

```
Out[415]:  
{1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

```
In [417]:
```

```
id(d1)
```

```
Out[417]:
```

```
2266874389632
```

```
In [173]:
```

```
id(d2)
```

```
Out[173]:
```

```
2293499501312
```

```
In [419]:
```

```
d1
```

```
Out[419]:
```

```
{1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

```
In [421]:
```

```
d1.pop(2)
```

```
Out[421]:
```

```
'two'
```

```
In [427]:
```

```
d2
```

```
-----  
NameError                                 Traceback (most recent call last)  
Cell In[427], line 1  
----> 1 d2
```

```
NameError: name 'd2' is not defined
```

Day-13(14-08-2024)

NUMPY -- MATRIX - COLLECTION OF DATA STRUCTURED DATA STRUCTURE - COLLECTION OF DATA TYPES
DataBase -- collection of tables TABLE -- COLLECTION OF ROWS & COLUMNS

images--array--numpy text--array--numpy

packages-- collection of modules modules-- collection of functions Function --> inbuild function-- print() len() id() user defient fun if developer want to read the data which is in table format then we need to call numpy library

import numpy as np

**parameter tunning -- laptop (10 version) - parameter tunning hyperparameter tunning -- upgrade to w 11 version
-- hyperparameter tunning model tunning**

machine learning, ai -- hyperparameter tunning llm model - finetune

Numpy crash course

```
In [432]:
```

```
import numpy as np
```

```
In [434]:
```

```
np.__version__
```

```
Out[434]:
```

```
'1.26.4'
```

Creating Arrays

```
In [442]:
```

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

```
Out[442]:
```

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

```
In [444]:
```

```
type(my_list)
```

```
Out[444]:
```

```
list
```

```
In [450]:
```

```
arr = np.array(my_list)
```

```
In [452]:
```

```
arr
```

```
Out[452]:
```

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

```
In [454]:
```

```
type(arr)
```

```
Out[454]:
```

```
numpy.ndarray
```

```
In [456]:
```

```
type(my_list)
```

```
Out[456]:
```

```
list
```

```
In [458]:
```

```
np.arange(15)
```

```
Out[458]:
```

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

```
In [460]:
```

```
np.arange(3.0)
```

```
Out[460]:
```

```
array([0., 1., 2.])
```

```
In [462]:
```

```
np.arange(10)
```

```
In [462]:
```

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

```
In [464]:
```

```
np.arange(0,5)
```

```
Out[464]:
```

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

```
In [466]:
```

```
np.arange(10,20)
```

```
Out[466]:
```

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

```
In [468]:
```

```
np.arange(20,10)#1st arg<2nd arg
```

```
Out[468]:
```

```
array([], dtype=int32)
```

```
In [470]:
```

```
np.arange(-20,10)
```

```
Out[470]:
```

```
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 [472]:
```

```
np.
```

```
Cell In[472], line 1
```

```
np.^
```

```
SyntaxError: invalid syntax
```

```
In [474]:
```

```
np.arange(-16,10)
```

```
Out[474]:
```

```
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 [476]:
```

```
np.arange(-20,-10)
```

```
Out[476]:
```

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

```
In [478]:
```

```
np.arange(30,20)#1st arg alwaysbe<then 2nd arg
```

```
Out[478]:
```

```
array([], dtype=int32)
```

```
In [480]:
```

```
ar = np.arange(-30,20)
ar
```

Out[480]:

```
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 [482]:

```
np.arange(10,10)
```

Out[482]:

```
array([], dtype=int32)
```

In [484]:

```
np.arange(10,30,5) # 10-starting from 30- end point -step count
```

Out[484]:

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

In [486]:

```
np.zeros(5)
```

Out[486]:

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

In [488]:

```
np.zeros(5,dtype=int) # hyperparameter tuning
```

Out[488]:

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

In [490]:

```
np.zeros([10,10])
```

Out[490]:

```
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.]])
```

In [492]:

```
np.zeros((2,2),dtype=int)
```

Out[492]:

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

In [494]:

```
zero =np.zeros([2,2])
type(zero)
```

Out[494]:

```
numpy.ndarray
```

```
In [496]:
```

```
np.zeros((2,10))
```

```
Out[496]:
```

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

```
In [498]:
```

```
np.zeros((5,10)) #bydefaul large --- will give row & 2nd arg --coloums
```

```
Out[498]:
```

```
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 [500]:
```

```
n=(6,7)  
n1=(6,8)  
print(np.zeros(n)) # parameter tuning g
```

```
[[0. 0. 0. 0. 0. 0. 0.]  
 [0. 0. 0. 0. 0. 0. 0.]  
 [0. 0. 0. 0. 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 [502]:
```

```
print(np.zeros(n1,dtype=int)) #
```

```
[[0 0 0 0 0 0 0]  
 [0 0 0 0 0 0 0]  
 [0 0 0 0 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 [504]:
```

```
np.ones(4,dtype=int)
```

```
Out[504]:
```

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

```
In [506]:
```

```
np.ones(4)
```

```
Out[506]:
```

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

```
In [508]:
```

```
n
```

```
Out[508]:
```

```
(6, 7)
```

```
In [510]:
```

```
np.ones(n)
```

Out[510]:

```
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.]])
```

In [512]:

```
np.ones((5,4),dtype=int) #4- rows & 5 columns
```

Out[512]:

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

In [514]:

```
np.twos((2,4))
```

```
-----  
AttributeError                                     Traceback (most recent call last)  
Cell In[514], line 1  
----> 1 np.twos((2,4))  
  
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                 "{}!".format(__name__, attr))
```

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

In [516]:

```
np.three((2,4,6))
```

```
-----  
AttributeError                                     Traceback (most recent call last)  
Cell In[516], line 1  
----> 1 np.three((2,4,6))  
  
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                 "{}!".format(__name__, attr))
```

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

In [522]:

```
from numpy import*  
arange(3)
```

Out[522]:

```
array([0, 1, 2])
```

In [524]:

```
range(12)  
list(range(12))
```

```
Cell In[524], line 2  
list(range(12))
```

```
SyntaxError: incomplete input
```

```
In [526]:
```

```
y = list(range(12))  
y
```

```
Out[526]:
```

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

```
In [528]:
```

```
np.
```

```
Cell In[528], line 1  
  np.  
  ^
```

```
SyntaxError: invalid syntax
```

```
In [530]:
```

```
from numpy import *  
zeros(5)
```

```
Out[530]:
```

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

```
In [534]:
```

```
np.threes((3,2))
```

```
-----  
AttributeError                                     Traceback (most recent call last)  
Cell In[534], line 1  
----> 1 np.threes((3,2))  
  
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 'threes'
```

```
In [536]:
```

```
range(5)
```

```
Out[536]:
```

```
range(0, 5)
```

```
In [538]:
```

```
r=range(5)  
r
```

```
Out[538]:
```

```
range(0, 5)
```

```
In [540]:
```

```
for i in r:  
    print(i)
```

```
0  
1  
2  
3
```

In [542]:

```
list(range(5))
```

Out[542]:

```
[0, 1, 2, 3, 4]
```

In [544]:

```
range(1,10)
```

Out[544]:

```
range(1, 10)
```

In [546]:

```
list(range(1,10))
```

Out[546]:

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

In [548]:

```
list(range(1,10,3))
```

Out[548]:

```
[1, 4, 7]
```

In [550]:

```
y=list(range(12))
y
```

Out[550]:

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

In [552]:

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

NameError

Cell In[552], line 1

----> 1 rand(3,2)

Traceback (most recent call last)

NameError: name 'rand' is not defined

In [554]:

```
rand(3,2)
random.rand(3,2)
```

NameError

Cell In[554], line 1

----> 1 rand(3,2)
 2 random.rand(3,2)

Traceback (most recent call last)

NameError: name 'rand' is not defined

In [556]:

```
np.random.rand(10)
```

Out[556]:

```
array([0.72608227, 0.73908534, 0.10368291, 0.28926603, 0.3586236 ,
```

```
0.46410996, 0.13348424, 0.98890823, 0.59697968, 0.30284529])
```

In [558]:

```
np.rand(4)
```

```
-----  
AttributeError                                     Traceback (most recent call last)  
Cell In[558], line 1  
----> 1 np.rand(4)
```

```
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 'rand'
```

In [560]:

```
np.random.rand(2,4)
```

Out[560]:

```
array([[0.1350084 , 0.79636629, 0.52413831, 0.68269853],  
       [0.05848054, 0.79017526, 0.65165434, 0.58049471]])
```

In [562]:

```
np.random.randint(2,4)
```

Out[562]:

```
2
```

In [564]:

```
np.random.rand(2,20) # 2nd argument is exclusive
```

Out[564]:

```
array([[0.57623704, 0.56136678, 0.40017694, 0.70694341, 0.14708934,  
       0.86351461, 0.59264413, 0.89875244, 0.05821362, 0.02396629,  
       0.09397777, 0.98455718, 0.87641329, 0.06401061, 0.12432605,  
       0.21586431, 0.72319373, 0.92526611, 0.82655486, 0.74643232],  
      [0.83714838, 0.01178732, 0.680504 , 0.71480628, 0.49583308,  
       0.72277986, 0.57478965, 0.70999042, 0.15506169, 0.74284506,  
       0.79147917, 0.13239759, 0.14857398, 0.23337175, 0.86283991,  
       0.05136998, 0.16601613, 0.94862893, 0.56500385, 0.32957686]])
```

In [566]:

```
np.random.randint(2,20)
```

Out[566]:

```
18
```

In [568]:

```
np.random.randint(0,2)
```

Out[568]:

```
0
```

In [572]:

```
np.random.randint(1,10,4)
```

Out[572]:

```
array([9, 8, 2, 7])
```

```
In [574]:
```

```
np.random.randint(1,3,5)
```

```
Out[574]:
```

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

```
In [576]:
```

```
np.random.randint(30,20,10)
```

```
-----  
ValueError
```

```
Cell In[576], line 1
```

```
----> 1 np.random.randint(30,20,10)
```

```
File numpy\\random\\mtrand.pyx:780, in numpy.random.mtrand.RandomState.randint()
```

```
File numpy\\random\\_bounded_integers.pyx:1425, in numpy.random._bounded_integers._rand_i nt32()
```

```
ValueError: low >= high
```

```
In [578]:
```

```
np.random.rand(1)
```

```
Out[578]:
```

```
array([0.74129483])
```

```
In [580]:
```

```
np.random.randint(5,9) #GET THE VALUE <=1 & >=5
```

```
Out[580]:
```

```
5
```

```
In [582]:
```

```
np.random.randint(10,21,3)
```

```
Out[582]:
```

```
array([20, 19, 10])
```

```
In [584]:
```

```
np.random.randint(1,12,10)
```

```
Out[584]:
```

```
array([ 6,  8,  5,  3, 11,  1,  3,  8, 11, 10])
```

```
In [586]:
```

```
np.random.randint(10,40,(10,10)) #generre the element 10 -30 with 4*4 mtri
```

```
Out[586]:
```

```
array([[22, 14, 17, 31, 21, 16, 30, 29, 18, 38],
       [27, 17, 20, 14, 33, 25, 33, 33, 38, 12],
       [22, 14, 18, 26, 14, 26, 39, 14, 18, 17],
       [34, 28, 26, 37, 28, 38, 13, 10, 32, 33],
       [13, 10, 26, 25, 10, 16, 12, 37, 22, 12],
       [37, 34, 14, 28, 38, 25, 26, 18, 16, 25],
       [18, 28, 31, 10, 26, 17, 37, 25, 19, 33],
       [18, 36, 36, 36, 13, 22, 18, 33, 18, 22],
       [21, 29, 12, 10, 26, 31, 36, 25, 26, 31],
       [17, 31, 12, 31, 38, 33, 15, 31, 26, 25]])
```

Day-14(16-08-2024)

In []:

```
>>python index begins with 0 but we count from 1  
start index : end index ( end index = n-1)  
numpy (2,3) --< by default 2 rows & 3 column  
  
>>return all the code  
: --rows  
, -- index rows & columns  
  
**  
devin ai -- automate software engineer
```

In [592]:

```
np.random.randint(1,12,10)
```

Out[592]:

```
array([11,  7,  6, 10, 10,  6,  9,  3,  6, 10])
```

In []:

In [598]:

```
b =np.random.randint(10,20,(5,4))  
b
```

Out[598]:

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

In [600]:

```
b[:]
```

Out[600]:

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

In [604]:

```
b[0:2]
```

Out[604]:

```
array([[16, 14, 15, 18],  
       [11, 13, 19, 16]])
```

In [608]:

```
b
```

Out[608]:

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

```
In [610]:
```

```
b[1:3]
```

```
Out[610]:
```

```
array([[11, 13, 19, 16],  
       [10, 12, 13, 15]])
```

```
In [612]:
```

```
b
```

```
Out[612]:
```

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

```
In [614]:
```

```
b[1,3]
```

```
Out[614]:
```

```
16
```

```
In [616]:
```

```
b[2:3]
```

```
Out[616]:
```

```
array([[10, 12, 13, 15]])
```

```
In [618]:
```

```
b[2,3]
```

```
Out[618]:
```

```
15
```

```
In [620]:
```

```
b[0:-1]
```

```
Out[620]:
```

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

```
In [622]:
```

```
b
```

```
Out[622]:
```

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

```
In [624]:
```

```
b[0,2]
```

```
Out[624]:
```

15

In [626]:

```
b[-5,-3]
```

Out[626]:

```
14
```

In [628]:

```
np.random.randint(10,20,(4,4,))
```

Out[628]:

```
array([[15, 16, 19, 13],  
       [18, 11, 16, 14],  
       [10, 19, 10, 16],  
       [11, 14, 10, 13]])
```

In [630]:

```
b[4,2]
```

Out[630]:

```
19
```

In [632]:

```
b[-4:2]
```

Out[632]:

```
array([[11, 13, 19, 16]])
```

In [634]:

```
b[:]
```

Out[634]:

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

Operations

In []:

```
reshape()  
order c ( c type printing)  
order f (fortran based)  
order a (arbitrary based)
```

In [636]:

```
a =np.random.randint(10,20,5)  
a
```

Out[636]:

```
array([11, 11, 11, 18, 15])
```

In [638]:

```
arr
```

None

```
Out[638]:
```

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

```
In [640]:
```

```
arr2 = np.random.randint(0,100,(10,10))
```

```
In [642]:
```

```
arr2
```

```
Out[642]:
```

```
array([[54, 80, 97, 64, 75, 55, 33, 72, 42, 64],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [6, 1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84],  
       [22, 32, 69, 19, 62, 13, 23, 6, 77, 64],  
       [95, 32, 7, 62, 74, 6, 42, 39, 50, 72],  
       [15, 52, 99, 62, 63, 23, 99, 2, 15, 49]])
```

```
In [644]:
```

```
arr
```

```
Out[644]:
```

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

```
In [646]:
```

```
arr[:]
```

```
Out[646]:
```

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

```
In [648]:
```

```
arr[:4]
```

```
Out[648]:
```

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

```
In [196]:
```

```
arr2[0:5]
```

```
Out[196]:
```

```
array([[27, 22, 15, 11, 32, 55, 35, 59, 71, 64],  
       [91, 56, 91, 4, 38, 42, 31, 6, 6, 42],  
       [58, 88, 78, 21, 88, 6, 73, 74, 52, 98],  
       [42, 36, 48, 92, 66, 3, 34, 82, 50, 38],  
       [50, 15, 3, 87, 59, 1, 3, 64, 65, 0]])
```

```
In [650]:
```

```
arr2[:]
```

```
Out[650]:
```

```
array([[54, 80, 97, 64, 75, 55, 33, 72, 42, 64],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [6, 1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84],  
       [22, 32, 69, 19, 62, 13, 23, 6, 77, 64],  
       [95, 32, 7, 62, 74, 6, 42, 39, 50, 72],  
       [15, 52, 99, 62, 63, 23, 99, 2, 15, 49]])
```

```
[44, 34, 99, 19, 62, 15, 23, 6, 11, 50],  
[95, 32, 7, 62, 74, 6, 42, 39, 50, 72],  
[15, 52, 99, 62, 63, 23, 99, 2, 15, 49]])
```

In [652]:

```
arr2[0:5]
```

Out[652]:

```
array([[54, 80, 97, 64, 75, 55, 33, 72, 42, 64],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [ 6,  1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74]])
```

In [664]:

```
arr2[1,5]
```

Out[664]:

```
79
```

In [666]:

```
arr2
```

Out[666]:

```
array([[54, 80, 97, 64, 75, 55, 33, 72, 42, 64],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [ 6,  1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84],  
       [22, 32, 69, 19, 62, 13, 23, 6, 77, 64],  
       [95, 32, 7, 62, 74, 6, 42, 39, 50, 72],  
       [15, 52, 99, 62, 63, 23, 99, 2, 15, 49]])
```

In [668]:

```
arr2[-5,5]
```

Out[668]:

```
26
```

In [670]:

```
arr2[-5,-5]
```

Out[670]:

```
26
```

In [672]:

```
arr2
```

Out[672]:

```
array([[54, 80, 97, 64, 75, 55, 33, 72, 42, 64],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [ 6,  1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84],  
       [22, 32, 69, 19, 62, 13, 23, 6, 77, 64],  
       [95, 32, 7, 62, 74, 6, 42, 39, 50, 72],  
       [15, 52, 99, 62, 63, 23, 99, 2, 15, 49]])
```

```
In [674]:
```

```
arr2[-1,-2]
```

```
Out[674]:
```

```
15
```

```
In [676]:
```

```
arr2
```

```
Out[676]:
```

```
array([[54, 80, 97, 64, 75, 55, 33, 72, 42, 64],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [ 6,  1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84],  
       [22, 32, 69, 19, 62, 13, 23, 6, 77, 64],  
       [95, 32, 7, 62, 74, 6, 42, 39, 50, 72],  
       [15, 52, 99, 62, 63, 23, 99, 2, 15, 49]])
```

```
In [656]:
```

```
arr2[::-1]
```

```
Out[656]:
```

```
array([[15, 52, 99, 62, 63, 23, 99, 2, 15, 49],  
       [95, 32, 7, 62, 74, 6, 42, 39, 50, 72],  
       [22, 32, 69, 19, 62, 13, 23, 6, 77, 64],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [ 6,  1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [54, 80, 97, 64, 75, 55, 33, 72, 42, 64]])
```

```
In [658]:
```

```
arr2
```

```
Out[658]:
```

```
array([[54, 80, 97, 64, 75, 55, 33, 72, 42, 64],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [ 6,  1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84],  
       [22, 32, 69, 19, 62, 13, 23, 6, 77, 64],  
       [95, 32, 7, 62, 74, 6, 42, 39, 50, 72],  
       [15, 52, 99, 62, 63, 23, 99, 2, 15, 49]])
```

```
In [660]:
```

```
arr2[::-2]
```

```
Out[660]:
```

```
array([[15, 52, 99, 62, 63, 23, 99, 2, 15, 49],  
       [22, 32, 69, 19, 62, 13, 23, 6, 77, 64],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46]])
```

```
In [662]:
```

```
arr2
```

Out[662]:

```
array([[54, 80, 97, 64, 75, 55, 33, 72, 42, 64],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [ 6,  1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84],  
       [22, 32, 69, 19, 62, 13, 23, 6, 77, 64],  
       [95, 32, 7, 62, 74, 6, 42, 39, 50, 72],  
       [15, 52, 99, 62, 63, 23, 99, 2, 15, 49]])
```

In [678]:

```
arr2[::-3]
```

Out[678]:

```
array([[15, 52, 99, 62, 63, 23, 99, 2, 15, 49],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [54, 80, 97, 64, 75, 55, 33, 72, 42, 64]])
```

In [680]:

```
arr2[::-1]
```

Out[680]:

```
array([[15, 52, 99, 62, 63, 23, 99, 2, 15, 49],  
       [95, 32, 7, 62, 74, 6, 42, 39, 50, 72],  
       [22, 32, 69, 19, 62, 13, 23, 6, 77, 64],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [ 6,  1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [54, 80, 97, 64, 75, 55, 33, 72, 42, 64]])
```

In [682]:

```
arr2[:-3]
```

Out[682]:

```
array([[54, 80, 97, 64, 75, 55, 33, 72, 42, 64],  
       [40, 33, 77, 28, 28, 79, 74, 44, 85, 46],  
       [ 6,  1, 51, 84, 45, 97, 15, 44, 35, 76],  
       [50, 32, 54, 57, 49, 69, 11, 53, 66, 74],  
       [97, 78, 38, 44, 52, 11, 31, 34, 53, 74],  
       [68, 39, 70, 47, 86, 26, 68, 74, 50, 4],  
       [18, 41, 51, 56, 86, 77, 96, 1, 7, 84]])
```

In [684]:

```
arr
```

Out[684]:

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

In [686]:

```
arr.max()
```

Out[686]:

5

In [688]:

```
arr.mean()
```

```
Out[688]:
```

```
2.5
```

```
In [690]:
```

```
arr
```

```
Out[690]:
```

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

```
In [692]:
```

```
arr.median()
```

```
-----  
AttributeError
```

```
Cell In[692], line 1
```

```
----> 1 arr.median()
```

```
Traceback (most recent call last)
```

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

```
In [694]:
```

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

```
Out[694]:
```

```
3.0
```

```
In [ ]:
```

```
(Without work on import* can you please find the median,mode)
```

```
In [696]:
```

```
arr
```

```
Out[696]:
```

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

```
In [698]:
```

```
arr.reshape(2,3)
```

```
Out[698]:
```

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

```
In [700]:
```

```
arr.reshape(6,1)
```

```
Out[700]:
```

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

```
In [702]:
```

```
arr.reshape(1,6)
```

```
Out[702]:
```

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

In [704]:

```
arr
```

Out[704]:

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

In [706]:

```
arr.reshape(1,5)
```

ValueError

Cell In[706], line 1
----> 1 arr.reshape(1,5)

Traceback (most recent call last)

```
ValueError: cannot reshape array of size 6 into shape (1,5)
```

In [708]:

```
arr
```

Out[708]:

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

In [710]:

```
arr.reshape(2,3,order='C')
```

Out[710]:

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

In [712]:

```
arr.reshape(3,2,order='F') #print element with fortron
```

Out[712]:

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

In [714]:

```
arr.reshape(2,3,order='A') #
```

Out[714]:

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

In [716]:

```
arr.reshape(3,2,order='K')
```

ValueError

Cell In[716], line 1
----> 1 arr.reshape(3,2,order='K')

Traceback (most recent call last)

```
ValueError: order 'K' is not permitted for reshaping
```

In [718]:

```
arr.reshape(3,2)
```

Out[718]:

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

Indexing

In [720]:

```
mat = np.arange(0,100).reshape(10,10)  
mat
```

Out[720]:

```
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 [728]:

```
row = 4  
col = 5
```

In [724]:

```
col
```

Out[724]:

```
5
```

In [730]:

```
row
```

Out[730]:

```
4
```

In [732]:

```
mat
```

Out[732]:

```
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 [734]:

```
mat[row,col]
```

Out[734]:

```
45
```

In [736]:

```
mat[4,5]
```

```
Out[736]:
```

```
45
```

```
In [738]:
```

```
mat[:]
```

```
Out[738]:
```

```
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 [740]:
```

```
col = 6
```

```
In [742]:
```

```
mat
```

```
Out[742]:
```

```
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 [744]:
```

```
# With Slices  
mat[:,col]
```

```
Out[744]:
```

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

```
In [746]:
```

```
mat[row,:]
```

```
Out[746]:
```

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

```
In [748]:
```

```
mat[:col]
```

```
Out[748]:
```

```
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]])
```

```
In [750]:
```

```
mat[:row]
```

```
Out[750]:
```

```
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]])
```

```
In [752]:
```

```
mat[row:]
```

```
Out[752]:
```

```
array([[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 [754]:
```

```
mat[col:]
```

```
Out[754]:
```

```
array([[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 [756]:
```

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

```
Out[756]:
```

```
array([ 8, 18, 28, 38, 48, 58, 68, 78, 88, 98])
```

```
In [758]:
```

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

```
Out[758]:
```

```
array([ 9, 19, 29, 39, 49, 59, 69, 79, 89, 99])
```

```
In [ ]:
```

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

```
In [760]:
```

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

```
Out[760]:
```

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

```
In [762]:
```

```
mat[1, 4]
```

```
Out[762]:
```

```
14
```

```
In [764]:
```

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

Out[764]:

```
array([[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]])
```

In [766]:

```
mat
```

Out[766]:

```
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 [768]:

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

Out[768]:

```
array([[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]])
```

In [770]:

```
mat[0]
```

Out[770]:

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

In [772]:

```
mat[6]
```

Out[772]:

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

In [774]:

```
mat[6:]
```

Out[774]:

```
array([[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 [776]:

```
mat[:6]
```

Out[776]:

```
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]])
```

In [778]:

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

Out[778]:

```
array([[50, 51, 52, 53, 54, 55, 56, 57, 58, 59],  
       [60, 61, 62, 63, 64, 65, 66, 67, 68, 69]])
```

In [780]:

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

Out[780]:

```
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 [782]:

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

Out[782]:

```
array([[ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9],  
       [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],  
       [60, 61, 62, 63, 64, 65, 66, 67, 68, 69],  
       [90, 91, 92, 93, 94, 95, 96, 97, 98, 99]])
```

In [784]:

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

Out[784]:

```
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 [786]:

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

Out[786]:

```
array([[ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9],  
       [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],  
       [60, 61, 62, 63, 64, 65, 66, 67, 68, 69],  
       [90, 91, 92, 93, 94, 95, 96, 97, 98, 99]])
```

In [788]:

```
mat[4:]
```

Out[788]:

```
array([[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]])
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9],  
[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 [790]:

```
mat[:4]
```

Out[790]:

```
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]])
```

In [792]:

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

Out[792]:

```
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 [794]:

```
mat[::-2]
```

Out[794]:

```
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 [796]:

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

Out[796]:

```
array([[90, 91, 92, 93, 94, 95, 96, 97, 98, 99],  
       [60, 61, 62, 63, 64, 65, 66, 67, 68, 69],  
       [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],  
       [ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9]])
```

In [798]:

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

Out[798]:

```
array([[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]])
```

In [800]:

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

Out[800]:

```
array([[22, 23],  
       [22, 23]]
```

```
[52, 53],  
[52, 53]])
```

In [802]:

```
mat[0,1]
```

Out[802]:

```
1
```

In [804]:

```
mat[1,6]
```

Out[804]:

```
16
```

In [806]:

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

Out[806]:

```
array([[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]])
```

In [808]:

```
mat[1:]
```

Out[808]:

```
array([[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 [810]:

```
mat[:6]
```

Out[810]:

```
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]])
```

In [812]:

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

Out[812]:

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

In [814]:

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

Out[814]:

```
array([[30, 31, 32, 33, 34, 35, 36, 37, 38, 39],  
       [40, 41, 42, 43, 44, 45, 46, 47, 48, 49]])
```

In [816]:

```
mat[3,5]
```

Out[816]:

```
35
```

In [818]:

```
mat[1:2,2:4]
```

Out[818]:

```
array([[12, 13]])
```

In [820]:

```
mat[2:3,2:3]
```

Out[820]:

```
array([[22]])
```

In [822]:

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

Out[822]:

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

In []:

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

Masking

In [824]:

```
mat # we also called as filter
```

Out[824]:

```
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 [826]:

```
id(mat)
```

Out[826]:

```
2266877030832
```

In [828]:

```
mat < 50
```

Out[828]:

```
array([[False, False, False, False, False, False, False, False, False, False],  
       [False, True, False, False, False, False, False, False, False, False],  
       [False, False, True, False, False, False, False, False, False, False],  
       [False, False, False, True, False, False, False, False, False, False],  
       [False, False, False, False, True, False, False, False, False, False],  
       [False, False, False, False, False, True, False, False, False, False],  
       [False, False, False, False, False, False, True, False, False, False],  
       [False, False, False, False, False, False, False, True, False, False],  
       [False, False, False, False, False, False, False, False, True, False],  
       [False, False, False, False, False, False, False, False, False, True]])
```

In [830]:

mat > 50

Out[830]:

In [832]:

```
mat == 50
```

Out[832]:

```
False]])
```

In [834]:

```
mat[mat==50]
```

Out[834]:

```
array([50])
```

In [836]:

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

```
a1
```

Out[836]:

```
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 []:

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

```
a1
```

In [838]:

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

```
a2
```

Out[838]:

```
array([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 [840]:

```
a3 = mat[mat>=50]
```

```
a3
```

Out[840]:

```
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 [842]:

```
a4 = mat[mat==50]
```

```
a4
```

Out[842]:

```
array([50])
```

In [844]:

```
mat>50
```

Out[844]:

```
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, False, False,
       False],
       [False, True, True, True, True, True, True, True,
       True]])
```