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

In [36]:
```
import sys
sys.version
```

Out[36]:
```
'3.12.7 | packaged by Anaconda, Inc. | (main, Oct  4 2024, 13:17:27) [MSC v.192
9 64 bit (AMD64)]'
```

In [37]:
```
import numpy as np
```

In [38]:
```
np.__version__
```

Out[38]:
```
'1.26.4'
```

In [39]:
```
# creating the list
mylist=[0,1,2,3,4,5]
mylist
```

Out[39]:
```
[0, 1, 2, 3, 4, 5]
```

In [40]:
```
type(mylist)
```

Out[40]:
```
list
```

In [41]:
```
arr= np.array(mylist)      # creating the array in list
arr
```

Out[41]:
```
array([0, 1, 2, 3, 4, 5])
```

In [42]:
```
type(arr)      # ndarray is the multi dimensional array
```

Out[42]:
```
numpy.ndarray
```

In [43]:
```
print(type(arr))
print(type(mylist))
```

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

In [44]:
```
np.arange(10)       # 1d array
```

Out[44]:
```
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

In [45]:
```
np.arange(10,20)
```

Out[45]:
```
array([10, 11, 12, 13, 14, 15, 16, 17, 18, 19])
```

In [46]:
```
np.arange(10,50,5)     # start,stop,step
```

Out[46]:
```
array([10, 15, 20, 25, 30, 35, 40, 45])
```

In [47]:
```
np.arange(10,30,3)         # the Arange is only apply in numpy
```

Out[47]:
```
array([10, 13, 16, 19, 22, 25, 28])
```

In [48]:
```
np.arange(10,20,5,2)     # we can only assian the three arguments only
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[48], line 1
----> 1 np.arange(10,20,5,2)

TypeError: Cannot interpret '2' as a data type
```

In [49]:
```
np.arange(20,8)    # here the the arguments are first is highest values it cant
```

Out[49]:
```
array([], dtype=int32)
```

In [50]:
```
np.arange(8,20)     # here the arguments is the first one is small value it can
```

Out[50]:
```
array([ 8,  9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19])
```

In [51]:
```
np.arange(-30,8)
```

Out[51]:
```
array([-30, -29, -28, -27, -26, -25, -24, -23, -22, -21, -20, -19, -18,
       -17, -16, -15, -14, -13, -12, -11, -10,  -9,  -8,  -7,  -6,  -5,
        -4,  -3,  -2,  -1,   0,   1,   2,   3,   4,   5,   6,   7])
```

In [52]:
```
n=np.arange(-30,8)
n
```

Out[52]:
```
array([-30, -29, -28, -27, -26, -25, -24, -23, -22, -21, -20, -19, -18,
       -17, -16, -15, -14, -13, -12, -11, -10,  -9,  -8,  -7,  -6,  -5,
        -4,  -3,  -2,  -1,   0,   1,   2,   3,   4,   5,   6,   7])
```

zeros()

In [53]:
```
np.zeros(3)
```

Out[53]:
```
array([0., 0., 0.])
```

In [54]:
```
np.zeros(3,dtype=int)
```

Out[54]:
```
array([0, 0, 0])
```

In [55]:
```
z=np.zeros(5)
z
```

Out[55]:
```
array([0., 0., 0., 0., 0.])
```

In [56]:
```
np.zeros((5,3))
```

Out[56]:
```
array([[0., 0., 0.],
       [0., 0., 0.],
       [0., 0., 0.],
       [0., 0., 0.],
       [0., 0., 0.]])
```

In [57]:
```
np.zeros((2,2))    #2d array
```

Out[57]:
```
array([[0., 0.],
       [0., 0.]])
```

In [58]:
```
np.zeros((3,3),dtype=int)    # creating the matrix in int values
                            #3x3  first is row and second is col
```

Out[58]:
```
array([[0, 0, 0],
       [0, 0, 0],
       [0, 0, 0]])
```

In [59]:
```
nd=np.zeros((5,9),dtype=int)
nd
```

Out[59]:
```
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, 0, 0],
       [0, 0, 0, 0, 0, 0, 0, 0, 0],
       [0, 0, 0, 0, 0, 0, 0, 0, 0]])
```

Ones() function

In [60]:
```
np.ones(3)
```

Out[60]:
```
array([1., 1., 1.])
```

In [61]:
```
np.ones((3),dtype=int)
```

Out[61]:
```
array([1, 1, 1])
```

In [62]:
```
np.ones((10,10),dtype=int)
```

Out[62]:
```
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, 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, 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, 1, 1, 1]])
```

In [63]:
```
nd1=np.ones((10,10),dtype=int)
nd1
```

Out[63]:
```
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, 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, 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, 1, 1, 1]])
```

In [64]:
```
np.three(3)    #module 'numpy' has no attribute 'three'
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[64], line 1
----> 1 np.three(3)

File ~\anaconda3\Lib\site-packages\numpy\__init__.py:333, in __getattr__(attr)
    330     "Removed in NumPy 1.25.0"
    331     raise RuntimeError("Tester was removed in NumPy 1.25.")
--> 333 raise AttributeError("module {!r} has no attribute "
    334                      "{!r}".format(__name__, attr))

AttributeError: module 'numpy' has no attribute 'three'
```

In [65]:
```
nd1
```

Out[65]:
```
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, 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, 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, 1, 1, 1]])
```

random.rand() function

In [66]:
```
import numpy as np
```

In [67]:
```
random.rand(3)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[67], line 1
----> 1 random.rand(3)

NameError: name 'random' is not defined
```

In [68]:
```
np.random.rand(2)
```

Out[68]:
```
array([0.09613636, 0.03600593])
```

In [69]:
```
np.random.rand(3)
```

Out[69]:
```
array([0.54053763, 0.50492772, 0.91214216])
```

In [70]:
```
np.random.rand(3,2)
```

Out[70]:
```
array([[0.14052693, 0.10604065],
       [0.63040054, 0.18326406],
       [0.89704888, 0.80194866]])
```

In [71]:
```
np.random.randint(5)
```

Out[71]:
```
4
```

In [72]:
```
np.random.randint(2,10)
```

Out[72]:
```
3
```

In [73]:
```
np.random.randint(2,10,4)
```

Out[73]:
```
array([6, 7, 4, 7])
```

In [74]:
```
np.random.randint(-30,20,10)
```

Out[74]:
```
array([  1,   4,  18,  -6,  12,  18, -28, -13, -13, -16])
```

In [75]:
```
np.random.randint(10,20,(10,10))
```

Out[75]:
```
array([[10, 10, 19, 12, 11, 15, 15, 13, 11, 10],
       [19, 17, 14, 12, 10, 19, 19, 17, 19, 17],
       [10, 11, 10, 12, 14, 16, 19, 18, 14, 19],
       [16, 18, 17, 16, 15, 13, 18, 10, 13, 12],
       [19, 13, 17, 16, 18, 10, 14, 17, 14, 11],
       [10, 18, 10, 17, 10, 16, 15, 12, 12, 14],
       [15, 18, 16, 12, 13, 13, 10, 12, 14, 19],
       [19, 10, 11, 11, 13, 14, 14, 19, 14, 17],
       [18, 10, 13, 11, 15, 19, 11, 12, 16, 12],
       [17, 11, 11, 19, 12, 17, 10, 16, 15, 14]])
```

In [76]:
```
m =np.random.randint(10,20,(10,10))
m
```

Out[76]:
```
array([[19, 14, 18, 11, 13, 14, 18, 16, 13, 14],
       [19, 16, 19, 19, 16, 18, 14, 18, 11, 10],
       [19, 13, 10, 16, 11, 10, 16, 15, 14, 11],
       [17, 14, 17, 18, 13, 11, 10, 15, 13, 12],
       [16, 18, 18, 19, 18, 17, 10, 18, 14, 12],
       [13, 16, 18, 15, 19, 11, 18, 14, 13, 11],
       [17, 18, 16, 19, 19, 11, 13, 18, 12, 18],
       [10, 12, 15, 11, 17, 17, 14, 13, 14, 14],
       [13, 18, 17, 11, 14, 14, 19, 13, 17, 16],
       [11, 15, 15, 12, 19, 14, 17, 10, 16, 10]])
```

Reshape() function

In [77]:
```
arr
```

Out[77]:
```
array([0, 1, 2, 3, 4, 5])
```

In [78]:
```
arr.reshape(2,3)
```

Out[78]:
```
array([[0, 1, 2],
       [3, 4, 5]])
```

In [81]:
```
arr.reshape(6,1)
```

Out[81]:
```
array([[0],
       [1],
       [2],
       [3],
       [4],
       [5]])
```

In [82]:
```
arr.reshape(1,6)
```

Out[82]:
```
array([[0, 1, 2, 3, 4, 5]])
```

sclicing in Matrix

In [83]:
```
m
```

Out[83]:
```
array([[19, 14, 18, 11, 13, 14, 18, 16, 13, 14],
       [19, 16, 19, 19, 16, 18, 14, 18, 11, 10],
       [19, 13, 10, 16, 11, 10, 16, 15, 14, 11],
       [17, 14, 17, 18, 13, 11, 10, 15, 13, 12],
       [16, 18, 18, 19, 18, 17, 10, 18, 14, 12],
       [13, 16, 18, 15, 19, 11, 18, 14, 13, 11],
       [17, 18, 16, 19, 19, 11, 13, 18, 12, 18],
       [10, 12, 15, 11, 17, 17, 14, 13, 14, 14],
       [13, 18, 17, 11, 14, 14, 19, 13, 17, 16],
       [11, 15, 15, 12, 19, 14, 17, 10, 16, 10]])
```

In [92]:
```
b=np.random.randint(10,20,(5,5))      #here it is the matrix using the method  b
b
```

Out[92]:
```
array([[19, 16, 14, 18, 12],
       [18, 13, 11, 18, 19],
       [12, 16, 14, 12, 13],
       [13, 18, 13, 19, 17],
       [14, 13, 16, 15, 14]])
```

In [93]:
```
b[:]
```

Out[93]:
```
array([[19, 16, 14, 18, 12],
       [18, 13, 11, 18, 19],
       [12, 16, 14, 12, 13],
       [13, 18, 13, 19, 17],
       [14, 13, 16, 15, 14]])
```

In [94]:
```
b[-1:]
```

Out[94]:
```
array([[14, 13, 16, 15, 14]])
```

In [95]:
```
b[:-1]   # it is delete the last row (n-1)
```

Out[95]:
```
array([[19, 16, 14, 18, 12],
       [18, 13, 11, 18, 19],
       [12, 16, 14, 12, 13],
       [13, 18, 13, 19, 17]])
```

In [96]:
```
b[1,3]  # (,) is used sepcific col number
```

Out[96]:
```
18
```

In [98]:
```
b[2,-1]
```

Out[98]:
```
13
```

NUMPY OPERATIONS

In [99]:
```
arr
```

Out[99]:
```
array([0, 1, 2, 3, 4, 5])
```

In [101…
```
arr.max()
```

Out[101…
```
5
```

In [102…
```
arr.min()
```

Out[102…
```
0
```

In [14]:
```
from numpy import *
a=array([1,2,3,4,7])
a
```

Out[14]:
```
array([1, 2, 3, 4, 7])
```

In [17]:
```
import numpy as np
```

In [18]:
```
mat=np.arange(0,100).reshape(10,10)
mat
```

Out[18]:
```
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, 24, 25, 26, 27, 28, 29],
       [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
       [40, 41, 42, 43, 44, 45, 46, 47, 48, 49],
       [50, 51, 52, 53, 54, 55, 56, 57, 58, 59],
       [60, 61, 62, 63, 64, 65, 66, 67, 68, 69],
       [70, 71, 72, 73, 74, 75, 76, 77, 78, 79],
       [80, 81, 82, 83, 84, 85, 86, 87, 88, 89],
       [90, 91, 92, 93, 94, 95, 96, 97, 98, 99]])
```

In [19]:
```
row=4
col=6
```

In [20]:
```
col
```

Out[20]:
```
6
```

In [21]:
```
mat[row,col]
```

Out[21]:
```
46
```

In [22]:
```
mat[1]    # it wll print the first row
```

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

In [23]:
```
mat[1,4]
```

Out[23]:
```
14
```

In [24]:
```
mat[:,col]    # here is the col printing single column
```

Out[24]:
```
array([ 6, 16, 26, 36, 46, 56, 66, 76, 86, 96])
```

In [25]:
```
mat[:,3]  # this is col
```

Out[25]:
```
array([ 3, 13, 23, 33, 43, 53, 63, 73, 83, 93])
```

In [26]:
```
mat[3] # this is row
```

Out[26]:
```
array([30, 31, 32, 33, 34, 35, 36, 37, 38, 39])
```

In [27]:
```
mat[::-1]   # it is used for revesre the matrix
```

Out[27]:
```
array([[90, 91, 92, 93, 94, 95, 96, 97, 98, 99],
       [80, 81, 82, 83, 84, 85, 86, 87, 88, 89],
       [70, 71, 72, 73, 74, 75, 76, 77, 78, 79],
       [60, 61, 62, 63, 64, 65, 66, 67, 68, 69],
       [50, 51, 52, 53, 54, 55, 56, 57, 58, 59],
       [40, 41, 42, 43, 44, 45, 46, 47, 48, 49],
       [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
       [20, 21, 22, 23, 24, 25, 26, 27, 28, 29],
       [10, 11, 12, 13, 14, 15, 16, 17, 18, 19],
       [ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9]])
```

In [28]:
```
mat[::-2]
```

Out[28]:
```
array([[90, 91, 92, 93, 94, 95, 96, 97, 98, 99],
       [70, 71, 72, 73, 74, 75, 76, 77, 78, 79],
       [50, 51, 52, 53, 54, 55, 56, 57, 58, 59],
       [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
       [10, 11, 12, 13, 14, 15, 16, 17, 18, 19]])
```

In [29]:
```
mat
```

Out[29]:
```
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, 24, 25, 26, 27, 28, 29],
       [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
       [40, 41, 42, 43, 44, 45, 46, 47, 48, 49],
       [50, 51, 52, 53, 54, 55, 56, 57, 58, 59],
       [60, 61, 62, 63, 64, 65, 66, 67, 68, 69],
       [70, 71, 72, 73, 74, 75, 76, 77, 78, 79],
       [80, 81, 82, 83, 84, 85, 86, 87, 88, 89],
       [90, 91, 92, 93, 94, 95, 96, 97, 98, 99]])
```

In [30]:
```
mat[2:6,2:4]
```

Out[30]:
```
array([[22, 23],
       [32, 33],
       [42, 43],
       [52, 53]])
```

In [31]:
```
mat[1:2,2:4]     # this is called sub-matrix
```

Out[31]:
```
array([[12, 13]])
```

In [32]:
```
mat
```

Out[32]:
```
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, 24, 25, 26, 27, 28, 29],
       [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
       [40, 41, 42, 43, 44, 45, 46, 47, 48, 49],
       [50, 51, 52, 53, 54, 55, 56, 57, 58, 59],
       [60, 61, 62, 63, 64, 65, 66, 67, 68, 69],
       [70, 71, 72, 73, 74, 75, 76, 77, 78, 79],
       [80, 81, 82, 83, 84, 85, 86, 87, 88, 89],
       [90, 91, 92, 93, 94, 95, 96, 97, 98, 99]])
```

In [33]:
```
mat[3:5,2:4]
```

Out[33]:
```
array([[32, 33],
       [42, 43]])
```

MASKING

In [34]:
```
mat
```

Out[34]:
```
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, 24, 25, 26, 27, 28, 29],
       [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
       [40, 41, 42, 43, 44, 45, 46, 47, 48, 49],
       [50, 51, 52, 53, 54, 55, 56, 57, 58, 59],
       [60, 61, 62, 63, 64, 65, 66, 67, 68, 69],
       [70, 71, 72, 73, 74, 75, 76, 77, 78, 79],
       [80, 81, 82, 83, 84, 85, 86, 87, 88, 89],
       [90, 91, 92, 93, 94, 95, 96, 97, 98, 99]])
```

In [35]:
```
mat>50
```

Out[35]:
```
array([[False, False, False, False, False, False, False, False, False,
        False],
       [False, False, False, False, False, False, False, False, False,
        False],
       [False, False, False, False, False, False, False, False, False,
        False],
       [False, False, False, False, False, False, False, False, False,
        False],
       [False, False, False, False, False, False, False, False, False,
        False],
       [False,  True,  True,  True,  True,  True,  True,  True,  True,
         True],
       [ True,  True,  True,  True,  True,  True,  True,  True,  True,
         True],
       [ True,  True,  True,  True,  True,  True,  True,  True,  True,
         True],
       [ True,  True,  True,  True,  True,  True,  True,  True,  True,
         True],
       [ True,  True,  True,  True,  True,  True,  True,  True,  True,
         True]])
```

In [36]:
```
mat[mat>=50]
```

Out[36]:
```
array([50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66,
       67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83,
       84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99])
```

In [37]:
```
mat[mat<50]    # here the
```

Out[37]:
```
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, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
       34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49])
```

In [38]:
```
mat[mat!=50]    #in this all the elements print except (50)
```

Out[38]:
```
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, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
       34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 51,
       52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68,
       69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85,
       86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99])
```

In [39]:
```
id(mat)
```

Out[39]:
```
2636942990224
```

In [40]:
```
mat
```

Out[40]:
```
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, 24, 25, 26, 27, 28, 29],
       [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
       [40, 41, 42, 43, 44, 45, 46, 47, 48, 49],
       [50, 51, 52, 53, 54, 55, 56, 57, 58, 59],
       [60, 61, 62, 63, 64, 65, 66, 67, 68, 69],
       [70, 71, 72, 73, 74, 75, 76, 77, 78, 79],
       [80, 81, 82, 83, 84, 85, 86, 87, 88, 89],
       [90, 91, 92, 93, 94, 95, 96, 97, 98, 99]])
```

In [42]:
```
mat[mat==50]
```

Out[42]:
```
array([50])
```

In [43]:
```
a1= mat[mat<50]
```

In [44]:
```
a1
```

Out[44]:
```
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, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
       34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49])
```

In [ ]: