Numpy Crash Course

NumPy

NumPy (Numerical Python) is a core package for scientific computing in Python. It provides a high-performance, multidimensional array object, and tools for working with these arrays. With NumPy, you can perform efficient mathematical and logical operations on large datasets, which is essential for data analysis, machine learning, and scientific computations.

Key Concepts in NumPy

1.ndarray (N-dimensional Array)

The primary object in NumPy is the ndarray, a powerful, multidimensional array structure. Unlike Python's built-in list, NumPy arrays are homogeneous (they store elements of the same type) and are optimized for performance. An ndarray is essentially a grid of values indexed by a tuple of non-negative integers.

2.Array Attributes: Arrays in NumPy have several important attributes:

Shape: Defines the dimensions of the array (e.g., for a 2D array, it could be rows and columns).

3.Array Creation: You can create arrays in different ways:

From Lists/Tuples: Directly convert lists or tuples to arrays

4.Array Indexing and Slicing:

Indexing: Access elements using indices. The index is zero-based, meaning it starts at 0.

5.Array Operations:

Element-wise operations: NumPy arrays allow for element-wise mathematical operations, which is much faster than using Python's built-in lists.

Universal Functions (ufuncs): NumPy includes a wide range of mathematical functions (such as np.sqrt(), np.exp(), np.log(), etc.) that operate element-wise on arrays.

6.Broadcasting:

Broadcasting is a powerful feature in NumPy that allows operations on arrays of different shapes and sizes. The smaller array is "broadcast" to the shape of the larger array so they can be operated on together. This avoids the need for explicit loops.

7.Reshaping Arrays:

You can change the shape of an array using the reshape() method. This does not modify the data, only the way it's viewed.

8.Aggregations and Reductions:

NumPy provides a set of functions that allow you to compute summary statistics (e.g., sum, mean, median, etc.) and perform reductions over arrays.

Mean: Calculates the average of elements. python Copy Edit

9.Linear Algebra:

NumPy provides a wide range of linear algebra operations (such as matrix multiplication, determinants, eigenvalues, etc.) via the np.linalg module.

10.Random Module:

NumPy has a random submodule to generate random numbers, shuffle data, and perform random sampling.

11.Masking and Conditional Indexing:

Boolean masking allows you to filter out elements from an array based on conditions.

12.Efficiency:

Memory efficiency: NumPy arrays are more memory-efficient than Python lists.
Performance: NumPy operations are typically faster than Python loops because NumPy operations are implemented in C and are optimized for performance.

In [3]:
```
import numpy as np
```

In [5]:
```
np.__version__
```

Out[5]: '1.26.4'

creating Arrays

In [8]:
```
my_list = [0,1,2,3,4,5]
my_list
```

Out[8]: [0, 1, 2, 3, 4, 5]

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

Out[10]: list

In [12]:
```
arr = np.array(my_list)
```

In [14]:
```
arr
```

Out[14]: array([0, 1, 2, 3, 4, 5])

In [16]:
```
type(arr)
```

Out[16]: numpy.ndarray

In [18]:
```
np.arange(5)
```

Out[18]: array([0, 1, 2, 3, 4])

In [20]:
```
np.arange(3.0)
```

Out[20]: array([0., 1., 2.])

In [22]:
```
np.arange(10)
```

Out[22]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

In [24]:
```
np.arange(0,5)
```

Out[24]: array([0, 1, 2, 3, 4])

In [26]:
```
np.arange(10,20)
```

Out[26]: array([10, 11, 12, 13, 14, 15, 16, 17, 18, 19])

In [28]:
```
np.arange(-20,10)
```

Out[28]:
```
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 [30]:
```
np.arange(-20,-10)
```

Out[30]:
```
array([-20, -19, -18, -17, -16, -15, -14, -13, -12, -11])
```

In [32]:
```
np.arange(30,20)
```

Out[32]:
```
array([], dtype=int32)
```

In [34]:
```
b = np.arange(-20 ,20)
b
```

Out[34]:
```
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,  10,  11,  12,  13,  14,  15,  16,  17,  18,
        19])
```

In [36]:
```
np.arange(10,10)
```

Out[36]:
```
array([], dtype=int32)
```

In [38]:
```
np.arange(-20,20)
```

Out[38]:
```
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,  10,  11,  12,  13,  14,  15,  16,  17,  18,
        19])
```

In [40]:
```
np.arange(10,30,10)
```

Out[40]:
```
array([10, 20])
```

In [42]:
```
np.arange(10,30,3)
```

Out[42]:
```
array([10, 13, 16, 19, 22, 25, 28])
```

In [44]:
```
np.arange(20,30,5)
```

Out[44]:
```
array([20, 25])
```

In [46]:
```
np.arange(10,20,5,5)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[46], line 1
----> 1 np.arange(10,20,5,5)

TypeError: Cannot interpret '5' as a data type
```

In [48]:
```
b1 = np.zeros(10,dtype=int)
b1
```

Out[48]:
```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
```

In [50]:
```
b1 = np.zeros(10,dtype =float)  #hyperparameter tunning
b1
```

Out[50]:
```
array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0.])
```

In [52]:
```
zero = np.zeros((3,7))
zero
```

Out[52]:
```
array([[0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0.]])
```

In [54]:
```
zero = np.ones((10,2))
zero
```

Out[54]:
```
array([[1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.]])
```

In [56]:
```
b1 = np.zeros(10+9,dtype=complex)
b1
```

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

In [58]:
```
zero
```

Out[58]:
```
array([[1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.],
       [1., 1.]])
```

In [60]:
```
np.zeros((12,10))  # bydefaul 1arge -- will give row & 2nd arg - columns
```

Out[60]:
```
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.]])
```

In [62]:
```
np.zeros((4,7))
```

Out[62]:
```
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.]])
```

In [64]:
```
n = (5,9)
n1 = (6,10)
print(np.zeros(n))  # parameter tunnin g
#print(np.zeros(n1,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. 0. 0. 0.]]
```

In [66]:
```
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. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]]
```

In [68]:
```
np.ones(5,dtype=int)
```

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

In [70]:
```
np.ones(9)
```

Out[70]:
```
array([1., 1., 1., 1., 1., 1., 1., 1., 1.])
```

In [367…
```
n
```

Out[367…
```
(5, 9)
```

In [369…
```
n1
```

Out[369…
```
(6, 10)
```

In [371…
```
np.ones(n)
```

Out[371…
```
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.]])
```

In [373…
```
np.ones((5,9),dtype=float) # by default 5- rows & 9 - columns
```

Out[373…
```
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.]])
```

In [72]:
```
np.twos((2,3))
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[72], line 1
----> 1 np.twos((2,3))

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 'twos'
```

In [74]:
```
np.ones((2,4))
```

Out[74]:
```
array([[1., 1., 1., 1.],
       [1., 1., 1., 1.]])
```

In [76]:
```
np.ones((6,10),dtype = int)
```

Out[76]:
```
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]])
```

please add new notebook & compare with np & *

In [79]:
```
from numpy import*
arange(9)
```

Out[79]:
```
array([0, 1, 2, 3, 4, 5, 6, 7, 8])
```

In [81]:
```
range(9)
list(range(9))
```

Out[81]:
```
[0, 1, 2, 3, 4, 5, 6, 7, 8]
```

In [83]:
```
range(9)
```

Out[83]:
```
range(0, 9)
```

In [85]:
```
y = list(range(9))
```

In [87]:
```
y
```

Out[87]:
```
[0, 1, 2, 3, 4, 5, 6, 7, 8]
```

In [89]:
```
from numpy import*
zeros(9)
```

Out[89]:
```
array([0., 0., 0., 0., 0., 0., 0., 0., 0.])
```

In [91]:
```
np.threes((3,9))
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[91], line 1
----> 1 np.threes((3,9))

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 [93]:
```
np.random.rand(9)
```

Out[93]:
```
array([0.78113171, 0.31219597, 0.60378229, 0.26747552, 0.27397891,
       0.4242537 , 0.0406149 , 0.4565745 , 0.2073735 ])
```

In [95]:
```
np.random.rand(2,8)
```

Out[95]:
```
array([[0.69419571, 0.29081374, 0.40709502, 0.61775567, 0.73545027,
        0.9266624 , 0.78001227, 0.97774683],
       [0.15151032, 0.83491688, 0.62450587, 0.04159462, 0.92164915,
        0.75981131, 0.03912169, 0.01199031]])
```

In [99]:
```
np.random.randint(5,9) # 2nd argument is exlusive
```

Out[99]:
```
6
```

In [101…
```
np.random.randint(10,50,5)
```

Out[101…
```
array([29, 41, 10, 35, 23])
```

In [103…
```
np.random.randint(1,3,5)
```

Out[103…
```
array([1, 2, 1, 2, 2])
```

In [105…
```
np.random.randint(10,20,30)
```

Out[105…
```
array([18, 17, 19, 19, 18, 14, 11, 16, 16, 14, 15, 19, 16, 11, 14, 12, 17,
       18, 19, 19, 17, 13, 19, 13, 15, 10, 15, 16, 15, 16])
```

In [107…
```
np.random.randint(90,80,60)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[107], line 1
----> 1 np.random.randint(90,80,60)

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

File numpy\\random\\_bounded_integers.pyx:1425, in numpy.random._bounded_integer
s._rand_int32()

ValueError: low >= high
```

In [109…
```
np.random.randint(10,21,3)
```

Out[109…
```
array([17, 14, 18])
```

In [113…
```
np.random.randint(5,9)  #GET THE VALUE <=1 & >=5
```

Out[113…
```
8
```

In [115…
```
np.random.randint(1,10,(6,10))
```

Out[115…
```
array([[6, 9, 2, 5, 3, 9, 1, 1, 2, 3],
       [5, 4, 5, 9, 2, 4, 3, 8, 7, 5],
       [3, 8, 7, 7, 3, 4, 7, 1, 8, 9],
       [3, 6, 9, 3, 8, 1, 4, 3, 4, 6],
       [6, 9, 2, 7, 5, 2, 4, 3, 9, 5],
       [4, 5, 8, 5, 2, 6, 1, 4, 9, 1]])
```

In [117…
```
np.random.rand(20,30)
```

Out[117…

```
array([[2.59693198e-01, 8.98503736e-01, 4.45456717e-01, 2.48529591e-02,
        6.94596626e-02, 7.28622066e-01, 6.77227241e-02, 3.03241468e-01,
        8.91434547e-01, 3.68757053e-01, 5.74060546e-01, 3.00302250e-01,
        7.04003297e-01, 9.27296549e-01, 9.61794086e-01, 4.23587039e-01,
        4.92002183e-01, 8.50987053e-01, 7.60967376e-01, 9.50339461e-01,
        9.51444790e-01, 9.81612210e-02, 3.55432894e-01, 9.24002532e-01,
        8.18739948e-01, 5.90514969e-01, 6.06013720e-01, 6.56138422e-01,
        6.39188577e-01, 7.57896923e-01],
       [1.63838222e-01, 9.02621011e-01, 6.33685933e-01, 7.55646135e-01,
        8.25156849e-01, 6.51360823e-02, 6.97722834e-01, 8.39148217e-01,
        7.88868080e-02, 5.32881898e-01, 9.56672258e-01, 8.96158358e-02,
        2.15103630e-01, 4.17132413e-01, 5.61307069e-01, 5.75983920e-02,
        5.91714875e-01, 4.20079349e-01, 6.12362911e-02, 9.18778875e-01,
        6.52576381e-01, 7.38582437e-02, 9.48743718e-02, 6.32611543e-01,
        8.98143141e-02, 7.87081721e-02, 5.56930905e-01, 7.88358936e-01,
        7.41763348e-01, 6.24097783e-02],
       [4.80872633e-01, 6.75203393e-01, 7.66428344e-01, 5.28327427e-01,
        5.43840170e-01, 1.27093777e-01, 8.46722568e-01, 2.54031655e-01,
        7.16237113e-01, 3.35326241e-02, 1.36849820e-01, 3.75254387e-01,
        5.72210098e-01, 2.34983179e-01, 4.18640573e-01, 8.37323745e-01,
        5.22503964e-02, 9.23932991e-01, 6.90789905e-01, 2.74624409e-01,
        6.75494522e-01, 9.93941939e-01, 9.70319097e-02, 9.03806353e-02,
        1.54557219e-02, 8.74139004e-01, 1.76313771e-01, 4.81416506e-01,
        1.02203087e-01, 4.95287127e-01],
       [1.09388952e-01, 3.64833542e-02, 3.61215126e-01, 9.63922606e-01,
        8.58631748e-01, 8.52168189e-01, 3.07900217e-01, 7.26005442e-01,
        9.83281149e-01, 6.33669429e-01, 4.92661091e-01, 7.50575693e-01,
        3.70536407e-01, 3.61675926e-01, 3.01630076e-01, 5.97221551e-01,
        4.93112712e-02, 1.75404055e-01, 7.73408420e-01, 4.85882833e-01,
        2.52505582e-01, 4.66654301e-01, 5.05585424e-02, 8.57874112e-01,
        6.28007931e-01, 8.57238655e-01, 8.05368130e-01, 7.45412017e-01,
        4.34386884e-01, 5.25988481e-01],
       [6.26996942e-01, 3.25672191e-01, 7.35248302e-01, 7.94219284e-01,
        2.39999708e-01, 9.92668137e-01, 2.27526217e-01, 4.36389793e-01,
        3.73378752e-01, 4.76886168e-01, 3.28891986e-01, 8.98916256e-01,
        8.74774205e-01, 3.23307813e-01, 4.45112938e-01, 2.46661188e-02,
        9.25758024e-01, 3.01829230e-01, 5.64248011e-01, 6.03130763e-01,
        5.72390958e-01, 6.53419322e-01, 4.99874200e-02, 5.15818072e-01,
        1.68367815e-01, 1.89981228e-01, 8.97546427e-01, 5.87857171e-01,
        4.78912577e-01, 8.18998436e-01],
       [5.66976012e-01, 3.32928869e-01, 2.00753516e-01, 5.64979665e-01,
        2.04218456e-02, 3.76888165e-01, 7.55295445e-01, 4.41146624e-01,
        2.34261394e-01, 2.17367653e-01, 7.87373313e-01, 4.24964090e-01,
        5.65526721e-01, 4.66004731e-01, 6.70429874e-01, 2.47199142e-01,
        5.77015161e-01, 2.47818839e-01, 3.24612084e-01, 1.96797555e-01,
        1.71535315e-01, 4.07831253e-01, 3.20640425e-01, 5.70291413e-01,
        4.07240381e-01, 9.10483088e-01, 9.22817479e-01, 9.16244819e-01,
        8.65454959e-01, 7.62071863e-01],
       [9.63761391e-01, 1.07981238e-02, 7.27674037e-01, 9.60788274e-02,
        7.41170753e-01, 3.92179632e-01, 6.26681225e-02, 4.58216569e-01,
        2.31653894e-02, 6.08221156e-01, 7.24963780e-01, 2.34206084e-02,
        2.86728038e-01, 5.74578100e-01, 9.12570132e-01, 9.07242263e-01,
        5.73339096e-01, 5.18018366e-01, 5.12847447e-01, 1.73403846e-01,
        3.72459637e-01, 1.92367777e-01, 3.88721863e-01, 4.59790754e-01,
        6.22867758e-01, 5.68719237e-02, 6.28098213e-01, 5.26273093e-02,
        1.73509531e-01, 5.74099661e-01],
       [7.76658939e-01, 2.30278918e-01, 6.59444253e-02, 2.31094586e-01,
        3.80651428e-01, 1.90620449e-01, 2.47319704e-01, 7.22859329e-01,
        8.32433753e-01, 2.52602129e-01, 1.20513313e-02, 9.57667120e-02,
        9.49075819e-01, 2.45369783e-01, 9.17015573e-01, 8.44438056e-01,
```

```
 7.51539465e-01, 1.30731254e-01, 7.08688645e-01, 4.13353806e-02,
 5.76668304e-01, 4.97169478e-02, 3.74040598e-01, 3.53709462e-01,
 4.96410679e-01, 8.61762207e-01, 9.81461794e-01, 1.50533130e-02,
 3.40117710e-01, 4.68150228e-01],
[7.95983601e-01, 9.20497380e-01, 6.37943305e-02, 4.14430503e-02,
 9.87203548e-01, 2.29637234e-01, 2.83503448e-01, 4.64910358e-01,
 7.74981120e-01, 5.48834500e-02, 8.77175118e-01, 1.49655219e-01,
 5.56049361e-01, 7.38169149e-02, 5.71069900e-02, 7.47555401e-01,
 1.78867945e-01, 1.97028561e-02, 9.67178964e-01, 6.44470445e-01,
 7.48432767e-01, 4.87575588e-01, 2.47098825e-02, 6.09933273e-01,
 3.11951025e-01, 7.66365340e-01, 5.76333529e-01, 9.22908211e-02,
 1.70705770e-01, 2.97408116e-01],
[4.80588674e-01, 4.21032317e-01, 8.02723727e-01, 4.68584278e-01,
 5.44232306e-01, 1.45019035e-01, 7.61398159e-01, 2.52157657e-01,
 8.56360698e-03, 5.62028814e-01, 9.57468341e-01, 4.14975672e-01,
 1.29789139e-01, 7.46792083e-01, 8.31803048e-01, 9.95475837e-01,
 9.54038783e-01, 8.94590446e-01, 5.15084494e-01, 8.58888183e-01,
 2.62076923e-01, 3.11814532e-01, 5.74620393e-01, 9.86361901e-02,
 9.25091259e-01, 4.72813330e-01, 9.54409605e-01, 2.15837819e-01,
 6.24303088e-01, 5.00689419e-01],
[5.01694356e-01, 9.32631399e-01, 4.44558987e-02, 5.21647394e-01,
 2.73903364e-01, 7.50728402e-01, 2.29396518e-01, 5.27300868e-01,
 1.49621910e-01, 2.59766520e-01, 3.97282707e-02, 2.15821623e-01,
 4.76847687e-03, 3.23148930e-01, 4.25470290e-01, 1.24125919e-01,
 3.84390042e-01, 3.99223316e-01, 5.91318422e-02, 8.77022105e-01,
 5.00046678e-01, 5.30947757e-01, 5.41343110e-01, 9.21183069e-01,
 5.47906995e-01, 1.72525810e-01, 2.16317273e-01, 7.43228192e-01,
 7.78276076e-01, 2.23703792e-01],
[5.56637430e-01, 4.53055842e-01, 3.50565618e-01, 9.57878403e-01,
 1.15643620e-01, 7.45473213e-01, 3.42229295e-01, 8.46953946e-01,
 8.67010721e-01, 4.68232639e-01, 1.91501075e-02, 7.86606899e-01,
 9.44585515e-01, 8.81182747e-01, 6.00733861e-01, 5.90886447e-01,
 8.38935867e-01, 9.58441094e-01, 9.34511099e-02, 6.39785775e-01,
 1.40669918e-01, 2.56235591e-01, 8.66116089e-01, 2.87669904e-01,
 9.31635222e-01, 6.31910252e-01, 9.92370248e-01, 1.62281898e-01,
 8.27004867e-01, 6.23441313e-01],
[7.85830086e-02, 5.24906945e-01, 1.47218806e-02, 2.46413043e-01,
 9.79887283e-01, 8.07217323e-01, 8.01169088e-01, 8.03437587e-01,
 2.77617640e-01, 3.62302032e-01, 1.08646398e-01, 6.14636739e-01,
 1.12419880e-01, 7.14899778e-01, 1.33190260e-01, 5.31463607e-01,
 3.38533306e-01, 1.23071998e-01, 4.20704772e-01, 9.87059531e-01,
 3.28141139e-03, 3.58702112e-01, 4.78759757e-02, 1.97904577e-01,
 9.93205028e-01, 2.72791855e-01, 3.76072370e-01, 6.65593865e-01,
 1.65239678e-01, 5.15587020e-01],
[3.72765718e-01, 6.70527424e-01, 8.59428461e-01, 6.36547714e-01,
 2.28521869e-01, 4.41927146e-01, 4.56377103e-01, 4.23942890e-01,
 4.14556232e-01, 9.05805769e-01, 5.74707866e-01, 9.13044796e-01,
 5.31158378e-01, 6.03735103e-01, 1.08207366e-01, 5.98083182e-01,
 3.01051307e-01, 8.40503833e-01, 4.79673891e-01, 4.12153874e-01,
 9.71520218e-01, 5.72535272e-01, 3.53799448e-01, 8.85829315e-01,
 7.53652669e-01, 5.88084064e-01, 7.97068647e-01, 9.94113720e-01,
 7.25028703e-01, 6.33679469e-01],
[5.74835369e-01, 1.09074308e-01, 1.35359677e-01, 5.23820664e-01,
 6.05418668e-01, 2.94480280e-01, 7.71412761e-01, 5.93678851e-01,
 8.49521103e-01, 3.45545060e-01, 6.88127403e-01, 4.25514880e-01,
 5.22961207e-01, 7.93338604e-01, 8.41940657e-02, 6.02490250e-01,
 5.59117157e-01, 6.73605902e-01, 6.96775576e-01, 9.47569330e-01,
 5.59124866e-01, 7.85178139e-01, 3.62342189e-01, 5.07374071e-01,
 6.57512863e-01, 2.29728417e-01, 8.12277204e-01, 6.52446696e-01,
 7.30860554e-01, 7.20314623e-01],
```

```
[5.79637657e-02, 2.77425661e-01, 4.39001642e-01, 5.30513448e-01,
 1.11011449e-01, 7.97116705e-01, 1.32345827e-01, 6.59298462e-02,
 2.00891310e-01, 8.66638714e-01, 1.44621258e-01, 2.19089861e-03,
 1.88693443e-02, 6.07905877e-01, 7.65038827e-01, 5.24072245e-01,
 2.93542975e-01, 7.71325663e-01, 8.94499856e-01, 2.27079421e-01,
 2.00728957e-01, 6.21245773e-02, 4.63947086e-01, 2.14168413e-01,
 1.67929284e-01, 7.91777335e-01, 9.63979106e-01, 4.05799266e-01,
 7.52017385e-01, 6.99599798e-03],
[7.96064553e-01, 9.95490104e-01, 8.86184969e-01, 4.87471277e-01,
 6.18450520e-01, 7.92116803e-01, 1.94835185e-01, 6.85802747e-02,
 7.30163192e-01, 4.31353674e-01, 1.82120398e-01, 4.53303236e-01,
 4.97803351e-01, 7.78350464e-01, 6.52702262e-01, 5.15615801e-01,
 1.34370784e-02, 9.31751066e-01, 6.95251068e-01, 3.47160090e-01,
 8.88799413e-01, 2.52504166e-01, 5.59346471e-01, 2.59204664e-01,
 5.01141365e-01, 3.67323049e-01, 2.53923126e-01, 9.52887641e-01,
 5.76060897e-01, 4.19996045e-01],
[7.36726187e-01, 1.25134657e-01, 8.62752821e-01, 6.79187665e-01,
 4.51816817e-01, 7.93726917e-01, 3.46821625e-01, 5.79815899e-02,
 2.92603126e-01, 9.81451484e-01, 2.18664889e-01, 7.84592466e-01,
 9.58356150e-01, 7.85958681e-01, 3.26240446e-01, 4.36456200e-01,
 9.17961461e-01, 5.79648935e-01, 1.11958865e-01, 3.94016747e-01,
 8.27396834e-02, 9.27768208e-01, 1.00741451e-01, 4.46076178e-01,
 2.36048638e-01, 1.06064776e-02, 6.44011340e-01, 2.59005308e-01,
 9.34342324e-01, 8.14926897e-01],
[8.27645890e-01, 3.59165805e-01, 7.74938178e-01, 5.94297800e-02,
 5.54268683e-01, 6.71801348e-01, 6.93330612e-01, 6.55512141e-02,
 3.23273050e-01, 8.41393912e-01, 6.33664169e-01, 1.97314275e-01,
 3.27874683e-01, 2.21261751e-01, 5.61538186e-01, 7.03456864e-01,
 9.03940491e-01, 5.92966265e-01, 1.50552034e-01, 8.59769724e-01,
 1.00898878e-01, 1.03630293e-01, 6.71305058e-01, 2.63890153e-01,
 3.60496126e-01, 4.80920866e-01, 6.59095910e-01, 6.70217347e-01,
 1.68875981e-02, 4.69399364e-02],
[6.15664145e-01, 9.93357705e-01, 5.98137549e-01, 3.98357786e-01,
 4.31843630e-01, 2.41848977e-01, 3.32511555e-04, 2.69469405e-02,
 6.30323727e-01, 6.70645366e-01, 7.80170872e-01, 2.33519511e-01,
 7.55018974e-01, 7.68005633e-01, 9.14364371e-02, 9.94428122e-01,
 5.56312471e-01, 3.76394609e-01, 3.83391159e-01, 7.50331465e-01,
 7.79326382e-01, 3.94018028e-02, 1.95726024e-02, 3.59617393e-01,
 9.65283457e-01, 1.93152717e-01, 2.25973516e-02, 1.59267541e-03,
 6.50918902e-01, 5.95305641e-02]])
```

In [119…

```
np.random.rand(9,9)
```

Out[119…
```
array([[0.3519976 , 0.04345955, 0.23739889, 0.33269728, 0.40624925,
        0.97095133, 0.47725833, 0.59637629, 0.73076017],
       [0.90320359, 0.24914435, 0.67432342, 0.20192022, 0.67201608,
        0.13516271, 0.28642754, 0.77295581, 0.53908426],
       [0.2613774 , 0.91376424, 0.80599482, 0.99225142, 0.41482354,
        0.5554168 , 0.61219359, 0.61553877, 0.66972868],
       [0.28689069, 0.3407645 , 0.75145562, 0.04349142, 0.32264622,
        0.10134876, 0.80221172, 0.47833291, 0.35348699],
       [0.35053566, 0.66800726, 0.01077699, 0.17193577, 0.55107729,
        0.19045068, 0.05664063, 0.9133506 , 0.73718949],
       [0.80320806, 0.48714694, 0.57298415, 0.36055712, 0.5527174 ,
        0.02895492, 0.67203206, 0.36494305, 0.87618376],
       [0.78403116, 0.00808515, 0.63526285, 0.97388892, 0.21658915,
        0.92000561, 0.18802338, 0.51093245, 0.67964887],
       [0.45381064, 0.0787703 , 0.64081797, 0.55610506, 0.18025371,
        0.84828187, 0.64303045, 0.24387714, 0.90345783],
       [0.57785801, 0.79897592, 0.94103548, 0.72115736, 0.45173056,
        0.82236595, 0.83617287, 0.75831617, 0.77054661]])
```

In [411…
```
np.random.randint(0,5)
```

Out[411…
```
1
```

In [121…
```
np.random.randint(10,40,(10,10)) #generre the element 10 -30 with 4*4 mtri
```

Out[121…
```
array([[10, 18, 13, 10, 29, 19, 35, 11, 39, 35],
       [17, 32, 11, 20, 34, 29, 32, 39, 29, 39],
       [32, 36, 15, 21, 36, 17, 33, 35, 20, 25],
       [26, 15, 25, 25, 27, 12, 29, 12, 15, 28],
       [11, 32, 20, 37, 12, 31, 21, 22, 39, 29],
       [10, 37, 27, 24, 17, 33, 35, 15, 24, 15],
       [39, 12, 14, 36, 35, 36, 21, 39, 12, 30],
       [12, 29, 21, 15, 21, 27, 24, 29, 38, 14],
       [10, 18, 15, 16, 31, 11, 16, 38, 34, 12],
       [17, 28, 28, 20, 38, 14, 12, 30, 24, 26]])
```

In [123…
```
np.random.randint(9,77)
```

Out[123…
```
73
```

In [125…
```
np.random. randint(0,2)
```

Out[125…
```
0
```

In [127…
```
b = np.random.randint(10,20,(5,4))
b
```

Out[127…
```
array([[12, 17, 12, 16],
       [10, 12, 19, 16],
       [12, 16, 16, 19],
       [10, 10, 16, 15],
       [11, 11, 19, 19]])
```

In [129…
```
b[:]
```

Out[129…
```
array([[12, 17, 12, 16],
       [10, 12, 19, 16],
       [12, 16, 16, 19],
       [10, 10, 16, 15],
       [11, 11, 19, 19]])
```

In [131…
```
b[0:2]
```

Out[131…
```
array([[12, 17, 12, 16],
       [10, 12, 19, 16]])
```

In [133…
```
b
```

Out[133…
```
array([[12, 17, 12, 16],
       [10, 12, 19, 16],
       [12, 16, 16, 19],
       [10, 10, 16, 15],
       [11, 11, 19, 19]])
```

In [135…
```
b[0:-1]
```

Out[135…
```
array([[12, 17, 12, 16],
       [10, 12, 19, 16],
       [12, 16, 16, 19],
       [10, 10, 16, 15]])
```

In [137…
```
np.random.randint(10)
```

Out[137…
```
6
```

In [139…
```
np.random.randint(1,2,3)
```

Out[139…
```
array([1, 1, 1])
```

In [141…
```
np.random.randint(11,15,3)
```

Out[141…
```
array([13, 14, 12])
```

In [143…
```
np.random.randint(9,36,(11,5))
```

Out[143…
```
array([[20, 35, 19, 23, 28],
       [22, 15, 28, 17, 28],
       [29, 29, 19, 27, 19],
       [12, 18, 17, 35, 22],
       [14, 16, 15, 29, 16],
       [19, 14, 16, 21, 30],
       [25, 33, 26, 23, 29],
       [18, 26, 31, 14, 19],
       [34, 23, 17, 22, 19],
       [24, 21, 31, 16, 19],
       [18, 34, 32, 17, 26]])
```

In [145…
```
b = np.random.randint(8,20,(9,5))
b
```

Out[145…
```
array([[14, 19, 14, 17, 10],
       [14,  9, 10, 16,  9],
       [19, 10, 11, 18, 12],
       [19, 15, 12, 19,  8],
       [15, 11, 15, 17, 15],
       [12, 18, 18, 15, 14],
       [17, 16, 10, 17, 16],
       [18,  9, 18,  8, 14],
       [19, 17, 10, 13, 19]])
```

In [147…
```
b[:]
```

Out[147…
```
array([[14, 19, 14, 17, 10],
       [14,  9, 10, 16,  9],
       [19, 10, 11, 18, 12],
       [19, 15, 12, 19,  8],
       [15, 11, 15, 17, 15],
       [12, 18, 18, 15, 14],
       [17, 16, 10, 17, 16],
       [18,  9, 18,  8, 14],
       [19, 17, 10, 13, 19]])
```

In [149…
```
b[2:9]
```

Out[149…
```
array([[19, 10, 11, 18, 12],
       [19, 15, 12, 19,  8],
       [15, 11, 15, 17, 15],
       [12, 18, 18, 15, 14],
       [17, 16, 10, 17, 16],
       [18,  9, 18,  8, 14],
       [19, 17, 10, 13, 19]])
```

In [151…
```
b[0:4]
```

Out[151…
```
array([[14, 19, 14, 17, 10],
       [14,  9, 10, 16,  9],
       [19, 10, 11, 18, 12],
       [19, 15, 12, 19,  8]])
```

In [153…
```
b[4:6]
```

Out[153…
```
array([[15, 11, 15, 17, 15],
       [12, 18, 18, 15, 14]])
```

In [155…
```
b
```

Out[155…
```
array([[14, 19, 14, 17, 10],
       [14,  9, 10, 16,  9],
       [19, 10, 11, 18, 12],
       [19, 15, 12, 19,  8],
       [15, 11, 15, 17, 15],
       [12, 18, 18, 15, 14],
       [17, 16, 10, 17, 16],
       [18,  9, 18,  8, 14],
       [19, 17, 10, 13, 19]])
```

In [157…
```
b[0,4]
```

Out[157…
```
10
```

In [159…
```
b[-2,-4]
```

Out[159…
```
9
```

In [161…
```
b[-4,-5]
```

Out[161…
```
12
```

OPERATIONS

In [166…
```
a = np.random.randint(10,20,5)
a
```

Out[166…
```
array([14, 10, 12, 15, 16])
```

In [168…
```
arr
```

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

In [170…
```
arr2 = np.random.randint(0,100,(10,10))
```

In [172…
```
arr2
```

Out[172…
```
array([[23, 74, 41, 46,  2, 89, 44, 64, 24, 43],
       [85,  0, 20, 45, 50, 83, 72, 27, 66, 54],
       [82, 12, 29, 83, 79, 36,  2, 65, 97, 45],
       [92, 23, 95, 68, 78, 50, 28, 41,  3, 37],
       [27, 49, 17, 26, 22, 66, 94, 33, 57, 89],
       [32,  4, 69, 29, 86, 71, 10, 55, 63, 42],
       [51,  1, 85, 40, 29, 83, 35, 50, 61, 23],
       [17, 85, 99, 22, 26, 39, 11, 42, 33, 90],
       [77, 84, 50,  5, 94, 76, 48, 41, 74, 23],
       [36, 75, 43, 38, 19, 62, 11, 83, 18, 72]])
```

In [174…
```
arr
```

Out[174…
```
array([0, 1, 2, 3, 4, 5])
```

In [176…
```
arr[:]
```

Out[176…
```
array([0, 1, 2, 3, 4, 5])
```

In [178…
```
arr[:4]
```

Out[178…
```
array([0, 1, 2, 3])
```

In [180…
```
arr2[:]
```

Out[180…
```
array([[23, 74, 41, 46,  2, 89, 44, 64, 24, 43],
       [85,  0, 20, 45, 50, 83, 72, 27, 66, 54],
       [82, 12, 29, 83, 79, 36,  2, 65, 97, 45],
       [92, 23, 95, 68, 78, 50, 28, 41,  3, 37],
       [27, 49, 17, 26, 22, 66, 94, 33, 57, 89],
       [32,  4, 69, 29, 86, 71, 10, 55, 63, 42],
       [51,  1, 85, 40, 29, 83, 35, 50, 61, 23],
       [17, 85, 99, 22, 26, 39, 11, 42, 33, 90],
       [77, 84, 50,  5, 94, 76, 48, 41, 74, 23],
       [36, 75, 43, 38, 19, 62, 11, 83, 18, 72]])
```

In [182…
```
arr2[0:5]
```

Out[182…
```
array([[23, 74, 41, 46,  2, 89, 44, 64, 24, 43],
       [85,  0, 20, 45, 50, 83, 72, 27, 66, 54],
       [82, 12, 29, 83, 79, 36,  2, 65, 97, 45],
       [92, 23, 95, 68, 78, 50, 28, 41,  3, 37],
       [27, 49, 17, 26, 22, 66, 94, 33, 57, 89]])
```

In [184…
```
arr2
```

Out[184…
```
array([[23, 74, 41, 46,  2, 89, 44, 64, 24, 43],
       [85,  0, 20, 45, 50, 83, 72, 27, 66, 54],
       [82, 12, 29, 83, 79, 36,  2, 65, 97, 45],
       [92, 23, 95, 68, 78, 50, 28, 41,  3, 37],
       [27, 49, 17, 26, 22, 66, 94, 33, 57, 89],
       [32,  4, 69, 29, 86, 71, 10, 55, 63, 42],
       [51,  1, 85, 40, 29, 83, 35, 50, 61, 23],
       [17, 85, 99, 22, 26, 39, 11, 42, 33, 90],
       [77, 84, 50,  5, 94, 76, 48, 41, 74, 23],
       [36, 75, 43, 38, 19, 62, 11, 83, 18, 72]])
```

In [186…
```
arr2[1,5]
```

Out[186…
```
83
```

In [188…
```
arr2
```

Out[188…
```
array([[23, 74, 41, 46,  2, 89, 44, 64, 24, 43],
       [85,  0, 20, 45, 50, 83, 72, 27, 66, 54],
       [82, 12, 29, 83, 79, 36,  2, 65, 97, 45],
       [92, 23, 95, 68, 78, 50, 28, 41,  3, 37],
       [27, 49, 17, 26, 22, 66, 94, 33, 57, 89],
       [32,  4, 69, 29, 86, 71, 10, 55, 63, 42],
       [51,  1, 85, 40, 29, 83, 35, 50, 61, 23],
       [17, 85, 99, 22, 26, 39, 11, 42, 33, 90],
       [77, 84, 50,  5, 94, 76, 48, 41, 74, 23],
       [36, 75, 43, 38, 19, 62, 11, 83, 18, 72]])
```

In [190…
```
arr2[-5,5]
```

Out[190…
```
71
```

In [192…
```
arr2[-5,-5]
```

Out[192…
```
71
```

In [194…
```
arr2
```

Out[194…
```
array([[23, 74, 41, 46,  2, 89, 44, 64, 24, 43],
       [85,  0, 20, 45, 50, 83, 72, 27, 66, 54],
       [82, 12, 29, 83, 79, 36,  2, 65, 97, 45],
       [92, 23, 95, 68, 78, 50, 28, 41,  3, 37],
       [27, 49, 17, 26, 22, 66, 94, 33, 57, 89],
       [32,  4, 69, 29, 86, 71, 10, 55, 63, 42],
       [51,  1, 85, 40, 29, 83, 35, 50, 61, 23],
       [17, 85, 99, 22, 26, 39, 11, 42, 33, 90],
       [77, 84, 50,  5, 94, 76, 48, 41, 74, 23],
       [36, 75, 43, 38, 19, 62, 11, 83, 18, 72]])
```

In [196…
```
arr2[-5,-5]
```

Out[196…
```
71
```

In [198…
```
arr2
```

Out[198…
```
array([[23, 74, 41, 46,  2, 89, 44, 64, 24, 43],
       [85,  0, 20, 45, 50, 83, 72, 27, 66, 54],
       [82, 12, 29, 83, 79, 36,  2, 65, 97, 45],
       [92, 23, 95, 68, 78, 50, 28, 41,  3, 37],
       [27, 49, 17, 26, 22, 66, 94, 33, 57, 89],
       [32,  4, 69, 29, 86, 71, 10, 55, 63, 42],
       [51,  1, 85, 40, 29, 83, 35, 50, 61, 23],
       [17, 85, 99, 22, 26, 39, 11, 42, 33, 90],
       [77, 84, 50,  5, 94, 76, 48, 41, 74, 23],
       [36, 75, 43, 38, 19, 62, 11, 83, 18, 72]])
```

In [200…
```
arr2[-1,-2]
```

Out[200…
```
18
```

In [202…
```
arr2[::-1]
```

Out[202…
```
array([[36, 75, 43, 38, 19, 62, 11, 83, 18, 72],
       [77, 84, 50,  5, 94, 76, 48, 41, 74, 23],
       [17, 85, 99, 22, 26, 39, 11, 42, 33, 90],
       [51,  1, 85, 40, 29, 83, 35, 50, 61, 23],
       [32,  4, 69, 29, 86, 71, 10, 55, 63, 42],
       [27, 49, 17, 26, 22, 66, 94, 33, 57, 89],
       [92, 23, 95, 68, 78, 50, 28, 41,  3, 37],
       [82, 12, 29, 83, 79, 36,  2, 65, 97, 45],
       [85,  0, 20, 45, 50, 83, 72, 27, 66, 54],
       [23, 74, 41, 46,  2, 89, 44, 64, 24, 43]])
```

In [204…
```
arr2[::-2]
```

Out[204…
```
array([[36, 75, 43, 38, 19, 62, 11, 83, 18, 72],
       [17, 85, 99, 22, 26, 39, 11, 42, 33, 90],
       [32,  4, 69, 29, 86, 71, 10, 55, 63, 42],
       [92, 23, 95, 68, 78, 50, 28, 41,  3, 37],
       [85,  0, 20, 45, 50, 83, 72, 27, 66, 54]])
```

In [206…
```
arr2[::-3]
```

Out[206…
```
array([[36, 75, 43, 38, 19, 62, 11, 83, 18, 72],
       [51,  1, 85, 40, 29, 83, 35, 50, 61, 23],
       [92, 23, 95, 68, 78, 50, 28, 41,  3, 37],
       [23, 74, 41, 46,  2, 89, 44, 64, 24, 43]])
```

In [212…
```
arr
```

Out[212…
```
array([0, 1, 2, 3, 4, 5])
```

In [214…
```
arr.max()
```

Out[214…
```
5
```

In [216…
```
arr.min()
```

Out[216…
```
0
```

In [218…
```
>>> from numpy import *
>>> a = array([1,2,3,4,9])
>>> median(a)
```

Out[218…
```
3.0
```

In [220…
```
arr
```

Out[220…
```
array([0, 1, 2, 3, 4, 5])
```

In [222…
```
arr.reshape(2,3)
```

Out[222…
```
array([[0, 1, 2],
       [3, 4, 5]])
```

In [224…
```
arr.reshape(6,1)
```

Out[224…
```
array([[0],
       [1],
       [2],
       [3],
       [4],
       [5]])
```

In [226…
```
arr.reshape(1,6)
```

Out[226…
```
array([[0, 1, 2, 3, 4, 5]])
```

In [228…
```
arr.reshape(1,6)
```

Out[228…
```
array([[0, 1, 2, 3, 4, 5]])
```

In [230…
```
arr.reshape(3,2,order='C')
```

Out[230…
```
array([[0, 1],
       [2, 3],
       [4, 5]])
```

In [232…
```
arr.reshape(3,2,order='F')
```

Out[232…
```
array([[0, 3],
       [1, 4],
       [2, 5]])
```

In [234…
```
arr.reshape(2,3)
```

Out[234…
```
array([[0, 1, 2],
       [3, 4, 5]])
```

In [236…
```
arr.reshape(1,6)
```

Out[236…
```
array([[0, 1, 2, 3, 4, 5]])
```

In [238…
```
arr.reshape(6,1)
```

Out[238…
```
array([[0],
       [1],
       [2],
       [3],
       [4],
       [5]])
```

In [240…
```
arr.reshape(2,6)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[240], line 1
----> 1 arr.reshape(2,6)

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

In [242…
```
arr.reshape(3,3)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[242], line 1
----> 1 arr.reshape(3,3)

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

In [244…
```
arr.reshape(3,2)
```

Out[244…
```
array([[0, 1],
       [2, 3],
       [4, 5]])
```

Indexing

In [255…
```
mat = np.arange(0,200).reshape(20,10)
```

In [257…
```
mat
```

Out[257…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [259…
```
row = 4
col = 5
```

In [261…
```
col
```

Out[261… 5

In [263…
```
row
```

Out[263… 4

In [265…
```
mat
```

Out[265…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [267…
```
mat[row,col]
```

Out[267… 45

In [269…
```
mat[4,5]
```

Out[269…
```
45
```

In [271…
```
mat
```

Out[271…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [273…
```
mat[:]
```

Out[273…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [279…
```
col = 7
```

In [281…
```
mat
```

Out[281…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [283…
```
# With Slices
mat[:,col]
```

Out[283…
```
array([  7,  17,  27,  37,  47,  57,  67,  77,  87,  97, 107, 117, 127,
       137, 147, 157, 167, 177, 187, 197])
```

In [285…
```
mat
```

Out[285…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [287…
```
mat[row,:]
```

Out[287…
```
array([40, 41, 42, 43, 44, 45, 46, 47, 48, 49])
```

In [289…
```
mat[:,col]
```

Out[289…
```
array([  7,  17,  27,  37,  47,  57,  67,  77,  87,  97, 107, 117, 127,
       137, 147, 157, 167, 177, 187, 197])
```

In [291…
```
mat[:col]
```

Out[291…
```
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]])
```

In [293…
```
mat
```

Out[293…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [297…
```
mat[:row]
```

Out[297…
```
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 [299…
```
mat[row:]
```

Out[299…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [301…
```
mat[:,8]
```

Out[301…
```
array([  8,  18,  28,  38,  48,  58,  68,  78,  88,  98, 108, 118, 128,
       138, 148, 158, 168, 178, 188, 198])
```

In [303…
```
mat[:,-1]
```

Out[303…
```
array([  9,  19,  29,  39,  49,  59,  69,  79,  89,  99, 109, 119, 129,
       139, 149, 159, 169, 179, 189, 199])
```

In [305…
```
mat[:,col]
```

Out[305…
```
array([  7,  17,  27,  37,  47,  57,  67,  77,  87,  97, 107, 117, 127,
       137, 147, 157, 167, 177, 187, 197])
```

In [307…
```
mat[1,4]
```

Out[307…
```
14
```

In [311…
```
mat[3:-5]
```

Out[311…
```
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],
       [ 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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149]])
```

In [313…
```
mat[0]
```

Out[313…
```
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

In [315…
```
mat[6:]
```

Out[315…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [317…
```
mat[0:10]
```

Out[317…
```
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 [319…
```
mat[0:10:3]
```

Out[319…
```
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 [321…
```
mat[0:10]
```

Out[321…
```
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 [323…
```
mat[0:10:3]
```

Out[323…
```
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 [325…
```
mat[4:]
```

Out[325…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [327…
```
mat[:4]
```

Out[327…
```
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 [329…
```
mat[::-1]
```

Out[329…
```
array([[190, 191, 192, 193, 194, 195, 196, 197, 198, 199],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [ 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 [331…
```
mat[::-2]
```

Out[331…
```
array([[190, 191, 192, 193, 194, 195, 196, 197, 198, 199],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [ 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 [333…
```
mat[::-5]
```

Out[333…
```
array([[190, 191, 192, 193, 194, 195, 196, 197, 198, 199],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [ 90,  91,  92,  93,  94,  95,  96,  97,  98,  99],
       [ 40,  41,  42,  43,  44,  45,  46,  47,  48,  49]])
```

In [335…
```
mat[::-5]
```

Out[335…
```
array([[190, 191, 192, 193, 194, 195, 196, 197, 198, 199],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [ 90,  91,  92,  93,  94,  95,  96,  97,  98,  99],
       [ 40,  41,  42,  43,  44,  45,  46,  47,  48,  49]])
```

In [337…
```
mat[2:6]
```

Out[337…
```
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 [339…
```
mat[2:6,2:4] # 1:5 --> only row part /// 1:3 -- it indicates only column parts
```

Out[339…
```
array([[22, 23],
       [32, 33],
       [42, 43],
       [52, 53]])
```

In [341…
```
mat[1:2,2:4]
```

Out[341…
```
array([[12, 13]])
```

In [343…
```
mat[2:3,2:3]
```

Out[343…
```
array([[22]])
```

In [345…
```
mat[3:5,2:4,]
```

Out[345…
```
array([[32, 33],
       [42, 43]])
```

In [347…
```
mat[2:3,4:5]
```

Out[347…
```
array([[24]])
```

Masking

In [350…
```
mat # we also called as filter
```

Out[350…
```
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],
       [100, 101, 102, 103, 104, 105, 106, 107, 108, 109],
       [110, 111, 112, 113, 114, 115, 116, 117, 118, 119],
       [120, 121, 122, 123, 124, 125, 126, 127, 128, 129],
       [130, 131, 132, 133, 134, 135, 136, 137, 138, 139],
       [140, 141, 142, 143, 144, 145, 146, 147, 148, 149],
       [150, 151, 152, 153, 154, 155, 156, 157, 158, 159],
       [160, 161, 162, 163, 164, 165, 166, 167, 168, 169],
       [170, 171, 172, 173, 174, 175, 176, 177, 178, 179],
       [180, 181, 182, 183, 184, 185, 186, 187, 188, 189],
       [190, 191, 192, 193, 194, 195, 196, 197, 198, 199]])
```

In [352…
```
id(mat)
```

Out[352…
```
1688032255056
```

In [354…
```
mat < 50
```

Out[354…
```
array([[ 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,
         True],
       [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, 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],
       [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]])
```

In [356…
```
mat > 50
```

Out[356…

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

In [358…

```
mat == 50
```

Out[358…
```
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],
       [ True, 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, 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, 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]])
```

In [360…
```
mat[mat==50]
```

Out[360…
```
array([50])
```

In [362…
```
a1 = mat[mat<50]
a1
```

Out[362…
```
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 [364…
```
a2 = mat[mat>50]
a2
```

Out[364…
```
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, 100, 101, 102,
       103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115,
       116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128,
       129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141,
       142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154,
       155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167,
       168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180,
       181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193,
       194, 195, 196, 197, 198, 199])
```

In [366…
```
a3 = mat[mat>=50]
a3
```

Out[366…
```
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, 100, 101,
       102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114,
       115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127,
       128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140,
       141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153,
       154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166,
       167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179,
       180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192,
       193, 194, 195, 196, 197, 198, 199])
```

In [368…
```
a4 = mat[mat==50]
a4
```

Out[368…
```
array([50])
```

In [370…
```
a1
```

Out[370…
```
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 [372…
```
a2
```

Out[372…
```
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, 100, 101, 102,
       103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115,
       116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128,
       129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141,
       142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154,
       155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167,
       168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180,
       181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193,
       194, 195, 196, 197, 198, 199])
```

In [ ]: