numpy

```
In [9]:  import numpy as np
```

```
In [11]:  np.__version__
```

```
Out[11]:  '1.26.4'
```

```
In [13]:  import sys
          sys.version
```

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

creating arrays

```
In [15]:  my_list=[0,1,2,3,4,5]
          my_list
```

```
Out[15]:  [0, 1, 2, 3, 4, 5]
```

```
In [17]:  type(my_list)
```

```
Out[17]:  list
```

```
In [19]:  ! pip install numpy
```

```
Requirement already satisfied: numpy in c:\users\91630\anaconda3\lib\site-package
s (1.26.4)
```

```
In [20]:  arr=np.array(my_list)
          arr
```

```
Out[20]:  array([0, 1, 2, 3, 4, 5])
```

```
In [21]:  type(arr)
```

```
Out[21]:  numpy.ndarray
```

```
In [22]:  type(my_list)
```

```
Out[22]:  list
```

12 mar

```
In [28]:  np.arange(15)
```

```
Out[28]:  array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14])
```

```
In [30]:  np.arange(3.0)
```

```
Out[30]:  array([0., 1., 2.])
```

```
In [32]:  np.arange(10)
```

```
Out[32]:  array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

```
In [34]:  np.arange(0,5)
```

```
Out[34]:  array([0, 1, 2, 3, 4])
```

```
In [36]:  np.arange(10,20)
```

```
Out[36]:  array([10, 11, 12, 13, 14, 15, 16, 17, 18, 19])
```

```
In [38]:  np.arange(20,10)
```

```
Out[38]:  array([], dtype=int32)
```

```
In [48]:  np.arange(-20,10)
```

```
Out[48]:  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 [42]:  np.arange(-20,-10)
```

```
Out[42]:  array([-20, -19, -18, -17, -16, -15, -14, -13, -12, -11])
```

```
In [44]:  np.arange(-16,10)
```

```
Out[44]:  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 [50]:  np.arange()
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[50], line 1
----> 1 np.arange()

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

```
In [52]:  np.arange(10,30,5)
```

```
Out[52]:  array([10, 15, 20, 25])
```

```
In [54]:  np.arange(0,10,3)
```

```
Out[54]:  array([0, 3, 6, 9])
```

```
In [56]:  np.arange(10,30,5,8)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[56], line 1
----> 1 np.arange(10,30,5,8)

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

```
In [58]: np.zeros(3)
```

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

```
In [60]: np.zeros(5,dtype=int)
```

```
Out[60]: array([0, 0, 0, 0, 0])
```

```
In [62]: np.zeros(10)
```

```
Out[62]: array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0.])
```

```
In [64]: np.zeros(10,dtype=int)
```

```
Out[64]: array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
```

```
In [66]: np.zeros((2,2),dtype=int)
```

```
Out[66]: array([[0, 0],
                [0, 0]])
```

```
In [68]: np.zeros((2,10))
```

```
Out[68]: array([[0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]])
```

```
In [70]: np.zeros((2,2))
```

```
Out[70]: array([[0., 0.],
                [0., 0.]])
```

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

```
Out[72]: array([[0., 0., 0.],
                [0., 0., 0.],
                [0., 0., 0.]])
```

```
In [74]: np.zeros((10,30))
```

```
Out[74]: array([[0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]])
```

```
In [78]: np.zeros((5,10) ,dtype=int)
```

```
Out[78]: 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 [80]: np.zeros((10,30),dtype=int)
```

```
Out[80]: array([[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                 0, 0, 0, 0, 0, 0, 0, 0],
                [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                 0, 0, 0, 0, 0, 0, 0, 0]])
```

```
In [82]: np.zeros((4,5))
```

```
Out[82]: array([[0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0.],
                [0., 0., 0., 0., 0.]])
```

```
In [84]: n=(6,7)
         n1=(6,8)
         print(np.zeros(n1))

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

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

```
In [86]: np.ones(3)

Out[86]: array([1., 1., 1.])
```

```
In [88]: np.ones((3),dtype=int)

Out[88]: array([1, 1, 1])
```

```
In [90]: np.ones((2,2))

Out[90]: array([[1., 1.],
                [1., 1.]])
```

```
In [92]: np.ones((3,3))

Out[92]: array([[1., 1., 1.],
                [1., 1., 1.],
                [1., 1., 1.]])
```

```
In [94]: np.ones((5,4),dtype=int)

Out[94]: array([[1, 1, 1, 1],
                [1, 1, 1, 1],
                [1, 1, 1, 1],
                [1, 1, 1, 1],
                [1, 1, 1, 1]])
```

```
In [96]: np.ones(n)

Out[96]: array([[1., 1., 1., 1., 1., 1., 1.],
                [1., 1., 1., 1., 1., 1., 1.],
                [1., 1., 1., 1., 1., 1., 1.],
                [1., 1., 1., 1., 1., 1., 1.],
                [1., 1., 1., 1., 1., 1., 1.],
                [1., 1., 1., 1., 1., 1., 1.]])
```

```
In [98]: np.two((2,3))
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[98], line 1
----> 1 np.two((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 'two'
```

In [100…

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

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[100], line 1
----> 1 np.three(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 'three'
```

In [102…

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

Out[102…

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

In [104…

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

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[104], line 1
----> 1 rand(3,2)

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

In [106…

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

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[106], line 1
----> 1 random.rand(2,3)

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

In [116…

```
np.random.rand(5)
```

Out[116…

```
array([0.47561306, 0.07316728, 0.15353268, 0.11369991, 0.20109498])
```

In [120…

```
np.random.rand(3,5)
```

Out[120…

```
array([[0.00410293, 0.60552931, 0.01048813, 0.88657004, 0.96350042],
       [0.60894121, 0.83814465, 0.80028972, 0.73958738, 0.76493569],
       [0.83263292, 0.22953813, 0.31660485, 0.78455843, 0.84951026]])
```

```
In [122…  np.random.rand(2,4)
```

```
Out[122…  array([[0.14027466, 0.16886904, 0.04210728, 0.71490359],
                 [0.86748032, 0.83595919, 0.58572162, 0.73464083]])
```

```
In [124…  np.random.rand(10)
```

```
Out[124…  array([0.18632627, 0.67964224, 0.37813084, 0.64929985, 0.26701721,
                 0.78293104, 0.99877067, 0.90829317, 0.36602054, 0.00923698])
```

```
In [126…  np.random.rand(2,20)
```

```
Out[126…  array([[0.49943555, 0.54245132, 0.37241806, 0.53651865, 0.36633174,
                  0.93707477, 0.51813935, 0.26874922, 0.10197615, 0.21597855,
                  0.52385694, 0.39140903, 0.45892836, 0.75018706, 0.93932778,
                  0.8326053 , 0.31560814, 0.93458946, 0.34105142, 0.09933268],
                 [0.59336364, 0.64288928, 0.5316189 , 0.59329712, 0.23960755,
                  0.16212206, 0.13290707, 0.88899192, 0.70769264, 0.37199758,
                  0.67030052, 0.07867605, 0.74067642, 0.0068822 , 0.45355532,
                  0.37437526, 0.80213338, 0.52454856, 0.10435111, 0.31279168]])
```

```
In [128…  np.random.rand(0,1)
```

```
Out[128…  array([], shape=(0, 1), dtype=float64)
```

```
In [130…  np.random.rand(10,20,5)
```

```
Out[130…  array([[[0.93755081, 0.92469859, 0.19220522, 0.87188353, 0.16805864],
                 [0.01982623, 0.16794926, 0.50322043, 0.75667417, 0.17105448],
                 [0.88407614, 0.70690404, 0.98871705, 0.95530002, 0.73811919],
                 [0.17837322, 0.95632187, 0.24277278, 0.73645594, 0.57440517],
                 [0.18593436, 0.00298354, 0.00669692, 0.65218074, 0.84192232],
                 [0.50670081, 0.39033068, 0.78164087, 0.63845588, 0.42778985],
                 [0.91141514, 0.15941084, 0.82612124, 0.62072681, 0.67687225],
                 [0.17802474, 0.699076  , 0.90163489, 0.82235391, 0.08555749],
                 [0.90406487, 0.1212242 , 0.28850656, 0.54347788, 0.27278537],
                 [0.35495248, 0.51493237, 0.81987362, 0.84576123, 0.763771  ],
                 [0.60584532, 0.62720225, 0.78919675, 0.70093793, 0.06161004],
                 [0.60336683, 0.31017504, 0.12967189, 0.07056048, 0.41788533],
                 [0.48103594, 0.3048832 , 0.47973889, 0.5197277 , 0.70652886],
                 [0.7265386 , 0.93380715, 0.52700584, 0.75042171, 0.80499917],
                 [0.22705027, 0.1825071 , 0.59728857, 0.56473615, 0.9310778 ],
                 [0.40973201, 0.00873237, 0.35224868, 0.60249956, 0.70484944],
                 [0.59873305, 0.38337794, 0.27965416, 0.15492128, 0.80830986],
                 [0.87808597, 0.28138207, 0.48327919, 0.5799018 , 0.97616523],
                 [0.37624923, 0.8790185 , 0.99307322, 0.06840873, 0.58731879],
                 [0.81473463, 0.26740041, 0.24481075, 0.7789023 , 0.98856779]],

                [[0.1761917 , 0.84316624, 0.12963358, 0.69980171, 0.58409799],
                 [0.27066184, 0.39181859, 0.63318978, 0.08202163, 0.48913748],
                 [0.97340536, 0.25378377, 0.63951577, 0.6394919 , 0.74356242],
                 [0.10668668, 0.11770995, 0.42443258, 0.62144234, 0.37787983],
                 [0.66237721, 0.06610827, 0.9972745 , 0.5148036 , 0.20930199],
                 [0.74423688, 0.73618934, 0.4858072 , 0.85969955, 0.27439003],
                 [0.43875522, 0.95051963, 0.2244558 , 0.40070592, 0.67354922],
                 [0.82616724, 0.64621539, 0.89937085, 0.6840513 , 0.93758272],
                 [0.17348478, 0.1427799 , 0.12117706, 0.10161116, 0.97706169],
                 [0.36432923, 0.30075588, 0.38126501, 0.13815038, 0.47140867],
                 [0.17811925, 0.24654424, 0.72743034, 0.16248446, 0.74992963],
                 [0.75309022, 0.07782246, 0.65520771, 0.17668457, 0.31432567],
                 [0.74912288, 0.206754  , 0.12142571, 0.47276831, 0.07851087],
                 [0.79007593, 0.58899847, 0.37271342, 0.44505188, 0.93194693],
                 [0.48580527, 0.49228141, 0.43113268, 0.54899232, 0.82639924],
                 [0.61085576, 0.9663926 , 0.71553081, 0.5730595 , 0.43991982],
                 [0.49857865, 0.02358445, 0.41667478, 0.63012897, 0.34401664],
                 [0.37264978, 0.61114405, 0.68299202, 0.54293146, 0.81716813],
                 [0.37374132, 0.01536992, 0.014801  , 0.2929615 , 0.11997457],
                 [0.45612348, 0.07800096, 0.19704094, 0.32727967, 0.05639642]],

                [[0.97247946, 0.30327257, 0.61239552, 0.89188103, 0.64158753],
                 [0.14420642, 0.2231552 , 0.62960056, 0.62918386, 0.42557851],
                 [0.5537418 , 0.46357664, 0.43014394, 0.51685347, 0.89028551],
                 [0.69964258, 0.15082094, 0.65380572, 0.47874497, 0.14140416],
                 [0.78368039, 0.44721175, 0.86464216, 0.87535802, 0.92375361],
                 [0.76688762, 0.81607478, 0.25534725, 0.76308636, 0.77173896],
                 [0.40164913, 0.65046744, 0.10171619, 0.39688617, 0.01050919],
                 [0.83121606, 0.43872152, 0.76905745, 0.58795727, 0.31426902],
                 [0.93043374, 0.17674657, 0.00336209, 0.62848404, 0.49769068],
                 [0.38804748, 0.46298134, 0.83336976, 0.76591157, 0.58984225],
                 [0.01975042, 0.48287681, 0.93598635, 0.77273764, 0.25889941],
                 [0.45015459, 0.73816701, 0.86458106, 0.18392249, 0.6147327 ],
                 [0.10895326, 0.06533652, 0.21041292, 0.08468431, 0.15344463],
                 [0.0170302 , 0.211347  , 0.38628253, 0.83763206, 0.09041976],
                 [0.09806552, 0.4561287 , 0.1386793 , 0.42950473, 0.39964666],
                 [0.46408069, 0.9971073 , 0.17736059, 0.86586199, 0.87568266],
                 [0.08039494, 0.45525071, 0.7836155 , 0.77526079, 0.57522701],
                 [0.96900828, 0.78873091, 0.15984609, 0.09343452, 0.41025113],
 [0.43462484, 0.15971207, 0.43319119, 0.78119622, 0.90988179],
 [0.1757688 , 0.07719074, 0.23603964, 0.46410195, 0.46652417]],

[[0.6248618 , 0.0589254 , 0.26313672, 0.95442104, 0.83486534],
 [0.22447278, 0.89130818, 0.21172906, 0.91962447, 0.31688246],
 [0.17679947, 0.12017954, 0.74058957, 0.47305921, 0.86352321],
 [0.21767093, 0.82855876, 0.71957996, 0.81548978, 0.86887514],
 [0.04596154, 0.65689877, 0.3061745 , 0.95914883, 0.26683124],
 [0.65966662, 0.44109922, 0.58302356, 0.98729456, 0.29897593],
 [0.78436267, 0.0553994 , 0.26023532, 0.91104314, 0.01621738],
 [0.16997061, 0.12426662, 0.09457711, 0.9570029 , 0.9855391 ],
 [0.7994653 , 0.63826809, 0.67244105, 0.67272055, 0.82568123],
 [0.15847093, 0.26158939, 0.53997559, 0.58246889, 0.45916024],
 [0.20574121, 0.22615407, 0.42962568, 0.74291401, 0.35230484],
 [0.17987605, 0.08662492, 0.46632076, 0.05380037, 0.89774341],
 [0.50793855, 0.26292061, 0.55137333, 0.37616975, 0.95971648],
 [0.75961419, 0.4424101 , 0.31064562, 0.95791965, 0.06288714],
 [0.9948801 , 0.61906153, 0.59140088, 0.28481217, 0.18759485],
 [0.04990506, 0.11590315, 0.08342282, 0.76777903, 0.42749714],
 [0.75372925, 0.99248846, 0.03927131, 0.02496899, 0.05156411],
 [0.24487068, 0.9010084 , 0.29849965, 0.99817937, 0.24907127],
 [0.58519634, 0.38372333, 0.07268763, 0.3080278 , 0.82166451],
 [0.55976355, 0.19785512, 0.87273525, 0.25914812, 0.09499147]],

[[0.36203622, 0.4175289 , 0.93980482, 0.79016442, 0.1872227 ],
 [0.93774299, 0.5362487 , 0.59287125, 0.65424476, 0.94684138],
 [0.96130582, 0.89677261, 0.66013225, 0.17518851, 0.12967714],
 [0.6590932 , 0.0076999 , 0.53839623, 0.34701784, 0.21898949],
 [0.22482172, 0.83411656, 0.87512116, 0.30515915, 0.45980726],
 [0.11915843, 0.60146065, 0.32754521, 0.17727922, 0.42229268],
 [0.68148813, 0.39599705, 0.0801293 , 0.95959117, 0.05769793],
 [0.27784415, 0.15762298, 0.86276169, 0.77735769, 0.9073255 ],
 [0.23378336, 0.4032843 , 0.940581  , 0.4106377 , 0.58335887],
 [0.54921443, 0.05908446, 0.01732796, 0.62374404, 0.07308626],
 [0.8010929 , 0.48412842, 0.30167412, 0.94321205, 0.76173753],
 [0.8577651 , 0.29923419, 0.9659991 , 0.67833165, 0.73064773],
 [0.38449831, 0.00295909, 0.44153383, 0.50834446, 0.98784219],
 [0.56431336, 0.76801716, 0.03897825, 0.78605461, 0.88673133],
 [0.19308313, 0.05294983, 0.48137496, 0.63148289, 0.5826107 ],
 [0.85241953, 0.90845969, 0.82632039, 0.03497535, 0.46421609],
 [0.70612403, 0.37037637, 0.58285944, 0.90192548, 0.21787977],
 [0.61359689, 0.24719364, 0.98037616, 0.26133721, 0.4914487 ],
 [0.88743863, 0.83334422, 0.81647883, 0.42243743, 0.55114517],
 [0.62190283, 0.11542929, 0.84750602, 0.70938398, 0.88784238]],

[[0.83549618, 0.81222618, 0.60109451, 0.41529393, 0.79280008],
 [0.02125666, 0.343593  , 0.50689183, 0.1563112 , 0.59666986],
 [0.11734573, 0.14912748, 0.50610879, 0.14694465, 0.29792575],
 [0.22071385, 0.17999407, 0.39608956, 0.97328809, 0.93114931],
 [0.75113775, 0.04327978, 0.528743  , 0.24788812, 0.6191245 ],
 [0.98545636, 0.71173243, 0.15035395, 0.13838345, 0.89553461],
 [0.49934504, 0.06673847, 0.19501636, 0.1222825 , 0.97777802],
 [0.1730867 , 0.03889436, 0.11671113, 0.73610612, 0.76464058],
 [0.43941567, 0.46163311, 0.2996907 , 0.87019299, 0.47989787],
 [0.96711471, 0.29024014, 0.15942945, 0.53023958, 0.71780621],
 [0.86605011, 0.49398359, 0.78402937, 0.82269807, 0.84979165],
 [0.12384802, 0.23352139, 0.98988096, 0.35674473, 0.13262479],
 [0.82086682, 0.96721975, 0.84529913, 0.88712216, 0.25022261],
 [0.5909798 , 0.23960318, 0.74339657, 0.88121407, 0.40924001],
 [0.50798997, 0.58628054, 0.44604671, 0.23674074, 0.47421924],
 [0.88821338, 0.43527897, 0.56645989, 0.71761749, 0.16749895],
 [0.32296786, 0.25836223, 0.14740451, 0.20935196, 0.95953251],
 [0.36198035, 0.28112774, 0.85436231, 0.62163384, 0.4924219 ],
 [0.33569044, 0.48074841, 0.0507581 , 0.7714956 , 0.14021944],
 [0.54810909, 0.49959319, 0.42203089, 0.65050787, 0.9577089 ]],

[[0.07480064, 0.25440825, 0.16430154, 0.94021736, 0.60583117],
 [0.18757704, 0.84180922, 0.63588383, 0.71988717, 0.97305739],
 [0.05281397, 0.93167347, 0.68116263, 0.18126546, 0.04803782],
 [0.52839316, 0.9712619 , 0.48637444, 0.39385036, 0.60548597],
 [0.51187895, 0.31524313, 0.39432977, 0.95979346, 0.7326715 ],
 [0.35688042, 0.64554688, 0.20620079, 0.04607273, 0.71250131],
 [0.92305638, 0.43665329, 0.82261902, 0.9758326 , 0.00953484],
 [0.43544446, 0.92189548, 0.4015479 , 0.11812455, 0.38571047],
 [0.06440657, 0.66981829, 0.50798123, 0.28210156, 0.18443351],
 [0.48463142, 0.77441522, 0.05443943, 0.42036863, 0.47275017],
 [0.95130817, 0.68493124, 0.99016173, 0.56620395, 0.89417254],
 [0.21085162, 0.13433671, 0.26929789, 0.91748271, 0.95942449],
 [0.45442993, 0.63517036, 0.2929015 , 0.81202926, 0.75003201],
 [0.1989736 , 0.22665893, 0.44062803, 0.77636143, 0.21047567],
 [0.57178105, 0.6942059 , 0.20627485, 0.93276121, 0.47469461],
 [0.79010903, 0.40292329, 0.21457634, 0.21974706, 0.24205762],
 [0.93060762, 0.17313879, 0.13662193, 0.64922253, 0.100933  ],
 [0.82918813, 0.73733626, 0.93187481, 0.11585571, 0.90068867],
 [0.77258157, 0.29122789, 0.70944652, 0.38185224, 0.92743546],
 [0.91340333, 0.74244892, 0.33677635, 0.54126637, 0.14037251]],

[[0.91728493, 0.05009984, 0.47076479, 0.6920927 , 0.97478057],
 [0.46389238, 0.12847375, 0.94831159, 0.98797751, 0.30490679],
 [0.80133476, 0.56094137, 0.15308254, 0.43193807, 0.57064751],
 [0.37628489, 0.83054605, 0.6305017 , 0.25535137, 0.64320918],
 [0.66250613, 0.65706376, 0.16270845, 0.45466079, 0.5064182 ],
 [0.24249582, 0.47170769, 0.08026828, 0.61429308, 0.09293654],
 [0.23119477, 0.53643352, 0.38626669, 0.68762701, 0.82871353],
 [0.74944286, 0.28117867, 0.20079804, 0.63002354, 0.32860899],
 [0.75200883, 0.12031128, 0.92311068, 0.13225864, 0.92088796],
 [0.91628618, 0.44495628, 0.65840782, 0.49426194, 0.26861788],
 [0.44378598, 0.8016041 , 0.25569888, 0.45335769, 0.82044163],
 [0.53118189, 0.2659231 , 0.54654729, 0.57641777, 0.70586654],
 [0.94113929, 0.255896  , 0.79861623, 0.26108866, 0.47894094],
 [0.14083281, 0.96866513, 0.36846893, 0.7854514 , 0.3136031 ],
 [0.7923511 , 0.04894864, 0.57443337, 0.57472696, 0.79104298],
 [0.41887815, 0.71515051, 0.10359356, 0.42604943, 0.13481164],
 [0.19001503, 0.42026705, 0.67861176, 0.39456473, 0.1294563 ],
 [0.86566167, 0.7443966 , 0.40497787, 0.67885156, 0.88218298],
 [0.22080293, 0.95251947, 0.79173228, 0.06685422, 0.10783863],
 [0.0285208 , 0.20309095, 0.23266794, 0.34383154, 0.2434904 ]],

[[0.10554682, 0.91123289, 0.37732193, 0.93144489, 0.09051688],
 [0.88585796, 0.81940892, 0.32437544, 0.59106239, 0.56526253],
 [0.01856545, 0.70711987, 0.01881853, 0.13225192, 0.20622027],
 [0.36286768, 0.54504817, 0.52753472, 0.494159  , 0.10435963],
 [0.31010893, 0.94002789, 0.15039393, 0.63302127, 0.33003064],
 [0.76635132, 0.00717313, 0.85421436, 0.06335244, 0.75000228],
 [0.10345135, 0.2857124 , 0.72413423, 0.14565473, 0.0474138 ],
 [0.29237419, 0.84194109, 0.06956954, 0.20899878, 0.17181866],
 [0.52643085, 0.08645564, 0.88669334, 0.53393211, 0.0960978 ],
 [0.67445425, 0.82392063, 0.12876963, 0.14479759, 0.12624783],
 [0.28681529, 0.20531471, 0.16158122, 0.37415571, 0.34978253],
 [0.28673365, 0.00388727, 0.93429192, 0.91272908, 0.27556635],
 [0.09607903, 0.6913798 , 0.3350125 , 0.72892208, 0.20696683],
 [0.9295899 , 0.21397602, 0.92609305, 0.45979854, 0.98406901],
 [0.94602975, 0.35934825, 0.24114186, 0.77395818, 0.81804625],
 [0.11987153, 0.59393402, 0.70168952, 0.71077275, 0.13998359],
 [0.69859154, 0.06325503, 0.27709963, 0.58468266, 0.04396856],
 [0.80106068, 0.4467056 , 0.56137451, 0.32593843, 0.94627004],
 [0.65903362, 0.76226441, 0.11256198, 0.05763486, 0.21321968],
 [0.64180461, 0.52808178, 0.49471372, 0.84501033, 0.5208755 ]],

[[0.24747012, 0.96232144, 0.80169124, 0.87235563, 0.11727511],
 [0.18992006, 0.4760233 , 0.40912132, 0.29361972, 0.3798361 ],
 [0.64356251, 0.94937179, 0.78384717, 0.93681182, 0.31894441],
 [0.67478144, 0.09014771, 0.09307271, 0.72943925, 0.71376751],
 [0.65789285, 0.11370517, 0.28577664, 0.2285036 , 0.39286168],
 [0.80760704, 0.07663374, 0.42284296, 0.83685235, 0.11271565],
 [0.21673806, 0.78550806, 0.79664314, 0.55721641, 0.08208381],
 [0.49245716, 0.87108221, 0.31157486, 0.44819079, 0.065238  ],
 [0.97266568, 0.66960622, 0.82900219, 0.67004631, 0.29705526],
 [0.018538  , 0.2597019 , 0.03982513, 0.99095797, 0.83453767],
 [0.34047234, 0.09783368, 0.49691429, 0.004878  , 0.21920408],
 [0.35059387, 0.07454877, 0.05379835, 0.89971131, 0.32683925],
 [0.03079589, 0.50916063, 0.84523733, 0.87622118, 0.92342811],
 [0.04853171, 0.66308213, 0.27606947, 0.88627091, 0.61559378],
 [0.6588167 , 0.58790933, 0.95700146, 0.79410172, 0.86255979],
 [0.42617043, 0.44188415, 0.90171694, 0.31383627, 0.93513143],
 [0.91319973, 0.58959245, 0.44064868, 0.37531816, 0.67349192],
 [0.42650626, 0.51677383, 0.54306384, 0.61064281, 0.93703402],
 [0.06681412, 0.86094215, 0.82532452, 0.63732465, 0.96207762],
 [0.88976435, 0.54098737, 0.1602934 , 0.71563281, 0.35285734]]])
```

In [134…

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

Out[134…

```
array([0.27068221, 0.53287958, 0.65778207, 0.77907225, 0.72850896,
       0.86263669, 0.17769263, 0.14314313, 0.13910618])
```

In [140…

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

Out[140…

```
3
```

In [152…

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

Out[152…

```
16
```

In [154…

```
np.random.randint(0,1)
```

Out[154…

```
0
```

In [156…

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

Out[156…

```
array([18, 11, 10, 16, 17])
```

In [158…

```
np.random.randint(1,6,4)
```

Out[158…

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

In [160…

```
np.random.randint(3)
```

Out[160…

```
1
```

In [162…

```
np.random.randint(1)
```

Out[162…

```
0
```

In [164…

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

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[164], 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_integer
s._rand_int32()

ValueError: low >= high
```

In [166…

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

Out[166…

```
array([ 17, -21,  10, -30, -23,  19, -16,  16, -26, -21])
```

In [168…

```
np.arange(1,13)
```

Out[168…

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

In [170…

```
np.arange(1,13).reshape(3,4)
```

Out[170…

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

In [192…

```
np.arange(1,21)
```

Out[192…

```
array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17,
       18, 19, 20])
```

In [196…

```
np.arange(1,21).reshape(5,4)
```

Out[196…

```
array([[ 1,  2,  3,  4],
       [ 5,  6,  7,  8],
       [ 9, 10, 11, 12],
       [13, 14, 15, 16],
       [17, 18, 19, 20]])
```

array creation functions

In [8]:

```
import numpy as np
```

In [10]:

```
a=np.array([1,2,3])
print("array a:",a)
```

```
array a: [1 2 3]
```

In [14]:

```
b=np.arange(0,10,2)
print("array b:",b)
```

```
array b: [0 2 4 6 8]
```

In [16]:

```
c=np.linspace(0,1,5)
print("array c:",c)
```

```
array c: [0.   0.25 0.5  0.75 1.  ]
```

In [18]:

```
d=np.zeros((2,3))
print("array d:\n",d)
```

```
array d:
 [[0. 0. 0.]
 [0. 0. 0.]]
```

In [20]:

```
e=np.ones((3,2))
print("array e:\n",e)
```

```
array e:
 [[1. 1.]
 [1. 1.]
 [1. 1.]]
```

In [22]:

```
f=np.eye(4)
print("identity matrix f:\n",f)
```

```
identity matrix f:
 [[1. 0. 0. 0.]
 [0. 1. 0. 0.]
 [0. 0. 1. 0.]
 [0. 0. 0. 1.]]
```

array manipulation functions

In [25]:

```
a1=np.array([1,2,3])
reshaped=np.reshape(a1,(1,3))
print("reshaped array:",reshaped)
```

```
reshaped array: [[1 2 3]]
```

In [27]:

```
f1=np.array([[1,2],[3,4]])
flattened=np.ravel(f1)
print("flattened array:",reshaped)
```

```
flattened array: [[1 2 3]]
```

In [29]:

```
e1=np.array([[1,2],[3,4]])
transposed=np.transpose(e1)
print("transposed array:\n",transposed)
```

```
transposed array:
 [[1 3]
 [2 4]]
```

In [31]:

```
a2=np.array([1,2])
b2=np.array([3,4])
stacked=np.vstack([a2,b2])
print("stacked arrays:\n",stacked)
```

```
stacked arrays:
 [[1 2]
 [3 4]]
```

matematical functions

In [34]:
```
g=np.array([1,2,3,4])
added=np.add(g,2)
print("added 2 to g:",added)
```

```
added 2 to g: [3 4 5 6]
```

In [36]:
```
squared=np.power(g,2)
print("squared g:",squared)
```

```
squared g: [ 1  4  9 16]
```

In [38]:
```
sqrt_val=np.sqrt(g)
print("squared g:",squared)
```

```
squared g: [ 1  4  9 16]
```

In [40]:
```
print(a1)
print(g)
```

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

In [46]:
```
a2=np.array([1,2,3])
dot_product=np.dot(a2,g)
print("dot product of a and g:",dot_product)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[46], line 2
      1 a2=np.array([1,2,3])
----> 2 dot_product=np.dot(a2,g)
      3 print("dot product of a and g:",dot_product)

ValueError: shapes (3,) and (4,) not aligned: 3 (dim 0) != 4 (dim 0)
```

In [48]:
```
print(a)
print(a1)
```

```
[1 2 3]
[1 2 3]
```

In [50]:
```
a3=np.array([1,2,3])
dot_product=np.dot(a1,a)
print("dot product of a1 and a:",dot_product)
```

```
dot product of a1 and a: 14
```

statistical functions

In [53]:
```
s=np.array([1,2,3,4])
mean=np.mean(s)
print("mean of s:",mean)
```

```
mean of s: 2.5
```

In [55]:
```
std_dev=np.std(s)
print("standard deviation of s:",std_dev)
```

```
standard deviation of s: 1.118033988749895
```

In [59]:

```
minimum=np.min(s)
print("min of s:",minimum)
```

```
min of s: 1
```

In [61]:

```
maximum=np.max(s)
print("max of s:",maximum)
```

```
max of s: 4
```

linear algebra functions

In [64]:

```
matrix=np.array([[1,2],[3,4]])
```

In [66]:

```
determinant=np.linalg.det(matrix)
print("determinant of matrix:",determinant)
```

```
determinant of matrix: -2.0000000000000004
```

In [68]:

```
inverse=np.linalg.inv(matrix)
print("inverse of matrix:\n",inverse)
```

```
inverse of matrix:
 [[-2.   1. ]
 [ 1.5 -0.5]]
```

random sampling functions

In [73]:

```
random_vals=np.random.rand(3)
print("random values:",random_vals)
```

```
random values: [0.29053474 0.30221541 0.18684256]
```

In [75]:

```
np.random.seed(0)
random_vals=np.random.rand(3)
print("random values:",random_vals)
```

```
random values: [0.5488135  0.71518937 0.60276338]
```

In [91]:

```
rand_ints=np.random.randint(0,10,size=5)
print("random integers:",rand_ints)
```

```
random integers: [0 1 9 9 0]
```

In [93]:

```
np.random.seed(0)
rand_ints=np.random.randint(0,10,size=5)
print("random integers:",rand_ints)
```

```
random integers: [5 0 3 3 7]
```

boolean & logical functions

In [96]:

```
logical_test=np.array([True,False,True])
all_true=np.all(logical_test)
print("All elements True:",all_true)
```

```
All elements True: False
```

In [100…

```
logical_test=np.array([[True,False,True]])
all_true=np.all(logical_test)
print("All elements True:",all_true)
```

```
All elements True: False
```

In [102…

```
logical_test=np.array([[False,False,False]])
all_true=np.all(logical_test)
print("All elements True:",all_true)
```

```
All elements True: False
```

In [104…

```
any_true=np.any(logical_test)
print("any elements true:",any_true)
```

```
any elements true: False
```

set operations

In [107…

```
set_a=np.array([1,2,3,4])
set_b=np.array([3,4,5,6])
intersection=np.intersect1d(set_a,set_b)
print("intersection of a and b:",intersection)
```

```
intersection of a and b: [3 4]
```

In [109…

```
union=np.union1d(set_a,set_b)
print("union of a and b:",union)
```

```
union of a and b: [1 2 3 4 5 6]
```

array attribute functions

In [114…

```
a=np.array([1,2,3])
shape=a.shape
size=a.size
dimensions=a.ndim
dtype=a.dtype
print("shape of a:",shape)
print("size of a:",size)
print("number of dimensions of a:",dimensions)
print("data type of a:",dtype)
```

```
shape of a: (3,)
size of a: 3
number of dimensions of a: 1
data type of a: int32
```

other functions

In [117…

```
a=np.array([1,2,3])
copied_array=np.copy(a)
print("copied array:",copied_array)
```

```
copied array: [1 2 3]
```

In [119…

```
array_size_in_bytes=a.nbytes
print("size of a in bytes:",array_size_in_bytes)
```

```
size of a in bytes: 12
```

In [121…

```
shared=np.shares_memory(a,copied_array)
print("do a and copied_array share memory?",shared)
```

```
do a and copied_array share memory? False
```

In [123…

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

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

In [125…

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

Out[125…

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

In [127…

```
np.zeros((3,3))
```

Out[127…

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

In [131…

```
n=(6,7)
n1=(6,8)
print(np.zeros(n1))
```

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

In [133…

```
print(np.zeros(n,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 [135…

```
range(5)
```

Out[135…

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

In [137…

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

Out[137…

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

In [139…

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

```
0
1
2
3
4
```

In [141…

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

Out[141…

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

In [143…

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

Out[143…

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

In [145…

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

Out[145…

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

In [147…

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

Out[147…

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

In [149…

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

Out[149…

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

In [151…

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

Out[151…

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

In [153…

```
type(b)
```

Out[153…

```
numpy.ndarray
```

In [155…

```
b
```

Out[155…

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

In [157…

```
b[:]
```

Out[157…

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

In [159…

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

Out[159…

```
array([[14, 17, 16, 18],
       [18, 11, 16, 17]])
```

In [161…

```
b
```

Out[161…

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

In [163…

```
b[1,2]
```

Out[163…

```
16
```

In [165…

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

Out[165…

```
18
```

In [167…

```
b[1,-1]
```

Out[167…

```
18
```

In [169…

```
b
```

Out[169…

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

In [171…

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

Out[171…

```
array([[18, 11, 16, 17]])
```

In [173…

```
b
```

Out[173…

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

In [175…

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

Out[175…

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

In [177…

```
b
```

Out[177…

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

In [179…

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

Out[179…

```
15
```

In [181…

```
b
```

Out[181…

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

In [183…

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

Out[183…

```
13
```

In [185…

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

Out[185…

```
16
```

In [193…

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

Out[193…

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

In [195…

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

Out[195…

```
16
```

In [197…

```
b
```

Out[197…

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

In [199…

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

Out[199…

```
array([[14, 17, 16, 18]])
```

In [201…

```
b[:]
```

Out[201…

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

operations

In [204…

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

Out[204…

```
array([11, 14, 19, 18, 11, 11, 17, 19, 19, 13])
```

In [206…

```
id(a)
```

Out[206…

```
2467642772912
```

In [212…

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

In [214…

```
arr2
```

Out[214…

```
array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
       [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
       [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
       [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
       [34, 48, 93,  3, 98, 42, 77, 21, 73,  0],
       [10, 43, 58, 23, 59,  2, 98, 62, 35, 94],
       [67, 82, 46, 99, 20, 81, 50, 27, 14, 41],
       [58, 65, 36, 10, 86, 43, 11,  2, 51, 80],
       [32, 54,  0, 38, 19, 46, 42, 56, 60, 77],
       [30, 24,  2,  3, 94, 98, 13, 40, 72, 19]])
```

In [216…

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

Out[216…

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

In [218…

```
arr[:]
```

Out[218…

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

In [220…

```
arr[:4]
```

Out[220…

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

In [222…

```
arr2[:]
```

Out[222…

```
array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
       [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
       [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
       [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
       [34, 48, 93,  3, 98, 42, 77, 21, 73,  0],
       [10, 43, 58, 23, 59,  2, 98, 62, 35, 94],
       [67, 82, 46, 99, 20, 81, 50, 27, 14, 41],
       [58, 65, 36, 10, 86, 43, 11,  2, 51, 80],
       [32, 54,  0, 38, 19, 46, 42, 56, 60, 77],
       [30, 24,  2,  3, 94, 98, 13, 40, 72, 19]])
```

In [224…

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

Out[224…

```
array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
       [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
       [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
       [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
       [34, 48, 93,  3, 98, 42, 77, 21, 73,  0]])
```

In [226…

```
arr2
```

Out[226…

```
array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
       [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
       [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
       [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
       [34, 48, 93,  3, 98, 42, 77, 21, 73,  0],
       [10, 43, 58, 23, 59,  2, 98, 62, 35, 94],
       [67, 82, 46, 99, 20, 81, 50, 27, 14, 41],
       [58, 65, 36, 10, 86, 43, 11,  2, 51, 80],
       [32, 54,  0, 38, 19, 46, 42, 56, 60, 77],
       [30, 24,  2,  3, 94, 98, 13, 40, 72, 19]])
```

In [228…

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

Out[228…

```
array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
       [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
       [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
       [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
       [34, 48, 93,  3, 98, 42, 77, 21, 73,  0]])
```

In [230…

```
arr2[1,4]
```

Out[230…

```
68
```

In [232…

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

Out[232…

```
2
```

In [234…

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

Out[234…

```
60
```

In [238…

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

Out[238…

```
2
```

In [240…

```
arr2
```

Out[240…

```
array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
       [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
       [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
       [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
       [34, 48, 93,  3, 98, 42, 77, 21, 73,  0],
       [10, 43, 58, 23, 59,  2, 98, 62, 35, 94],
       [67, 82, 46, 99, 20, 81, 50, 27, 14, 41],
       [58, 65, 36, 10, 86, 43, 11,  2, 51, 80],
       [32, 54,  0, 38, 19, 46, 42, 56, 60, 77],
       [30, 24,  2,  3, 94, 98, 13, 40, 72, 19]])
```

In [242…

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

Out[242…

```
72
```

In [244…

```
arr2
```

```
Out[244…  array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
                 [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
                 [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
                 [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
                 [34, 48, 93,  3, 98, 42, 77, 21, 73,  0],
                 [10, 43, 58, 23, 59,  2, 98, 62, 35, 94],
                 [67, 82, 46, 99, 20, 81, 50, 27, 14, 41],
                 [58, 65, 36, 10, 86, 43, 11,  2, 51, 80],
                 [32, 54,  0, 38, 19, 46, 42, 56, 60, 77],
                 [30, 24,  2,  3, 94, 98, 13, 40, 72, 19]])
```

```
In [246…  arr2[::-1]
```

```
Out[246…  array([[30, 24,  2,  3, 94, 98, 13, 40, 72, 19],
                 [32, 54,  0, 38, 19, 46, 42, 56, 60, 77],
                 [58, 65, 36, 10, 86, 43, 11,  2, 51, 80],
                 [67, 82, 46, 99, 20, 81, 50, 27, 14, 41],
                 [10, 43, 58, 23, 59,  2, 98, 62, 35, 94],
                 [34, 48, 93,  3, 98, 42, 77, 21, 73,  0],
                 [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
                 [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
                 [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
                 [11, 46, 82, 91,  0, 14, 99, 53, 12, 42]])
```

```
In [248…  arr2
```

```
Out[248…  array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
                 [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
                 [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
                 [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
                 [34, 48, 93,  3, 98, 42, 77, 21, 73,  0],
                 [10, 43, 58, 23, 59,  2, 98, 62, 35, 94],
                 [67, 82, 46, 99, 20, 81, 50, 27, 14, 41],
                 [58, 65, 36, 10, 86, 43, 11,  2, 51, 80],
                 [32, 54,  0, 38, 19, 46, 42, 56, 60, 77],
                 [30, 24,  2,  3, 94, 98, 13, 40, 72, 19]])
```

```
In [250…  arr2[::-2]
```

```
Out[250…  array([[30, 24,  2,  3, 94, 98, 13, 40, 72, 19],
                 [58, 65, 36, 10, 86, 43, 11,  2, 51, 80],
                 [10, 43, 58, 23, 59,  2, 98, 62, 35, 94],
                 [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
                 [84, 75, 68,  6, 68, 47,  3, 76, 52, 78]])
```

```
In [252…  arr2[::-3]
```

```
Out[252…  array([[30, 24,  2,  3, 94, 98, 13, 40, 72, 19],
                 [67, 82, 46, 99, 20, 81, 50, 27, 14, 41],
                 [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
                 [11, 46, 82, 91,  0, 14, 99, 53, 12, 42]])
```

```
In [254…  arr2[:-3]
```

```
Out[254…  array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
                 [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
                 [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
                 [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
                 [34, 48, 93,  3, 98, 42, 77, 21, 73,  0],
                 [10, 43, 58, 23, 59,  2, 98, 62, 35, 94],
                 [67, 82, 46, 99, 20, 81, 50, 27, 14, 41]])
```

```
In [457…  arr2[:3]
```

```
Out[457…  array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
                 [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
                 [15, 20, 99, 58, 23, 79, 13, 85, 48, 49]])
```

```
In [256…  arr
```

```
Out[256…  [0, 1, 2, 3, 4, 5]
```

```
In [261…  arr2.max()
```

```
Out[261…  99
```

```
In [263…  arr2.min()
```

```
Out[263…  0
```

```
In [265…  arr2
```

```
Out[265…  array([[11, 46, 82, 91,  0, 14, 99, 53, 12, 42],
                 [84, 75, 68,  6, 68, 47,  3, 76, 52, 78],
                 [15, 20, 99, 58, 23, 79, 13, 85, 48, 49],
                 [69, 41, 35, 64, 95, 69, 94,  0, 50, 36],
                 [34, 48, 93,  3, 98, 42, 77, 21, 73,  0],
                 [10, 43, 58, 23, 59,  2, 98, 62, 35, 94],
                 [67, 82, 46, 99, 20, 81, 50, 27, 14, 41],
                 [58, 65, 36, 10, 86, 43, 11,  2, 51, 80],
                 [32, 54,  0, 38, 19, 46, 42, 56, 60, 77],
                 [30, 24,  2,  3, 94, 98, 13, 40, 72, 19]])
```

```
In [267…  arr2.mean()
```

```
Out[267…  48.1
```

```
In [269…  arr2.median()
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[269], line 1
----> 1 arr2.median()

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

```
In [271…  from numpy import*
          a=array([1,2,3,4,9])
          median(a)
```

```
Out[271…  3.0
```

indexing

In [274…

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

In [276…

```
mat
```

Out[276…

```
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 [278…

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

In [280…

```
col
```

Out[280…

```
5
```

In [282…

```
row
```

Out[282…

```
4
```

In [284…

```
mat
```

Out[284…

```
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 [286…

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

Out[286…

```
45
```

In [288…

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

Out[288…

```
45
```

In [290…

```
mat
```

Out[290…

```
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 [292…

```
mat[:]
```

Out[292…

```
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 [294…

```
col=6
```

In [296…

```
mat
```

Out[296…

```
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 [298…

```
mat[6]
```

Out[298…

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

In [300…

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

Out[300…

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

In [302…

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

Out[302…

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

In [304…

```
mat
```

Out[304…
```
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 [306…
```
mat[:,8]
```

Out[306…
```
array([ 8, 18, 28, 38, 48, 58, 68, 78, 88, 98])
```

In [308…
```
mat[:col]
```

Out[308…
```
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 [310…
```
mat[:6]
```

Out[310…
```
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 [312…
```
row
```

Out[312…
```
4
```

In [316…
```
mat
```

Out[316…
```
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 [318…
```
mat[row:]
```

Out[318…
```
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 [320…

```
mat[:]
```

Out[320…

```
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 [322…

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

Out[322…

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

In [324…

```
mat
```

Out[324…

```
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 [326…

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

Out[326…

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

In [328…

```
mat
```

Out[328…

```
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 [330…

```
row
```

Out[330…

```
4
```

In [332…

```
col
```

Out[332…

```
6
```

In [334…

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

Out[334…

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

In [338…

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

Out[338…

```
14
```

In [340…

```
mat
```

Out[340…

```
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 [342…

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

Out[342…

```
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 [344…

```
mat[0]
```

Out[344…

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

In [346…

```
mat[6:]
```

Out[346…

```
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 [348…

```
mat[:6]
```

Out[348…

```
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 [350…

```
mat
```

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]])
```

In [352…

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

```
Out[352…  array([[50, 51, 52, 53, 54, 55, 56, 57, 58, 59],
                 [60, 61, 62, 63, 64, 65, 66, 67, 68, 69]])
```

```
In [354…  mat[0:10]
```

```
Out[354…  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 [356…  mat[0:10:3]
```

```
Out[356…  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 [358…  mat[0:10]
```

```
Out[358…  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 [362…  mat[0:10:3]
```

```
Out[362…  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 [364…  mat
```

```
Out[364…  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 [366…  mat[:4]
```

Out[366…

```
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 [368…

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

Out[368…

```
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 [370…

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

Out[370…

```
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 [372…

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

Out[372…

```
array([[90, 91, 92, 93, 94, 95, 96, 97, 98, 99],
       [40, 41, 42, 43, 44, 45, 46, 47, 48, 49]])
```

In [374…

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

Out[374…

```
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 [376…

```
mat
```

Out[376…

```
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 [378…

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

Out[378…

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

In [380…

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

Out[380…

```
1
```

In [382…

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

Out[382…

```
16
```

In [384…

```
mat
```

Out[384…

```
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 [386…

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

Out[386…

```
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 [388…

```
mat[1:]
```

Out[388…

```
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 [390…

```
mat[:6]
```

Out[390…

```
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 [392…

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

Out[392…

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

In [394…

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

Out[394…

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

In [396…

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

Out[396…

```
35
```

In [398…

```
mat
```

Out[398…

```
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 [400…

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

Out[400…

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

In [402…

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

Out[402…

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

In [404…

```
mat
```

Out[404…

```
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 [406…

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

Out[406…

```
array([[23, 24],
       [33, 34]])
```

In [408…

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

Out[408…

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

In [410…

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

Out[410…

```
array([[24]])
```

masking

In [415…

```
mat
```

```
Out[415…  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 [417…  id(mat)
```

```
Out[417…  2467643074448
```

```
In [461…  mat<50
```

```
Out[461…  array([False])
```

```
In [419…  mat[mat<50]
```

```
Out[419…  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 [421…  mat[mat<=50]
```

```
Out[421…  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])
```

```
In [423…  mat>50
```

```
Out[423…  array([[False, False, False, False, False, False, False, False, False,
                  False],
                 [False, False, False, False, False, False, False, False, False,
                  False],
                 [False, False, False, False, False, False, False, False, False,
                  False],
                 [False, False, False, False, False, False, False, False, False,
                  False],
                 [False, False, False, False, False, False, False, False, False,
                  False],
                 [False,  True,  True,  True,  True,  True,  True,  True,  True,
                   True],
                 [ True,  True,  True,  True,  True,  True,  True,  True,  True,
                   True],
                 [ True,  True,  True,  True,  True,  True,  True,  True,  True,
                   True],
                 [ True,  True,  True,  True,  True,  True,  True,  True,  True,
                   True],
                 [ True,  True,  True,  True,  True,  True,  True,  True,  True,
                   True]])
```

```
In [425…  mat[mat==50]
```

```
Out[425…  array([50])
```

```
In [427…  mat
```

Out[427…
```
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 [429…
```
mat==50
```

Out[429…
```
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]])
```

In [488…
```
mat
```

Out[488…
```
array([50])
```

In [469…
```
mat[mat!=50]
```

Out[469…
```
array([], dtype=int32)
```

In [470…
```
a1=mat[mat<50]
a1
```

Out[470…
```
array([], dtype=int32)
```

In [471…
```
mat
```

Out[471…
```
array([50])
```

In [472…
```
a2=mat[mat>50]
a2
```

Out[472…
```
array([], dtype=int32)
```

In [473…
```
a3=mat[mat<=50]
a3
```

Out[473…

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

In [474…

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

Out[474…

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

In [475…

```
import random

def generate_otp(length=4):
    """Generate a numeric OTP of a specified length."""
    digits='012345'
    otp= ''.join(random.choice(digits) for _ in range(length))
    return otp
# example usage
otp_length=4
otp=generate_otp(otp_length)
print(f"your OTP is:{otp}")
```

```
your OTP is:3415
```

In [495…

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

In [501…

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

In [503…

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

```
<built-in method pop of list object at 0x0000023E8B1BE180>
```

In [505…

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

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

In [509…

```
arr=np.array([1,2,3,4,5])
arr
```

Out[509…

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

In [511…

```
zeros_arr=np.zeros(5)
ones_arr=np.ones(5)
print(zeros_arr)
print(ones_arr)
```

```
[0. 0. 0. 0. 0.]
[1. 1. 1. 1. 1.]
```

In [515…

```
range_arr=np.random.rand(3,3)
range_arr
```

Out[515…

```
array([[0.41182014, 0.67543908, 0.24979628],
       [0.31321833, 0.96541622, 0.58846509],
       [0.65966841, 0.53320625, 0.23053302]])
```

In [519…

```
arr1=np.array([2,3,4,5,6])
arr2=np.array([5,6,7,8,9])
print(arr1)
print(arr2)
result=arr1+arr2
print(result)
```

```
[2 3 4 5 6]
[5 6 7 8 9]
[ 7  9 11 13 15]
```

In [523…

```
result1=arr1-arr2
result2=arr1*arr2
result3=arr1/arr2
result4=arr1%arr2
print(result1)
print(result2)
print(result3)
print(result4)
```

```
[-3 -3 -3 -3 -3]
[10 18 28 40 54]
[0.4        0.5        0.57142857 0.625      0.66666667]
[2 3 4 5 6]
```

In [525…

```
result=np.square(arr)
result
```

Out[525…

```
array([ 1,  4,  9, 16, 25])
```

In [529…

```
result=np.sqrt(arr)
result
```

Out[529…

```
array([1.        , 1.41421356, 1.73205081, 2.        , 2.23606798])
```

In [531…

```
result=np.exp(arr)
result
```

Out[531…

```
array([  2.71828183,   7.3890561 ,  20.08553692,  54.59815003,
       148.4131591 ])
```

In [533…

```
dot_product=np.dot(arr1,arr2)
dot_product
```

Out[533…

```
150
```

In [535…

```
result=arr+5
result
```

Out[535…

```
array([ 6,  7,  8,  9, 10])
```

In [555…

```
combined_arr=np.concatenate((arr1,arr2),axis=0)
combined_arr
```

Out[555…

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

In [557…

```
stacked_arr=np.vstack((arr1,arr2))
stacked_arr
```

Out[557…

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

In [562…

```
new_arr=arr.copy()
new_arr
```

Out[562…

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

In [564…

```
has_nan=np.isnan(arr).any()
has_nan
```

Out[564…

```
False
```

In [568…

```
result=np.sin(arr)
result
```

Out[568…

```
array([ 0.84147098,  0.90929743,  0.14112001, -0.7568025 , -0.95892427])
```

In [574…

```
size_in_bytes=arr.nbytes
size_in_bytes
```

Out[574…

```
20
```

In [ ]:

In [ ]:

In [ ]:

In [ ]: