NUMPY

In [ ]:

```
#install library numpy if vscode
#pip install numpy
```

In [1]:

```
#import libarary
import numpy as np
```

In [2]:

```
#make an array
a=np.array([5,5,5,5,5])
a
```

Out[2]:

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

In [3]:

```
#see type
type(a)
```

Out[3]:

```
numpy.ndarray
```

In [4]:

```
b=np.array([[5,5,5,5,5],[5,5,5,5,5],[5,5,5,5,5]])
b
```

Out[4]:

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

In [5]:

```
len(a)
```

Out[5]:

```
5
```

In [6]:

```
len(b[1])
```

Out[6]:

```
5
```

In [7]:

```
b[1]
```

Out[7]:

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

In [8]:

```
b[0:]
```

Out[8]:

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

In [9]:

```
c=np.zeros(4)
c
```

Out[9]:

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

In [10]:

```
d=np.ones(6)
d
```

Out[10]:

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

In [11]:

```
e= np.empty(5)
e
```

Out[11]:

```
array([1.93101617e-312, 1.23075756e-312, 2.45068304e+198, 2.45942347e+198,
       1.27966391e-152])
```

In [12]:

```
#range of elements
f=np.arange(6)
f
```

Out[12]:

```
array([0, 1, 2, 3, 4, 5])
```

In [13]:

```
##range strat, end
g=np.arange(6,20)
g
```

Out[13]:

```
array([ 6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19])
```

In [ ]:

In [14]:

```
#range strat, end, gap
f=np.arange(3,30,3)
f
```

Out[14]:

```
array([ 3,  6,  9, 12, 15, 18, 21, 24, 27])
```

In [15]:

```
#linearly spaced arrays
h=np.linspace(0,20,num=5)
h
```

Out[15]:

```
array([ 0.,  5., 10., 15., 20.])
```

In [16]:

```
i=np.ones(5,dtype=int)
i
```

Out[16]:

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

In [17]:

```
c=np.zeros(4,dtype=int)
c
```

Out[17]:

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

In [18]:

```
i=np.ones(5,dtype=float)
i
```

Out[18]:

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

In [19]:

```
#2Darray
i=np.ones((5,2))
i
```

Out[19]:

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

In [20]:

```
j=np.zeros((2,3))
j
```

Out[20]:

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

In [21]:

```
j=np.empty((2,3))
j
```

Out[21]:

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

In [22]:

```
#3d array
c=np.arange(24).reshape(2,3,4)
c
```

Out[22]:

```
array([[[ 0,  1,  2,  3],
        [ 4,  5,  6,  7],
        [ 8,  9, 10, 11]],

       [[12, 13, 14, 15],
        [16, 17, 18, 19],
        [20, 21, 22, 23]]])
```

In [23]:

```
bag=np.array(["pen","pencil","rub","shopner","paper","book","board","tissue"])
bag
```

Out[23]:

```
array(['pen', 'pencil', 'rub', 'shopner', 'paper', 'book', 'board',
       'tissue'], dtype='<U7')
```

In [24]:

```
price=np.array([10,5,5,5,2,20,15,15])
price
```

Out[24]:

```
array([10,  5,  5,  5,  2, 20, 15, 15])
```

In [25]:

```
type(bag)
```

Out[25]:

```
numpy.ndarray
```

In [26]:

```
type(price)
```

Out[26]:

```
numpy.ndarray
```

In [27]:

```
len(bag)
```

Out[27]:
```
8
```

In [28]:
```
bag[0:]
```

Out[28]:
```
array(['pen', 'pencil', 'rub', 'shopner', 'paper', 'book', 'board',
       'tissue'], dtype='<U7')
```

In [29]:
```
price.max()
```

Out[29]:
```
20
```

In [30]:
```
price.mean()
```

Out[30]:
```
9.625
```

In [31]:
```
tb=np.arange(6,66,6)
tb
```

Out[31]:
```
array([ 6, 12, 18, 24, 30, 36, 42, 48, 54, 60])
```

In [32]:
```
hn=np.linspace(5,50,num=10,dtype=int)
hn
```

Out[32]:
```
array([ 5, 10, 15, 20, 25, 30, 35, 40, 45, 50])
```

In [33]:
```
np.ones(20,dtype=np.float64)
```

Out[33]:
```
array([1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
       1., 1., 1.])
```

In [34]:
```
np.ones(20,dtype=np.int64)
```

Out[34]:
```
array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
      dtype=int64)
```

In [35]:
```
np.ones(20,dtype=int)
```

Out[35]:
```
array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1])
```

In [36]:
```
np.ones(20,dtype=float)
```

Out[36]:
```
array([1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
       1., 1., 1.])
```

Array functions

In [37]:
```
a=np.array([34,76,96,4,9.64,87,39])
a
```

Out[37]:
```
array([34.  , 76.  , 96.  ,  4.  ,  9.64, 87.  , 39.  ])
```

In [38]:
```
a.sort()
a
```

Out[38]:
```
array([ 4.  ,  9.64, 34.  , 39.  , 76.  , 87.  , 96.  ])
```

In [39]:
```
b=np.array([67,98
            ,45,6.5,54.9])
b
```

Out[39]:
```
array([67. , 98. , 45. ,  6.5, 54.9])
```

In [40]:
```
c=np.concatenate((a,b))
c
```

Out[40]:
```
array([ 4.  ,  9.64, 34.  , 39.  , 76.  , 87.  , 96.  , 67.  , 98.  ,
       45.  ,  6.5 , 54.9 ])
```

In [41]:
```
c.sort()
c
```

Out[41]:
```
array([ 4.  ,  6.5 ,  9.64, 34.  , 39.  , 45.  , 54.9 , 67.  , 76.  ,
       87.  , 96.  , 98.  ])
```

2d arrays

In [42]:
```
a=np.array([[5,6,7,8],[8,5,4,3]])
a
```

Out[42]:
```
array([[5, 6, 7, 8],
       [8, 5, 4, 3]])
```

In [43]:
```
b=np.array([[5,5,7,8],[8,9,2,1]])
b
```

Out[43]:
```
array([[5, 5, 7, 8],
       [8, 9, 2, 1]])
```

In [44]:
```
c=np.concatenate((a,b),axis=0)
c
```

Out[44]:
```
array([[5, 6, 7, 8],
       [8, 5, 4, 3],
       [5, 5, 7, 8],
       [8, 9, 2, 1]])
```

In [45]:
```
c=np.concatenate((a,b),axis=1)
c
```

Out[45]:
```
array([[5, 6, 7, 8, 5, 5, 7, 8],
       [8, 5, 4, 3, 8, 9, 2, 1]])
```

In [46]:
```
##3d array
```

In [47]:
```
f=np.array([[[2,3,4],[4,5,6]],
            [[6,8,9],[7,8,9]],
              [[1,2,3],[5,3,2]]])
f
```

Out[47]:

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

       [[6, 8, 9],
        [7, 8, 9]],

       [[1, 2, 3],
        [5, 3, 2]]])
```

In [48]:

```
#find dimension of array
f.ndim
```

Out[48]:

```
3
```

In [49]:

```
#numer of elements
f.size
```

Out[49]:

```
18
```

In [50]:

```
#dimension,rows,columns
f.shape
```

Out[50]:

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

In [51]:

```
a=np.arange(9)
a
```

Out[51]:

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

In [52]:

```
#3*3=9
a.reshape(3,3)
```

Out[52]:

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

In [53]:

```
np.reshape(a,newshape=(1,9),order='C')
```

Out[53]:

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

In [54]:

```
#convert 1d into 2d
a
```

Out[54]:

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

In [57]:

```
a=np.arange(1,10)
a
```

Out[57]:

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

In [58]:
```
a.shape
```
Out[58]:
```
(9,)
```

In [59]:
```
#row wise
b=a[np.newaxis,:]
b
```
Out[59]:
```
array([[1, 2, 3, 4, 5, 6, 7, 8, 9]])
```

In [60]:
```
b.shape
```
Out[60]:
```
(1, 9)
```

In [61]:
```
#columnwise
c=a[:,np.newaxis]
c
```
Out[61]:
```
array([[1],
       [2],
       [3],
       [4],
       [5],
       [6],
       [7],
       [8],
       [9]])
```

In [62]:
```
a
```
Out[62]:
```
array([1, 2, 3, 4, 5, 6, 7, 8, 9])
```

In [63]:
```
a[2:7]
```
Out[63]:
```
array([3, 4, 5, 6, 7])
```

In [64]:
```
a*2
```
Out[64]:
```
array([ 2,  4,  6,  8, 10, 12, 14, 16, 18])
```

In [65]:
```
a+4
```
Out[65]:
```
array([ 5,  6,  7,  8,  9, 10, 11, 12, 13])
```

In [66]:
```
a.sum()
```
Out[66]:
```
45
```

In [67]:
```
a.mean()
```
Out[67]:
```
5.0
```

In [68]:

```
a.min()
```

Out[68]:

```
1
```

In [70]:

```
a.size
```

Out[70]:

```
9
```