

```
In [1]: import numpy as np  
  
In [2]: np.__version__  
  
Out[2]: '2.1.3'  
  
In [2]: import sys  
sys.version  
  
Out[2]: '3.13.5 | packaged by Anaconda, Inc. | (main, Jun 12 2025, 16:37:03) [MSC v.1929 6  
4 bit (AMD64)]'
```

Creating Arrays

```
In [3]: my_list = [0,1,2,3,4,5]  
my_list  
  
Out[3]: [0, 1, 2, 3, 4, 5]  
  
In [3]: type(my_list)  
  
Out[3]: list  
  
In [4]: arr  
  
-----  
NameError                                                 Traceback (most recent call last)  
Cell In[4], line 1  
----> 1 arr  
  
NameError: name 'arr' is not defined  
  
In [4]: arr = np.array(my_list)  
  
In [7]: arr  
  
Out[7]: array([0, 1, 2, 3, 4, 5])  
  
In [8]: type(arr)  
  
Out[8]: numpy.ndarray  
  
In [5]: type(my_list)  
  
Out[5]: list  
  
In [10]: np. # we Learn important function
```

```
Cell In[10], line 1
    np. # we learn important function
    ^
SyntaxError: invalid syntax
```

arange

```
In [7]: np.arange(3.0)
```

```
NameError                                 Traceback (most recent call last)
Cell In[7], line 1
----> 1 np.arange(3.0)

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

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

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

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

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

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

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

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

```
Out[15]: array([], dtype=int64)
```

```
In [16]: np.arange(-20,10)
```

```
Out[16]: 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 [17]: np.arange(-16,10)
```

```
Out[17]: 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 [18]: np.arange(-20,-10)
```

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

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

```
Out[20]: array([], dtype=int64)
```

```
In [22]: ar = np.arange(-30,20)
ar
```

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

```
In [23]: np.arange(10,10)
```

```
Out[23]: array([], dtype=int64)
```

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

```
-----  

TypeError                                     Traceback (most recent call last)  

Cell In[25], line 1  

----> 1 np.arange()  


```

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

```
In [27]: np.arange(10,30,5) # 10- starting from 30- end point 5 - step count
```

```
Out[27]: array([10, 15, 20, 25])
```

```
In [28]: np.arange(0,10,3)
```

```
Out[28]: array([0, 3, 6, 9])
```

```
In [29]: np.arange(10,30,5,8)
```

```
-----  

TypeError                                     Traceback (most recent call last)  

Cell In[29], line 1  

----> 1 np.arange(10,30,5,8)  


```

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

zeros

```
In [30]: np.zeros(3) #parameter tunning
```

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

```
In [32]: np.zeros(5, dtype=int) # hyperparameter tunning
```

```
Out[32]: array([0, 0, 0, 0, 0])
```

```
In [33]: np.zeros((2,2), dtype=int)
```

```
Out[33]: array([[0, 0],
 [0, 0]])
```

```
In [34]: zero = np.zeros([2,2])
print(zero)
print(type(zero))
```

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

```
In [35]: zero = np.zeros([2,2])
print(zero)
```

```
print('#####')
print(type(zero))
```

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

```
In [36]: np.zeros((2,10))
```

```
Out[36]: array([[0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]])
```

```
In [37]: np.zeros((2,2))
```

```
Out[37]: array([[0., 0.],
 [0., 0.]])
```

```
In [38]: np.zeros((3,3))
```

```
Out[38]: array([[0., 0., 0.],
 [0., 0., 0.],
 [0., 0., 0.]])
```

```
In [39]: np.zeros((10,30))
```

```
In [41]: np.zeros((5,10)) # by default large -- will give row & 2nd arg - columns
```

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

```
In [42]: n = (6,7)
          n1 = (6,8)
          print(np.zeros(n1)) # parameter tuning
```

```
In [43]: print(np.zeros(n,dtype=int)) # hyperparameter tunning
```

```

[[0 0 0 0 0 0]
 [0 0 0 0 0 0]
 [0 0 0 0 0 0]
 [0 0 0 0 0 0]
 [0 0 0 0 0 0]
 [0 0 0 0 0 0]
 [0 0 0 0 0 0]]

```

In [44]: n

```
Out[44]: (6, 7)
```

In [45]: n1

```
Out[45]: (6, 8)
```

```
In [46]: 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.]]
```

Ones

```
In [47]: np.ones(3)
```

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

```
In [48]: np.ones(4, dtype=int)
```

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

```
In [49]: n
```

```
Out[49]: (6, 7)
```

```
In [50]: np.ones(n)
```

```
Out[50]: 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 [51]: np.ones((5,4),dtype=int) # by default 5- rows & 4 - columns
```

```
Out[51]: array([[1, 1, 1, 1],  
 [1, 1, 1, 1],  
 [1, 1, 1, 1],  
 [1, 1, 1, 1],  
 [1, 1, 1, 1]])
```

```
In [54]: np.twos((2,3)) # two not valid
```

```
-----
AttributeError                                         Traceback (most recent call last)
Cell In[54], line 1
----> 1 np.twos((2,3))

File ~\anaconda3\Lib\site-packages\numpy\__init__.py:414, in __getattr__(attr)
 411     import numpy.char as char
 412     return char.chararray
--> 414 raise AttributeError("module {!r} has no attribute "
 415                         "{!r}".format(__name__, attr))

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

In [55]: `np.three(2,3) # three not valid`

```
-----
AttributeError                                         Traceback (most recent call last)
Cell In[55], line 1
----> 1 np.three(2,3)

File ~\anaconda3\Lib\site-packages\numpy\__init__.py:414, in __getattr__(attr)
 411     import numpy.char as char
 412     return char.chararray
--> 414 raise AttributeError("module {!r} has no attribute "
 415                         "{!r}".format(__name__, attr))

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

In [56]: `np.ones(2)`

Out[56]: `array([1., 1.])`

In [57]: `np.ones((2,4))`

Out[57]: `array([[1., 1., 1., 1.],
 [1., 1., 1., 1.]])`

In [58]: `np.ones((6,10), dtype = int)`

Out[58]: `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]])`

range

In [59]: `range(5)`

Out[59]: `range(0, 5)`

```
In [60]: r = range(5)
r
```

```
Out[60]: range(0, 5)
```

```
In [61]: for i in r:
    print(i)
```

```
0
1
2
3
4
```

```
In [62]: list(range(5))
```

```
Out[62]: [0, 1, 2, 3, 4]
```

```
In [63]: range(1,10)
```

```
Out[63]: range(1, 10)
```

```
In [64]: list(range(1,10))
```

```
Out[64]: [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
In [68]: list(range(1,10,3))
```

```
Out[68]: [1, 4, 7]
```

```
In [69]: y = list(range(12))
y
```

```
Out[69]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
```

rand

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

```
-----
```

```
NameError
Cell In[70], line 1
----> 1 rand(3,2)
```

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

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

```
In [71]: rand(3,2)
random.rand(3,2)
```

```
NameError Traceback (most recent call last)
Cell In[71], line 1
----> 1 rand(3,2)
      2 random.rand(3,2)
```

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

```
In [74]: np.random.rand(5)
```

```
Out[74]: array([0.95770689, 0.45131413, 0.85741789, 0.01068859, 0.20419548])
```

```
In [75]: np.rand(5)
```

```
AttributeError Traceback (most recent call last)
Cell In[75], line 1
----> 1 np.rand(5)

File ~\anaconda3\Lib\site-packages\numpy\__init__.py:414, in __getattr__(attr)
 411     import numpy.char as char
 412     return char.chararray
--> 414 raise AttributeError("module {!r} has no attribute "
 415                         "{!r}".format(__name__, attr))
```

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

```
In [77]: np.random.rand(2,4)
```

```
Out[77]: array([[0.931066 , 0.46329846, 0.53847278, 0.0758648 ],
 [0.67386528, 0.50112929, 0.74883727, 0.55926141]])
```

```
In [81]: np.random.randint(2,20) # 2nd argument is exclusive
```

```
Out[81]: 4
```

```
In [82]: np.random.randint(0,1)
```

```
Out[82]: 0
```

```
In [85]: np.random.randint(10,20,5)
```

```
Out[85]: array([10, 10, 11, 14, 13], dtype=int32)
```

```
In [86]: np.random.randint(1,6,4)
```

```
Out[86]: array([3, 5, 3, 2], dtype=int32)
```

```
In [87]: np.random.rand(3)
```

```
Out[87]: array([0.45927199, 0.1570465 , 0.89060368])
```

```
In [88]: np.random.randint(1)
```

Out[88]: 0

In [89]: np.random.randint(30,20,10)

```
-----
ValueError                                Traceback (most recent call last)
Cell In[89], line 1
----> 1 np.random.randint(30,20,10)

File numpy\random\mtrand.pyx:796, in numpy.random.mtrand.RandomState.randint()

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

ValueError: low >= high
```

In [90]: np.random.randint(-30,20,10)

Out[90]: array([-7, 3, -12, -10, -13, -6, -4, -14, -13, -13], dtype=int32)

In [91]: np.random.randint(20,30,10)

Out[91]: array([22, 21, 25, 28, 24, 28, 28, 28, 29, 28], dtype=int32)

In [94]: np.random.randint(5,9) # get the value <=1 & >=5

Out[94]: 5

In [96]: np.random.randint(10,40,(10,10)) #generre the element 10 -30 with 4*4 mtri

Out[96]: array([[38, 22, 32, 12, 29, 21, 15, 33, 10, 24],
 [19, 20, 27, 17, 30, 31, 17, 36, 37, 36],
 [39, 19, 19, 31, 14, 37, 31, 38, 12, 37],
 [29, 25, 38, 35, 33, 21, 31, 25, 28, 23],
 [29, 23, 31, 23, 15, 16, 17, 32, 28, 39],
 [16, 35, 15, 30, 35, 26, 25, 27, 37, 17],
 [11, 17, 25, 29, 11, 10, 25, 16, 24, 33],
 [16, 12, 37, 31, 25, 39, 34, 18, 24, 11],
 [32, 19, 30, 36, 24, 13, 16, 11, 10, 29],
 [39, 25, 37, 15, 14, 37, 31, 33, 33]], dtype=int32)

In [97]: np.arange(1,13).reshape(3,4)

Out[97]: array([[1, 2, 3, 4],
 [5, 6, 7, 8],
 [9, 10, 11, 12]])

In [98]: np.arange(1,13).reshape(12,1)

```
Out[98]: array([[ 1],  
 [ 2],  
 [ 3],  
 [ 4],  
 [ 5],  
 [ 6],  
 [ 7],  
 [ 8],  
 [ 9],  
 [10],  
 [11],  
 [12]])
```

slicing

```
In [2]: b = np.random.randint(10,20,(5,4))  
b
```

```
NameError Traceback (most recent call last)  
Cell In[2], line 1  
----> 1 b = np.random.randint(10,20,(5,4))  
      2 b  
  
NameError: name 'np' is not defined
```

```
In [101... type(b)
```

```
Out[101... numpy.ndarray
```

```
In [102... b
```

```
Out[102... array([[17, 11, 17, 19],  
 [15, 17, 12, 17],  
 [11, 16, 16, 17],  
 [19, 10, 13, 15],  
 [15, 10, 14, 18]], dtype=int32)
```

```
In [103... b[:]
```

```
Out[103... array([[17, 11, 17, 19],  
 [15, 17, 12, 17],  
 [11, 16, 16, 17],  
 [19, 10, 13, 15],  
 [15, 10, 14, 18]], dtype=int32)
```

```
In [104... b
```

```
Out[104... array([[17, 11, 17, 19],  
 [15, 17, 12, 17],  
 [11, 16, 16, 17],  
 [19, 10, 13, 15],  
 [15, 10, 14, 18]], dtype=int32)
```

```
In [109... b[1:3]
```

```
Out[109... array([[15, 17, 12, 17],  
                   [11, 16, 16, 17]], dtype=int32)
```

```
In [110... b
```

```
Out[110... array([[17, 11, 17, 19],  
                   [15, 17, 12, 17],  
                   [11, 16, 16, 17],  
                   [19, 10, 13, 15],  
                   [15, 10, 14, 18]], dtype=int32)
```

```
In [111... b[1,2]
```

```
Out[111... np.int32(12)
```

```
In [112... b
```

```
Out[112... array([[17, 11, 17, 19],  
                   [15, 17, 12, 17],  
                   [11, 16, 16, 17],  
                   [19, 10, 13, 15],  
                   [15, 10, 14, 18]], dtype=int32)
```

```
In [113... b[1,3]
```

```
Out[113... np.int32(17)
```

```
In [114... b[1,-1]
```

```
Out[114... np.int32(17)
```

```
In [115... b
```

```
Out[115... array([[17, 11, 17, 19],  
                   [15, 17, 12, 17],  
                   [11, 16, 16, 17],  
                   [19, 10, 13, 15],  
                   [15, 10, 14, 18]], dtype=int32)
```

```
In [116... b[2:3]
```

```
Out[116... array([[11, 16, 16, 17]], dtype=int32)
```

```
In [117... b
```

```
Out[117... array([[17, 11, 17, 19],  
                   [15, 17, 12, 17],  
                   [11, 16, 16, 17],  
                   [19, 10, 13, 15],  
                   [15, 10, 14, 18]], dtype=int32)
```

```
In [118... b[0:-2]
```

```
Out[118]: array([[17, 11, 17, 19],  
                 [15, 17, 12, 17],  
                 [11, 16, 16, 17]], dtype=int32)
```

```
In [119]: b[-4:2]
```

```
Out[119]: array([[15, 17, 12, 17]], dtype=int32)
```

Operations

```
In [6]: a = np.random.randint(10,20,10)  
a
```

```
Out[6]: array([13, 17, 12, 19, 18, 17, 14, 10, 14, 10], dtype=int32)
```

```
In [ ]:
```

```
In [122]: id(a)
```

```
Out[122]: 1717864859664
```

```
In [7]: arr
```

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

```
In [8]: arr2 = np.random.randint(0,100,(10,10))  
arr2
```

```
Out[8]: array([[86, 10, 12, 91, 75, 38, 69, 46, 66, 22],  
               [36, 67, 55, 73, 32, 98, 14, 40, 45, 56],  
               [ 9, 29, 22, 61, 34, 20,  7,  6, 58, 82],  
               [19, 65, 85, 31, 85, 99, 70, 60, 41, 22],  
               [18, 19, 28, 99, 41, 13, 60, 88, 57, 97],  
               [50, 24, 15, 31, 87, 69, 16, 82, 70, 70],  
               [83, 24, 31, 34,  2,  6, 30, 80, 34, 10],  
               [78, 67, 71, 50, 35, 94,  4, 95, 19, 66],  
               [83, 65, 80, 33, 29, 63, 39, 47, 27, 61],  
               [35, 25, 46,  4, 80, 30, 37, 27, 91, 55]], dtype=int32)
```

```
In [127]: arr
```

```
Out[127]: array([0, 1, 2, 3, 4, 5])
```

```
In [128]: arr[:4]
```

```
Out[128]: array([0, 1, 2, 3])
```

```
In [129]: arr2
```

```
Out[129]: array([[91, 74, 4, 75, 24, 20, 2, 66, 53, 77],  
   [23, 79, 31, 83, 17, 99, 49, 98, 21, 34],  
   [64, 44, 91, 21, 20, 41, 5, 68, 33, 90],  
   [92, 5, 82, 74, 87, 47, 20, 23, 40, 6],  
   [50, 41, 7, 75, 50, 1, 89, 22, 95, 20],  
   [60, 10, 97, 43, 33, 69, 31, 6, 81, 18],  
   [58, 12, 45, 63, 19, 7, 73, 70, 7, 48],  
   [43, 18, 69, 79, 95, 56, 81, 8, 59, 94],  
   [77, 91, 29, 3, 40, 97, 23, 52, 44, 85],  
   [44, 63, 63, 13, 17, 20, 8, 92, 66, 27]], dtype=int32)
```

```
In [131]: arr2[:]
```

```
Out[131]: array([[91, 74, 4, 75, 24, 20, 2, 66, 53, 77],  
   [23, 79, 31, 83, 17, 99, 49, 98, 21, 34],  
   [64, 44, 91, 21, 20, 41, 5, 68, 33, 90],  
   [92, 5, 82, 74, 87, 47, 20, 23, 40, 6],  
   [50, 41, 7, 75, 50, 1, 89, 22, 95, 20],  
   [60, 10, 97, 43, 33, 69, 31, 6, 81, 18],  
   [58, 12, 45, 63, 19, 7, 73, 70, 7, 48],  
   [43, 18, 69, 79, 95, 56, 81, 8, 59, 94],  
   [77, 91, 29, 3, 40, 97, 23, 52, 44, 85],  
   [44, 63, 63, 13, 17, 20, 8, 92, 66, 27]], dtype=int32)
```

```
In [132]: arr2[0:5]
```

```
Out[132]: array([[91, 74, 4, 75, 24, 20, 2, 66, 53, 77],  
   [23, 79, 31, 83, 17, 99, 49, 98, 21, 34],  
   [64, 44, 91, 21, 20, 41, 5, 68, 33, 90],  
   [92, 5, 82, 74, 87, 47, 20, 23, 40, 6],  
   [50, 41, 7, 75, 50, 1, 89, 22, 95, 20]], dtype=int32)
```

```
In [133]: arr2[1,4]
```

```
Out[133]: np.int32(17)
```

```
In [23]: arr2
```

```
Out[23]: array([[86, 10, 12, 91, 75, 38, 69, 46, 66, 22],  
   [36, 67, 55, 73, 32, 98, 14, 40, 45, 56],  
   [9, 29, 22, 61, 34, 20, 7, 6, 58, 82],  
   [19, 65, 85, 31, 85, 99, 70, 60, 41, 22],  
   [18, 19, 28, 99, 41, 13, 60, 88, 57, 97],  
   [50, 24, 15, 31, 87, 69, 16, 82, 70, 70],  
   [83, 24, 31, 34, 2, 6, 30, 80, 34, 10],  
   [78, 67, 71, 50, 35, 94, 4, 95, 19, 66],  
   [83, 65, 80, 33, 29, 63, 39, 47, 27, 61],  
   [35, 25, 46, 4, 80, 30, 37, 27, 91, 55]], dtype=int32)
```

```
In [24]: arr2[-5,5]
```

```
Out[24]: np.int32(69)
```

```
In [25]: arr2[-5,-5]
```

```
Out[25]: np.int32(69)
```

```
In [26]: arr2[-1,-2]
```

```
Out[26]: np.int32(91)
```

```
In [27]: arr2
```

```
Out[27]: array([[86, 10, 12, 91, 75, 38, 69, 46, 66, 22],  
                 [36, 67, 55, 73, 32, 98, 14, 40, 45, 56],  
                 [ 9, 29, 22, 61, 34, 20, 7, 6, 58, 82],  
                 [19, 65, 85, 31, 85, 99, 70, 60, 41, 22],  
                 [18, 19, 28, 99, 41, 13, 60, 88, 57, 97],  
                 [50, 24, 15, 31, 87, 69, 16, 82, 70, 70],  
                 [83, 24, 31, 34, 2, 6, 30, 80, 34, 10],  
                 [78, 67, 71, 50, 35, 94, 4, 95, 19, 66],  
                 [83, 65, 80, 33, 29, 63, 39, 47, 27, 61],  
                 [35, 25, 46, 4, 80, 30, 37, 27, 91, 55]], dtype=int32)
```

```
In [28]: arr2[::-1]
```

```
Out[28]: array([[35, 25, 46, 4, 80, 30, 37, 27, 91, 55],  
                 [83, 65, 80, 33, 29, 63, 39, 47, 27, 61],  
                 [78, 67, 71, 50, 35, 94, 4, 95, 19, 66],  
                 [83, 24, 31, 34, 2, 6, 30, 80, 34, 10],  
                 [50, 24, 15, 31, 87, 69, 16, 82, 70, 70],  
                 [18, 19, 28, 99, 41, 13, 60, 88, 57, 97],  
                 [19, 65, 85, 31, 85, 99, 70, 60, 41, 22],  
                 [ 9, 29, 22, 61, 34, 20, 7, 6, 58, 82],  
                 [36, 67, 55, 73, 32, 98, 14, 40, 45, 56],  
                 [86, 10, 12, 91, 75, 38, 69, 46, 66, 22]], dtype=int32)
```

```
In [29]: arr2
```

```
Out[29]: array([[86, 10, 12, 91, 75, 38, 69, 46, 66, 22],  
                 [36, 67, 55, 73, 32, 98, 14, 40, 45, 56],  
                 [ 9, 29, 22, 61, 34, 20, 7, 6, 58, 82],  
                 [19, 65, 85, 31, 85, 99, 70, 60, 41, 22],  
                 [18, 19, 28, 99, 41, 13, 60, 88, 57, 97],  
                 [50, 24, 15, 31, 87, 69, 16, 82, 70, 70],  
                 [83, 24, 31, 34, 2, 6, 30, 80, 34, 10],  
                 [78, 67, 71, 50, 35, 94, 4, 95, 19, 66],  
                 [83, 65, 80, 33, 29, 63, 39, 47, 27, 61],  
                 [35, 25, 46, 4, 80, 30, 37, 27, 91, 55]], dtype=int32)
```

```
In [31]: arr2[::-2]
```

```
Out[31]: array([[35, 25, 46, 4, 80, 30, 37, 27, 91, 55],  
                 [78, 67, 71, 50, 35, 94, 4, 95, 19, 66],  
                 [50, 24, 15, 31, 87, 69, 16, 82, 70, 70],  
                 [19, 65, 85, 31, 85, 99, 70, 60, 41, 22],  
                 [36, 67, 55, 73, 32, 98, 14, 40, 45, 56]], dtype=int32)
```

```
In [32]: arr2
```

```
Out[32]: array([[86, 10, 12, 91, 75, 38, 69, 46, 66, 22],  
                 [36, 67, 55, 73, 32, 98, 14, 40, 45, 56],  
                 [ 9, 29, 22, 61, 34, 20, 7, 6, 58, 82],  
                 [19, 65, 85, 31, 85, 99, 70, 60, 41, 22],  
                 [18, 19, 28, 99, 41, 13, 60, 88, 57, 97],  
                 [50, 24, 15, 31, 87, 69, 16, 82, 70, 70],  
                 [83, 24, 31, 34, 2, 6, 30, 80, 34, 10],  
                 [78, 67, 71, 50, 35, 94, 4, 95, 19, 66],  
                 [83, 65, 80, 33, 29, 63, 39, 47, 27, 61],  
                 [35, 25, 46, 4, 80, 30, 37, 27, 91, 55]], dtype=int32)
```

```
In [33]: arr2[:-3]
```

```
Out[33]: array([[86, 10, 12, 91, 75, 38, 69, 46, 66, 22],  
                 [36, 67, 55, 73, 32, 98, 14, 40, 45, 56],  
                 [ 9, 29, 22, 61, 34, 20, 7, 6, 58, 82],  
                 [19, 65, 85, 31, 85, 99, 70, 60, 41, 22],  
                 [18, 19, 28, 99, 41, 13, 60, 88, 57, 97],  
                 [50, 24, 15, 31, 87, 69, 16, 82, 70, 70],  
                 [83, 24, 31, 34, 2, 6, 30, 80, 34, 10]], dtype=int32)
```

```
In [9]: arr[0:10:3]
```

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

```
In [34]: arr
```

```
Out[34]: array([0, 1, 2, 3, 4, 5])
```

```
In [10]: arr.max()
```

```
Out[10]: np.int64(5)
```

```
In [35]: arr.min()
```

```
Out[35]: np.int64(0)
```

```
In [11]: arr
```

```
Out[11]: array([0, 1, 2, 3, 4, 5])
```

```
In [12]: arr.mean()
```

```
Out[12]: np.float64(2.5)
```

```
In [36]: arr.median()
```

```
-----  
AttributeError  
Cell In[36], line 1  
----> 1 arr.median()
```

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

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

```
In [37]: from numpy import *
a = array([1,2,3,4,9])
median(a)
```

```
Out[37]: np.float64(3.0)
```

Indexing

```
In [13]: mat = np.arange(0,100).reshape(10,10)
```

```
In [14]: mat
```

```
Out[14]: 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 [15]: row = 4
col = 5
```

```
In [16]: col
```

```
Out[16]: 5
```

```
In [17]: row
```

```
Out[17]: 4
```

```
In [18]: mat
```

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

```
In [19]: mat[row,col]
```

```
Out[19]: np.int64(45)
```

```
In [140]: mat[4,5]
```

```
Out[140]: np.int64(45)
```

```
In [20]: col = 6
```

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

```
Out[21]: np.int64(46)
```

```
In [141]: mat
```

```
Out[141]: 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 [38]: mat[:,]
```

```
Out[38]: array([[ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9],
   [10, 11, 12, 13, 14, 15, 16, 17, 18, 19],
   [20, 21, 22, 23, 24, 25, 26, 27, 28, 29],
   [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
   [40, 41, 42, 43, 44, 45, 46, 47, 48, 49],
   [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 [39]: mat[6]
```

```
Out[39]: array([60, 61, 62, 63, 64, 65, 66, 67, 68, 69])
```

```
In [22]: # with slicing  
mat[:,col] # how to print column in for
```

```
Out[22]: array([ 6, 16, 26, 36, 46, 56, 66, 76, 86, 96])
```

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

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

```
In [41]: mat[row,:]
```

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

```
In [42]: mat
```

```
Out[42]: 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 [43]: mat[:,8]
```

```
Out[43]: array([ 8, 18, 28, 38, 48, 58, 68, 78, 88, 98])
```

```
In [44]: mat
```

```
Out[44]: array([[ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9],
   [10, 11, 12, 13, 14, 15, 16, 17, 18, 19],
   [20, 21, 22, 23, 24, 25, 26, 27, 28, 29],
   [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
   [40, 41, 42, 43, 44, 45, 46, 47, 48, 49],
   [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 [45]: mat[:col]
```

```
Out[45]: 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 [46]: mat[:6]
```

```
Out[46]: 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 [47]: row
```

```
Out[47]: 4
```

```
In [48]: mat
```

```
Out[48]: 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 [49]: mat[:row]
```

```
Out[49]: array([[ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9],
   [10, 11, 12, 13, 14, 15, 16, 17, 18, 19],
   [20, 21, 22, 23, 24, 25, 26, 27, 28, 29],
   [30, 31, 32, 33, 34, 35, 36, 37, 38, 39]])
```

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

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

```
In [51]: mat[row:]
```

```
Out[51]: 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 [52]: mat
```

```
Out[52]: 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 [53]: mat[2:6,2:4]
```

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

```
In [54]: mat
```

```
Out[54]: 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 [55]: mat[1:2,2:4]
```

```
Out[55]: array([[12, 13]])
```

```
In [56]: mat
```

```
Out[56]: 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 [57]: mat[2:3,2:3]
```

```
Out[57]: array([[22]])
```

```
In [58]: mat[3:5,2:4]
```

```
Out[58]: array([[32, 33],  
                 [42, 43]])
```

```
In [59]: mat
```

```
Out[59]: 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 [60]: mat[2:3,4:5]
```

```
Out[60]: array([[24]])
```

Masking

```
In [61]: mat # we also called as filter
```

```
Out[61]: 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 [62]: id(mat)
```

```
Out[62]: 1444021778640
```

```
In [63]: mat
```

```
Out[63]: 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 [64]: mat[mat<50]
```

```
Out[64]: 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 [65]: mat[mat<=50]
```

```
Out[65]: 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 [66]: mat > 50
```

```
Out[66]: array([[False, False, False, False, False, False, False, False,
                  False],
                  [False, False, False, False, False, False, False, False,
                  False],
                  [False, False, False, False, False, False, False, False,
                  False],
                  [False, False, False, False, False, False, False, False,
                  False],
                  [False, False, False, False, False, False, False, False,
                  False],
                  [False, False, False, False, False, False, False, False,
                  False],
                  [False, True, True, True, True, True, True, True, True],
                  [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 [67]: mat[mat==50]
```

```
Out[67]: array([50])
```

```
In [68]: mat
```

```
Out[68]: 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 [69]: mat == 50
```

```
Out[69]: array([[False, False, False, False, False, False, False, False,
   False],
   [False, False, False, False, False, False, False, False, False],
   [False, False, False, False, False, False, False, False, False],
   [False, False, False, False, False, False, False, False, False],
   [False, False, False, False, False, False, False, False, False],
   [False, False, False, False, False, False, False, False, False],
   [False, 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]])
```

```
In [70]: a1 = mat[mat<50]
a1
```

```
Out[70]: 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 [72]: a2 = mat[mat>50]
a2
```

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

```
In [73]: a3 = mat[mat<=50]
a3
```

```
Out[73]: 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 [75]: a4 = mat[mat==50]
a4
```

```
Out[75]: array([50])
```

Array Manipulation Functions

Reshape ()

```
In [6]: a1 = np.array([1,2,3])
reshape = np.reshape(a1,(3,1))
reshape
```

```
Out[6]: array([[1],
 [2],
 [3]])
```

Ravel() --- convert any dimension to 1D array

```
In [8]: # flatten the array
f1 = np.array([[1,2],[3,4]])
flat = np.ravel(f1)
flat
```

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

Transpose() --- transpose rows to columns & vise versa

```
In [9]: f1 = np.array([[1,2],[3,4]])
tran = np.transpose(f1)
tran
```

```
Out[9]: array([[1, 3],
 [2, 4]])
```

vstack() --- vertical stacking

```
In [12]: a = np.array([1,2,3])
b = np.array([3,4,5])
stack = np.vstack([a,b])
stack
```

```
Out[12]: array([[1, 2, 3],
 [3, 4, 5]])
```

hstack() --- horizontal stacking

```
In [13]: a = np.array([1,2])
b = np.array([3,4])
stack = np.hstack([a,b])
stack
```

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

eye() --- to create identical matrix

```
In [14]: f = np.eye(4) #4x4 identity matrix  
f
```

```
Out[14]: array([[1., 0., 0., 0.],  
                 [0., 1., 0., 0.],  
                 [0., 0., 1., 0.],  
                 [0., 0., 0., 1.]])
```

identity() --- the diagonal items will be ones and will be zeros

```
In [15]: np.identity(3)
```

```
Out[15]: array([[1., 0., 0.],  
                 [0., 1., 0.],  
                 [0., 0., 1.]])
```

ndim --- to find number number of dimemsional

```
In [16]: a1 = np.array([[[[1,2],[3,4],[6,8]]]])  
a1
```

```
Out[16]: array([[[[1, 2],  
                  [3, 4],  
                  [6, 8]]]])
```

```
In [17]: a1.ndim
```

```
Out[17]: 4
```

```
In [ ]:
```