Numpy Crash Course

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

In [2]:
```
np.__version__
```

Out[2]:
```
'1.26.4'
```

In [3]:
```
import sys
sys.version
```

Out[3]:
```
'3.12.7 | packaged by Anaconda, Inc. | (main, Oct  4 2024, 13:17:27) [MSC v.1
929 64 bit (AMD64)]'
```

Creating Arrays

In [4]:
```
my_list = [0,1,2,3,4,5]
my_list
```

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

In [5]:
```
type(my_list)
```

Out[5]:
```
list
```

Below code we are converting list to array

In [6]:
```
#!pip install numpy
```

In [7]:
```
arr = np.array(my_list)
```

In [8]:
```
arr
```

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

In [10]:
```
type(arr)
```

Out[10]:
```
numpy.ndarray
```

In [9]:
```
type(my_list)
```

Out[9]:
```
list
```

we learn important function

In [10]:
```
np.arange(15)
```

Out[10]:
```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14])
```

In [11]:
```
np.arange(3.0)
```

Out[11]:
```
array([0., 1., 2.])
```

In [12]:
```
np.arange(10)
```

Out[12]:
```
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

In [13]:
```
np.arange(0, 5)
```

Out[13]:
```
array([0, 1, 2, 3, 4])
```

In [14]:
```
np.arange(10,20)
```

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

In [17]:
```
np.arange(20,10) # 1st arg < 2nd arg
```

Out[17]:
```
array([], dtype=int64)
```

In [15]:
```
np.arange(-20,10)
```

Out[15]:
```
array([-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,   8,   9])
```

In [16]:
```
np.arange(-16,10)
```

Out[16]:
```
array([-16, -15, -14, -13, -12, -11, -10,  -9,  -8,  -7,  -6,  -5,  -4,
        -3,  -2,  -1,   0,   1,   2,   3,   4,   5,   6,   7,   8,   9])
```

In [17]:
```
np.arange(-20,-10)
```

Out[17]:
```
array([-20, -19, -18, -17, -16, -15, -14, -13, -12, -11])
```

In [26]:
```
np.arange(30,20)  # 1st arg always be < then 2nd arg
```

Out[26]:
```
array([], dtype=int64)
```

In [18]:
```
ar = np.arange(-30,20)
ar
```

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

In [19]:
```
np.arange(10,10)
```

Out[19]:
```
array([], dtype=int32)
```

In [25]:
```
np.arange()
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[25], line 1
----> 1 np.arange()

TypeError: arange() requires stop to be specified.
```

In [20]:
```
np.arange(10,30,5)  # 10- starting from 30- end point 5 - step count
```

Out[20]:
```
array([10, 15, 20, 25])
```

In [21]:
```
np.arange(0,10,3)
```

Out[21]:
```
array([0, 3, 6, 9])
```

In [28]:
```
np.arange(10,30,5,8)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[28], line 1
----> 1 np.arange(10,30,5,8)

TypeError: Cannot interpret '8' as a data type
```

In [22]:
```
np.zeros(3) # parameter  tunning
```

Out[22]:
```
array([0., 0., 0.])
```

In [23]:
```
np.zeros(5, dtype=int) # hyperparameter tunning
```

Out[23]:
```
array([0, 0, 0, 0, 0])
```

In [24]:
```
np.zeros((2,2), dtype=int)
```

Out[24]:
```
array([[0, 0],
       [0, 0]])
```

In [25]:
```
zero = np.zeros([2,2])
print(zero)
print(type(zero))
```

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

In [26]:
```
zero = np.zeros([2,2])
print(zero)

print('####')

print(type(zero))
```

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

In [27]:
```
np.zeros((2,10))
```

Out[27]:
```
array([[0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]])
```

In [28]:
```
np.zeros((2,2))
```

Out[28]:
```
array([[0., 0.],
       [0., 0.]])
```

In [29]:
```
np.zeros((3,3))
```

Out[29]:
```
array([[0., 0., 0.],
       [0., 0., 0.],
       [0., 0., 0.]])
```

In [30]:
```
np.zeros((10,30))
```

Out[30]:
```
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., 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.],
       [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., 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.],
       [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., 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.],
       [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.]])
```

In [31]:
```
np.zeros((5,10),  dtype=int) # bydefaul 1large -- will give row & 2nd arg - col
```

Out[31]:
```
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, 0, 0, 0, 0, 0]])
```

In [32]:
```
n = (6,7)
n1 = (6,8)
print(np.zeros(n1)) # parameter tunning
```

```
[[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. 0. 0. 0.]]
```

In [33]:
```
print(np.zeros(n,dtype=int)) ## hypyerparameter tunning
```

```
[[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]]
```

In [34]:
```
n
```

Out[34]:
```
(6, 7)
```

In [35]:
```
n1
```

Out[35]:
```
(6, 8)
```

In [36]:
```
print(np.zeros(n1))
```

```
[[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. 0. 0. 0.]]
```

In [37]:
```
np.ones(3)
```

Out[37]:
```
array([1., 1., 1.])
```

In [38]:
```
np.ones(4, dtype=int)
```

Out[38]:
```
array([1, 1, 1, 1])
```

In [39]:
```
np.ones(4)
```

Out[39]:
```
array([1., 1., 1., 1.])
```

In [40]:
```
n
```

Out[40]:
```
(6, 7)
```

In [41]:
```
np.ones(n)
```

Out[41]:
```
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.]])
```

In [42]:
```
np.ones((5,4),dtype=int) # by default 5- rows & 4 - columns
```

Out[42]:
```
array([[1, 1, 1, 1],
       [1, 1, 1, 1],
       [1, 1, 1, 1],
       [1, 1, 1, 1],
       [1, 1, 1, 1]])
```

In [43]:
```
np.
```

```
  Cell In[43], line 1
    np.
       ^
SyntaxError: invalid syntax
```

In [44]:
```
np.twos((2,3))
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[44], line 1
----> 1 np.twos((2,3))

File ~\anaconda3\Lib\site-packages\numpy\__init__.py:347, in __getattr__(attr)
    344     "Removed in NumPy 1.25.0"
    345     raise RuntimeError("Tester was removed in NumPy 1.25.")
--> 347 raise AttributeError("module {!r} has no attribute "
    348                      "{!r}".format(__name__, attr))

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

In [46]:
```
np.three(2,3)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[46], line 1
----> 1 np.three(2,3)

File ~\anaconda3\Lib\site-packages\numpy\__init__.py:347, in __getattr__(attr)
    344     "Removed in NumPy 1.25.0"
    345     raise RuntimeError("Tester was removed in NumPy 1.25.")
--> 347 raise AttributeError("module {!r} has no attribute "
    348                      "{!r}".format(__name__, attr))

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

In [47]:
```
np.ones(2)
```

Out[47]:
```
array([1., 1.])
```

In [48]:
```
np.ones((2,4))
```

Out[48]:
```
array([[1., 1., 1., 1.],
       [1., 1., 1., 1.]])
```

In [49]:
```
np.ones((6,10),dtype = int)
```

Out[49]:
```
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]])
```

In [50]:
```
np.twos((2,4))
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[50], line 1
----> 1 np.twos((2,4))

File ~\anaconda3\Lib\site-packages\numpy\__init__.py:347, in __getattr__(attr)
    344     "Removed in NumPy 1.25.0"
    345     raise RuntimeError("Tester was removed in NumPy 1.25.")
--> 347 raise AttributeError("module {!r} has no attribute "
    348                      "{!r}".format(__name__, attr))

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

In [51]:
```
np.
```

```
  Cell In[51], line 1
    np.
       ^

SyntaxError: invalid syntax
```

In [52]:
```
np.three((2,4))
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[52], line 1
----> 1 np.three((2,4))

File ~\anaconda3\Lib\site-packages\numpy\__init__.py:347, in __getattr__(attr)
    344     "Removed in NumPy 1.25.0"
    345     raise RuntimeError("Tester was removed in NumPy 1.25.")
--> 347 raise AttributeError("module {!r} has no attribute "
    348                      "{!r}".format(__name__, attr))

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

In [53]:
```
range(5)
```

Out[53]:
```
range(0, 5)
```

In [54]:
```
r = range(5)
r
```

Out[54]:
```
range(0, 5)
```

In [55]:
```
for i in r:
    print(i)
```

```
0
1
2
3
4
```

In [56]:
```
list(range(5))
```

Out[56]:
```
[0, 1, 2, 3, 4]
```

In [57]:
```
range(1,10)
```

Out[57]:
```
range(1, 10)
```

In [58]:
```
list(range(1,10))
```

Out[58]:
```
[1, 2, 3, 4, 5, 6, 7, 8, 9]
```

In [59]:
```
list(range(1,10,3))
```

Out[59]:
```
[1, 4, 7]
```

In [60]:
```
y = list(range(12))
y
```

Out[60]:
```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
```

In [65]:
```
rand(3,2)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[65], line 1
----> 1 rand(3,2)

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

In [44]:
```
random.rand(3,2)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[44], line 1
----> 1 random.rand(3,2)

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

In [61]:
```
np.random.rand(5)
```

Out[61]:
```
array([0.0276602 , 0.65381707, 0.4016303 , 0.98248194, 0.91504861])
```

In [62]:
```
np.rand(4)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[62], line 1
----> 1 np.rand(4)

File ~\anaconda3\Lib\site-packages\numpy\__init__.py:347, in __getattr__(attr)
    344     "Removed in NumPy 1.25.0"
    345     raise RuntimeError("Tester was removed in NumPy 1.25.")
--> 347 raise AttributeError("module {!r} has no attribute "
    348                      "{!r}".format(__name__, attr))

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

In [63]:
```
np.random.rand(2,4)
```

Out[63]:
```
array([[0.93899382, 0.69390065, 0.01739729, 0.5982447 ],
       [0.1734148 , 0.93536777, 0.00354134, 0.45142417]])
```

In [64]:
```
np.random.randint(2,4)
```

Out[64]:
```
2
```

In [65]:
```
np.random.randint(2,20) # 2nd argument is exlusive
```

Out[65]:
```
3
```

In [66]:
```
np.random.randint(0,1)
```

Out[66]:
```
0
```

In [67]:
```
np.random.randint(10,20,5)
```

Out[67]:
```
array([12, 17, 10, 19, 17])
```

In [68]:
```
np.random.randint(1,6,4)
```

Out[68]:
```
array([3, 1, 1, 4])
```

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

Out[69]:
```
array([0.95122014, 0.65666495, 0.15891715])
```

In [70]:
```
np.random.randint(1)
```

Out[70]:
```
0
```

In [71]:
```
np.random.randint(30,20,10)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[71], line 1
----> 1 np.random.randint(30,20,10)

File numpy\\random\\mtrand.pyx:780, in numpy.random.mtrand.RandomState.randin
t()

File numpy\\random\\_bounded_integers.pyx:2885, in numpy.random._bounded_intege
rs._rand_int32()

ValueError: low >= high
```

In [72]:
```
np.random.randint(-30,20,10)
```

Out[72]:
```
array([ -5,  10, -24,  17, -18,  -6,   5,  -9, -30, -25])
```

In [73]:
```
np.random.randint(20,30,10)
```

Out[73]:
```
array([21, 22, 26, 23, 26, 29, 22, 23, 25, 24])
```

In [74]:
```
np.random.randint(5,9)  #GET THE VALUE <=1 & >=5
```

Out[74]:
```
7
```

In [75]:
```
np.random.randint(10,21,3)
```

Out[75]:
```
array([14, 15, 13])
```

In [76]:
```
np.random.randint(1,12,10)
```

Out[76]:
```
array([ 6,  4,  6, 10,  4,  2,  4, 10, 10, 11])
```

In [77]:
```
np.random.randint(10,40,(10,10)) #generate the element 10 -30 with 4*4 matrix
```

Out[77]:
```
array([[24, 30, 13, 34, 39, 29, 37, 35, 29, 28],
       [31, 11, 27, 10, 18, 17, 38, 37, 15, 13],
       [29, 14, 14, 16, 24, 11, 36, 19, 15, 36],
       [21, 36, 13, 13, 25, 23, 33, 17, 21, 24],
       [18, 21, 33, 13, 34, 22, 17, 38, 30, 34],
       [12, 22, 16, 32, 39, 23, 37, 19, 12, 11],
       [29, 21, 39, 26, 39, 28, 21, 36, 39, 11],
       [22, 33, 17, 26, 13, 16, 10, 28, 17, 12],
       [11, 15, 30, 19, 10, 18, 38, 18, 30, 12],
       [35, 23, 26, 35, 37, 27, 32, 25, 38, 29]])
```

In [78]:
```
np.random.randint(1,100,(12,12)) #generate the element 10 -30 with 4*4 matrix
```

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

In [79]:
```
np.arange(1,13).reshape(3,4)
```

Out[79]:
```
array([[ 1,  2,  3,  4],
       [ 5,  6,  7,  8],
       [ 9, 10, 11, 12]])
```

In [80]:
```
np.arange(1,13).reshape(12, 1)
```

Out[80]:
```
array([[ 1],
       [ 2],
       [ 3],
       [ 4],
       [ 5],
       [ 6],
       [ 7],
       [ 8],
       [ 9],
       [10],
       [11],
       [12]])
```

In [81]:
```
b = np.random.randint(10,20,(5,4))
b
```

Out[81]:
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [82]:
```
type(b)
```

Out[82]:
```
numpy.ndarray
```

In [83]:
```
b
```

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

In [84]:
```
b[:]
```

Out[84]:
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [85]:
```
b[1:3]
```

Out[85]:
```
array([[15, 11, 18, 12],
       [16, 10, 13, 11]])
```

In [86]:
```
b
```

Out[86]:
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [87]:
```
b[1,2]
```

Out[87]:
```
18
```

In [88]:
```
b
```

Out[88]:
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [89]:
```
b[1,3]
```

Out[89]:
```
12
```

In [90]:
```
b[1,-1]
```

Out[90]:
```
12
```

In [91]:
```
b
```

Out[91]:
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [92]:
```
b[2:3]
```

Out[92]:
```
array([[16, 10, 13, 11]])
```

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

Out[93]:
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [94]:
```
b[0:-2]
```

Out[94]:
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11]])
```

In [95]:
```
b
```

Out[95]:
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [96]:
```
b[0,2]
```

Out[96]:
```
11
```

In [97]:
```
b
```

Out[97]:
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [98]:
```
b[-5,-3]
```

Out[98]:
```
19
```

In [99]:
```
b
```

Out[99]:
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [100…
```
b[-4,2]
```

Out[100…
```
18
```

In [101…
```
np.random.randint(10,20,(4,4))
```

Out[101…
```
array([[11, 13, 11, 15],
       [11, 17, 11, 11],
       [13, 15, 14, 15],
       [10, 16, 11, 19]])
```

In [102…
```
b
```

Out[102…
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [103…
```
b[-4,-2]
```

Out[103…
```
18
```

In [104…
```
b
```

Out[104…
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

In [105…
```
b[-4:2]
```

Out[105…
```
array([[15, 11, 18, 12]])
```

In [106…
```
b[:]
```

Out[106…
```
array([[12, 19, 11, 18],
       [15, 11, 18, 12],
       [16, 10, 13, 11],
       [18, 17, 10, 10],
       [19, 14, 15, 11]])
```

Operations

In [107…
```
a = np.random.randint(10,20,10)
a
```

Out[107…
```
array([14, 18, 14, 11, 10, 17, 18, 19, 11, 13])
```

In [108…
```
id(a)
```

Out[108…
```
1582770183248
```

In [109…
```
arr
```

Out[109…
```
array([0, 1, 2, 3, 4, 5])
```

In [110…
```
arr2 = np.random.randint(0,100,(10,10))
```

In [111…
```
arr2
```

Out[111…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [37, 59, 99, 27, 36, 60, 68, 84, 64, 60]])
```

In [112…
```
arr
```

Out[112…
```
array([0, 1, 2, 3, 4, 5])
```

In [82]:
```
arr[:]
```

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

In [113…
```
arr
```

Out[113…
```
array([0, 1, 2, 3, 4, 5])
```

In [114…
```
arr[:4]
```

Out[114… array([0, 1, 2, 3])

In [115…
```
arr2[:]
```

Out[115…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [37, 59, 99, 27, 36, 60, 68, 84, 64, 60]])
```

In [116…
```
arr2[0:5]
```

Out[116…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29]])
```

In [117…
```
arr2
```

Out[117…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [37, 59, 99, 27, 36, 60, 68, 84, 64, 60]])
```

In [118…
```
arr2[1,4]
```

Out[118… 40

In [119…
```
arr2
```

Out[119…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [37, 59, 99, 27, 36, 60, 68, 84, 64, 60]])
```

In [120…
```
arr2[-5,5]
```

Out[120…
```
38
```

In [121…
```
arr2[-5,-5]
```

Out[121…
```
38
```

In [122…
```
arr2
```

Out[122…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [37, 59, 99, 27, 36, 60, 68, 84, 64, 60]])
```

In [123…
```
arr2[-5,-5]
```

Out[123…
```
38
```

In [124…
```
arr2
```

Out[124…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [37, 59, 99, 27, 36, 60, 68, 84, 64, 60]])
```

In [125…
```
arr2[-1,-2]
```

Out[125…
```
64
```

In [126…
```
arr2
```

Out[126…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [37, 59, 99, 27, 36, 60, 68, 84, 64, 60]])
```

In [127…
```
arr2[::-1]
```

Out[127…
```
array([[37, 59, 99, 27, 36, 60, 68, 84, 64, 60],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [66, 41, 68,  2, 65,  5, 30, 64, 80, 52]])
```

In [128…
```
arr2
```

Out[128…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [37, 59, 99, 27, 36, 60, 68, 84, 64, 60]])
```

In [129…
```
arr2[::-2]
```

Out[129…
```
array([[37, 59, 99, 27, 36, 60, 68, 84, 64, 60],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50]])
```

In [130…
```
arr2
```

Out[130…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [37, 59, 99, 27, 36, 60, 68, 84, 64, 60]])
```

In [131…
```
arr2[::-3]
```

Out[131…
```
array([[37, 59, 99, 27, 36, 60, 68, 84, 64, 60],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [66, 41, 68,  2, 65,  5, 30, 64, 80, 52]])
```

In [132…
```
arr2
```

Out[132…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9],
       [47, 86, 63, 54, 99, 41, 95, 42, 77, 17],
       [44,  3, 74, 31, 70, 49,  1, 20, 79, 13],
       [37, 59, 99, 27, 36, 60, 68, 84, 64, 60]])
```

In [133…
```
arr2[:-3]
```

Out[133…
```
array([[66, 41, 68,  2, 65,  5, 30, 64, 80, 52],
       [71, 88,  2, 89, 40, 97, 60, 30, 35, 50],
       [22, 73, 31,  8, 96, 25, 93, 65,  0, 11],
       [40,  7, 61, 76, 74, 59, 92, 53, 98, 59],
       [96, 28, 18, 34, 44, 26, 51, 51, 10, 29],
       [37,  9, 59, 61, 55, 38, 96,  7, 70, 64],
       [30, 14, 66, 82, 47, 67,  4, 15,  0,  9]])
```

In [134…
```
arr
```

Out[134…
```
array([0, 1, 2, 3, 4, 5])
```

In [135…
```
arr.max()
```

Out[135…
```
5
```

In [136…
```
arr.min()
```

Out[136…
```
0
```

In [137…
```
arr
```

Out[137…
```
array([0, 1, 2, 3, 4, 5])
```

In [138…
```
arr.mean()
```

Out[138…
```
2.5
```

In [139…
```
arr
```

Out[139…
```
array([0, 1, 2, 3, 4, 5])
```

In [140…
```
arr.median()
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[140], line 1
----> 1 arr.median()

AttributeError: 'numpy.ndarray' object has no attribute 'median'
```

In [141…
```
from numpy import *
a = array([1,2,3,4,9])
median(a)
```

Out[141…
```
3.0
```

In [142…
```
arr
```

Out[142…
```
array([0, 1, 2, 3, 4, 5])
```

In [143…
```
arr.reshape(3,2)
```

Out[143…
```
array([[0, 1],
       [2, 3],
       [4, 5]])
```

In [152…
```
arr.reshape(6,1)
```

Out[152…
```
array([[0],
       [1],
       [2],
       [3],
       [4],
       [5]])
```

In [153…
```
arr.reshape(1,6)
```

Out[153…
```
array([[0, 1, 2, 3, 4, 5]])
```

In [154…
```
arr
```

Out[154… array([0, 1, 2, 3, 4, 5])

In [155…
```
arr.reshape(2,4)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[155], line 1
----> 1 arr.reshape(2,4)

ValueError: cannot reshape array of size 6 into shape (2,4)
```

In [156…
```
arr
```

Out[156… array([0, 1, 2, 3, 4, 5])

In [157…
```
arr.reshape(2,3,order='C')
```

Out[157…
```
array([[0, 1, 2],
       [3, 4, 5]])
```

In [158…
```
arr.reshape(2,3,order='F') # print element with fortran
```

Out[158…
```
array([[0, 2, 4],
       [1, 3, 5]])
```

In [159…
```
arr.reshape(2,3,order='A') # A almost give you c type output
```

Out[159…
```
array([[0, 1, 2],
       [3, 4, 5]])
```

In [160…
```
arr
```

Out[160… array([0, 1, 2, 3, 4, 5])

In [161…
```
arr.reshape(2,3)
```

Out[161…
```
array([[0, 1, 2],
       [3, 4, 5]])
```

In [162…
```
arr.reshape(1,4)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[162], line 1
----> 1 arr.reshape(1,4)

ValueError: cannot reshape array of size 6 into shape (1,4)
```

In [163…
```
arr.reshape(1,6)
```

Out[163… array([[0, 1, 2, 3, 4, 5]])

In [164…
```
arr.reshape(6,1)
```

Out[164…
```
array([[0],
       [1],
       [2],
       [3],
       [4],
       [5]])
```

In [165…
```
arr.reshape(2,6)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[165], line 1
----> 1 arr.reshape(2,6)

ValueError: cannot reshape array of size 6 into shape (2,6)
```

In [166…
```
arr.reshape(3,3)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[166], line 1
----> 1 arr.reshape(3,3)

ValueError: cannot reshape array of size 6 into shape (3,3)
```

In [167…
```
arr
```

Out[167…
```
array([0, 1, 2, 3, 4, 5])
```

In [168…
```
arr.reshape(3,2)
```

Out[168…
```
array([[0, 1],
       [2, 3],
       [4, 5]])
```

Indexing

In [42]:
```
mat = np.arange(0,100).reshape(10,10)
```

In [43]:
```
mat
```

Out[43]:
```
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 [44]:
```
row = 4
col = 5
```

In [45]:
```
col
```

Out[45]:
```
5
```

In [46]:
```
row
```

Out[46]:
```
4
```

In [47]:
```
mat
```

Out[47]:
```
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 [48]:
```
mat[row,col]
```

Out[48]:
```
np.int64(45)
```

In [176…
```
mat[4,5]
```

Out[176…
```
np.int64(45)
```

In [177…
```
mat
```

Out[177…
```
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 [178…
```
mat[:]
```

Out[178…
```
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 [179…
```
col = 6
```

In [180…
```
mat
```

Out[180…
```
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 [181…
```
mat[6] # befault it represent to rows
```

Out[181…
```
array([60, 61, 62, 63, 64, 65, 66, 67, 68, 69])
```

In [182…
```
mat
```

Out[182…
```
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 [183…
```
# With Slices
mat[:,col]
```

Out[183…
```
array([ 6, 16, 26, 36, 46, 56, 66, 76, 86, 96])
```

In [184…
```
mat
```

Out[184…
```
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 [243…
```
mat[row,:]
```

Out[243…
```
array([40, 41, 42, 43, 44, 45, 46, 47, 48, 49])
```

In [244…
```
mat
```

Out[244…
```
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 [ ]:
```
mat[:,8]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[:col]
```

In [ ]:
```
mat[:6]
```

In [ ]:
```
row
```

In [ ]:
```
mat
```

In [ ]:
```
mat[:row]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[row:]
```

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

In [ ]:
```
mat[:,8]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[:,-1]
```

In [ ]:
```
mat
```

In [ ]:
```
row
```

In [ ]:
```
col
```

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

In [ ]:
```
mat
```

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

In [ ]:
```
mat
```

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

In [ ]:
```
mat
```

In [ ]:
```
mat[3:-3]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[0]
```

In [ ]:
```
mat[6]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[6:]
```

In [ ]:
```
mat[:6]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[5:7]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[0:10]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[0:10:3]
```

In [ ]:
```
mat[0:10]
```

In [ ]:
```
mat[0:10:3]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[4:]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[:4]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[::-1]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[::-3]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[::-3]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[::-5]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[2:6]
```

In [49]:
```
mat
```

Out[49]:
```
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 [246…
```
mat[2:6,2:4] # 1:5 --> only row part /// 1:3 -- it indicates only column parts
```

Out[246…
```
array([[22, 23],
       [32, 33],
       [42, 43],
       [52, 53]])
```

In [ ]:
```
mat
```

In [ ]:
```
mat[0,1]
```

In [ ]:
```
mat[1,6]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[1:6]
```

In [ ]:
```
mat[1:]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[:6]
```

In [ ]:
```
mat[0:1]
```

In [ ]:
```
mat
```

In [ ]:
```
mat[3:5]
```

In [ ]:
```
mat[3,5]
```

In [50]:
```
mat
```

Out[50]:
```
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 [51]:
```
mat[1:2,2:4]
```

Out[51]:
```
array([[12, 13]])
```

In [247…
```
mat
```

Out[247…
```
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 [248…
```
mat[2:3,2:3]
```

Out[248…
```
array([[22]])
```

In [249…
```
mat
```

Out[249…
```
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 [268…
```
mat[2:4,3:5]
```

Out[268…
```
array([[23, 24],
       [33, 34]])
```

In [250…
```
mat[3:5,2:4]
```

Out[250…
```
array([[32, 33],
       [42, 43]])
```

In [ ]:
```
mat
```

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

Masking

In [251…
```
mat # we also called as filter
```

Out[251…
```
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 [252…
```
id(mat)
```

Out[252…
```
1888731501776
```

In [253…
```
mat
```

Out[253…
```
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 [261…
```
mat[mat<50]
```

Out[261…
```
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 [ ]:
```
mat[mat<=50]
```

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

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

In [ ]:
```
mat
```

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

In [ ]:
```
mat
```

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

In [ ]:
```
mat
```

In [ ]:
```
a2 = mat[mat>50]
a2
```

In [ ]:
```
a3 = mat[mat<=50]
a3
```

In [ ]:
```
a4 = mat[mat==50]
a4
```

python program to generate otp

In [144…
```
import random

def generate_otp(length=4):
    """Generate a numeric OTP of a specified length."""
    digits = '012345'
    otp = ''.join(random.choice(digits) for _ in range(length))
    return otp

# Example usage
otp_length = 4  # You can change this to any length you prefer
otp = generate_otp(otp_length)
print(f"Your OTP is: {otp}")
```

```
Your OTP is: 3332
```

In [145…
```
def wish():
    print('good even')
wish()

def wish():
    print('good even')
wish()

def wish():
    print('good even')
wish()
```

```
good even
good even
good even
```

In [146…
```
def wish():
    print('good even')
wish()

wish()

wish()
```

```
good even
good even
good even
```

In [147…
```
list1=['a','b','g',1,5]
print(list1.pop)
```

```
<built-in method pop of list object at 0x00000170FD2EF080>
```

In [148…
```
x = [1, 2, 3]
y = x.copy()
x.append(4)
print(x)
```

```
[1, 2, 3, 4]
```

In [ ]: