  2,  3,  4],
       [ 5,  6,  7,  8],
       [ 9, 10, 11, 12]])
```

In [57]:

```
x2[:]
```

Out[57]:

```
array([[[ 1,  2,  3,  4],
        [ 5,  6,  7,  8],
        [ 9, 10, 11, 12]],

       [[13, 14, 15, 16],
        [17, 18, 19, 20],
        [21, 22, 23, 24]]])
```

In [58]:

```
x2[0][1]
```

Out[58]:

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

In [ ]:

```

```

In [ ]:

```
# arry_name[start : end : step]
# slicing
```

In [59]:

```
# 1d array
a = np.array([1,2,3,4,5])
a
```

Out[59]:

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

In [60]:

```
a[1:4]
```

Out[60]:

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

In [61]:

```
a[:]
```

Out[61]:

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

In [62]:

```
a[1 :: 2]
```

Out[62]:

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

In [63]:

```
a[:3]
```

Out[63]:

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

In [ ]:

```

```

In [64]:

```
# 2d array

z1 = np.array([

    [1,2],
    [3,4],
    [5,6]
])
z1
```

Out[64]:

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

In [65]:

```
z1[1: , 1:]
```

Out[65]:

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

In [67]:

```
z1[: , :]
```

Out[67]:

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

In [ ]:

```

```

In [68]:

```
z2 = np.array([

    [1,2,3,4],
    [5,6,7,8],
    [9,10,11,12]
])
z2
```

Out[68]:

```
array([[ 1,  2,  3,  4],
       [ 5,  6,  7,  8],
       [ 9, 10, 11, 12]])
```

In [69]:

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

Out[69]:

```
array([[ 6,  7,  8],
       [10, 11, 12]])
```

In [70]:

```
z2[::,::2]
```

Out[70]:

```
array([[ 1,  3],
       [ 5,  7],
       [ 9, 11]])
```

In [72]:

```
z2[::,::3]
```

Out[72]:

```
array([[ 1,  4],
       [ 5,  8],
       [ 9, 12]])
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: