In [4]:
```python
import numpy as np
```

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

Out[5]: '2.1.3'

In [1]:
```python
import sys
sys.version
```

Out[1]: '3.13.5 | packaged by Anaconda, Inc. | (main, Jun 12 2025, 16:37:03) [MSC v.192
9 64 bit (AMD64)]'

create list

In [9]:
```python
my_list=[0,1,2,3,4,5,6,7,8]
my_list
```

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

In [6]:
```python
arr=np.array(my_list)
arr
```

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

In [7]:
```python
type(arr)
```

Out[7]: numpy.ndarray

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

Out[10]: list

In [11]:
```python
np.arange(10)
```

Out[11]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

In [12]:
```python
list(range(10))
```

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

In [ ]:

16th

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

In [4]:
```python
np.arange(20)
```

Out[4]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
       17, 18, 19])

```
In [5]:   np.arange(5.0)
```

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

```
In [6]:   np.arange(0,5)
```

```
Out[6]:   array([0, 1, 2, 3, 4])
```

```
In [7]:   np.arange(10,20)
```

```
Out[7]:   array([10, 11, 12, 13, 14, 15, 16, 17, 18, 19])
```

```
In [2]:   import numpy as np
```

```
In [3]:   np.arange(-16,10)
```

```
Out[3]:   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 [4]:   np.arange(-20,-10)
```

```
Out[4]:   array([-20, -19, -18, -17, -16, -15, -14, -13, -12, -11])
```

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

```
Out[5]:   array([], dtype=int64)
```

```
In [6]:   np.arange(10,10)
```

```
Out[6]:   array([], dtype=int64)
```

```
In [11]:  np.arange(10,30,5)
```

```
Out[11]:  array([10, 15, 20, 25])
```

```
In [7]:   np.arange(100,150,3)
```

```
Out[7]:   array([100, 103, 106, 109, 112, 115, 118, 121, 124, 127, 130, 133, 136,
                  139, 142, 145, 148])
```

```
In [13]:  np.zeros(3)#parameter tunning
```

```
Out[13]:  array([0., 0., 0.])
```

```
In [14]:  np.zeros(3,dtype=int)#hyperparameter
```

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

```
In [15]:  np.zeros([2,2])
```

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

```
In [16]:  np.zeros([2,2], dtype=int)
```

```
Out[16]:  array([[0, 0],
                  [0, 0]])
```

```
In [9]:  n=(6,7)
         n1=(6,8)
         print(np.zeros(n1)) #parameter tunning
```

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

```
In [10]:  print(np.zeros(n,dtype=int)) #parameter tunning
```

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

```
In [17]:  np.ones(2)
```

```
Out[17]:  array([1., 1.])
```

```
In [18]:  np.ones([10,4],dtype=int)
```

```
Out[18]:  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]])
```

```
In [ ]:
```

```
In [11]:  np.ones(4)
```

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

```
In [12]:  np.ones(4,dtype=int)
```

```
Out[12]:  array([1, 1, 1, 1])
```

```
In [13]:  np.ones(n)
```

```
Out[13]:  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 [19]: np.twos(2)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[19], line 1
----> 1 np.twos(2)

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 [20]: rand(3,2)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[20], line 1
----> 1 rand(3,2)

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

```
In [21]: random.rand(3,2)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[21], line 1
----> 1 random.rand(3,2)

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

```
In [22]: np.random.rand(3,4)
```

```
Out[22]: array([[0.44603127, 0.4118301 , 0.35000076, 0.60068486],
                [0.43327577, 0.35285088, 0.32799258, 0.75968936],
                [0.27047181, 0.20915858, 0.94905864, 0.58333891]])
```

```
In [23]: np.random.randint(2)
```

```
Out[23]: 0
```

```
In [24]: np.random.randint(3)
```

```
Out[24]: 2
```

```
In [25]: np.random.randint(1,8)
```

```
Out[25]: 3
```

```
In [26]: np.random.randint(1,8,100)
```

```
Out[26]: array([2, 5, 7, 2, 3, 6, 3, 1, 7, 5, 4, 3, 2, 6, 5, 4, 4, 3, 7, 2, 7, 7,
                3, 7, 4, 2, 5, 6, 2, 6, 3, 7, 2, 6, 5, 5, 1, 3, 5, 1, 5, 2, 3, 5,
                1, 3, 3, 7, 2, 2, 2, 2, 6, 1, 3, 6, 2, 7, 7, 3, 2, 6, 2, 3, 1, 2,
                3, 1, 3, 5, 1, 4, 6, 1, 7, 1, 1, 2, 2, 7, 7, 4, 6, 3, 4, 1, 4, 5,
                1, 1, 2, 2, 1, 4, 2, 7, 7, 3, 2, 1], dtype=int32)
```

```
In [27]: np.random.randint(10,40,(10,10))
```

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

```
In [29]: np.arange(1,13).reshape(3,4)
```

```
Out[29]: array([[ 1,  2,  3,  4],
                [ 5,  6,  7,  8],
                [ 9, 10, 11, 12]])
```

```
In [30]: np.arange(1,13).reshape(3,5)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[30], line 1
----> 1 np.arange(1,13).reshape(3,5)

ValueError: cannot reshape array of size 12 into shape (3,5)
```

```
In [31]: np.arange(1,16).reshape(3,5)
```

```
Out[31]: array([[ 1,  2,  3,  4,  5],
                [ 6,  7,  8,  9, 10],
                [11, 12, 13, 14, 15]])
```

```
In [ ]:
```

Matrics slicing

```
In [33]: b=np.random.randint(10,20,(5,4))
         b
```

```
Out[33]: array([[13, 16, 17, 11],
                [13, 18, 13, 19],
                [16, 16, 15, 16],
                [10, 18, 18, 10],
                [19, 12, 19, 19]], dtype=int32)
```

```
In [34]: b[:]
```

```
Out[34]: array([[13, 16, 17, 11],
                [13, 18, 13, 19],
                [16, 16, 15, 16],
                [10, 18, 18, 10],
                [19, 12, 19, 19]], dtype=int32)
```

```
In [36]: b[0:5]
```

```
Out[36]:  array([[13, 16, 17, 11],
                 [13, 18, 13, 19],
                 [16, 16, 15, 16],
                 [10, 18, 18, 10],
                 [19, 12, 19, 19]], dtype=int32)
```

In [37]: `b[0]`

```
Out[37]:  array([13, 16, 17, 11], dtype=int32)
```

In [39]: `b[0:4]`

```
Out[39]:  array([[13, 16, 17, 11],
                 [13, 18, 13, 19],
                 [16, 16, 15, 16],
                 [10, 18, 18, 10]], dtype=int32)
```

In [40]: `b[0,1]`

```
Out[40]:  np.int32(16)
```

In [41]: `b[::-1]`

```
Out[41]:  array([[19, 12, 19, 19],
                 [10, 18, 18, 10],
                 [16, 16, 15, 16],
                 [13, 18, 13, 19],
                 [13, 16, 17, 11]], dtype=int32)
```

In [42]: `b[::-2]`

```
Out[42]:  array([[19, 12, 19, 19],
                 [16, 16, 15, 16],
                 [13, 16, 17, 11]], dtype=int32)
```

In [45]: `b[:]`

```
Out[45]:  array([[13, 16, 17, 11],
                 [13, 18, 13, 19],
                 [16, 16, 15, 16],
                 [10, 18, 18, 10],
                 [19, 12, 19, 19]], dtype=int32)
```

In [46]: `print(b[:])`

```
[[13 16 17 11]
 [13 18 13 19]
 [16 16 15 16]
 [10 18 18 10]
 [19 12 19 19]]
```

In [47]: `b[-3,-3]`

```
Out[47]:  np.int32(16)
```

In [48]: `b.max()`

```
Out[48]:  np.int32(19)
```

```
In [49]: b.min()
```

```
Out[49]: np.int32(10)
```

how to find colouns

```
In [14]: b
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[14], line 1
----> 1 b

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

```
In [15]: b=np.random.randint(10,20,(5,4))
```

```
In [16]: b
```

```
Out[16]: array([[17, 11, 17, 14],
                [13, 13, 19, 19],
                [16, 14, 11, 12],
                [13, 14, 10, 19],
                [14, 13, 14, 19]], dtype=int32)
```

```
In [17]: b[:3]
```

```
Out[17]: array([[17, 11, 17, 14],
                [13, 13, 19, 19],
                [16, 14, 11, 12]], dtype=int32)
```

```
In [18]: b[3:]
```

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

```
In [19]: b[:,3]
```

```
Out[19]: array([14, 19, 12, 19, 19], dtype=int32)
```

```
In [20]: b[:,0]
```

```
Out[20]: array([17, 13, 16, 13, 14], dtype=int32)
```

```
In [23]: b[:,2]
```

```
Out[23]: array([17, 19, 11, 10, 14], dtype=int32)
```

```
In [ ]:
```

print a specific portion of Array

```
In [24]: b
```

```
Out[24]:  array([[17, 11, 17, 14],
                 [13, 13, 19, 19],
                 [16, 14, 11, 12],
                 [13, 14, 10, 19],
                 [14, 13, 14, 19]], dtype=int32)
```

In [25]: `b[2:3,0:2]`

```
Out[25]:  array([[16, 14]], dtype=int32)
```

In [26]: `b[1:4,1:4]`

```
Out[26]:  array([[13, 19, 19],
                 [14, 11, 12],
                 [14, 10, 19]], dtype=int32)
```

In [27]: `b[0:3,0:3]`

```
Out[27]:  array([[17, 11, 17],
                 [13, 13, 19],
                 [16, 14, 11]], dtype=int32)
```

In [ ]:

In [28]: `mat=np.arange(0,100).reshape(10,10)`

In [29]: `mat`

```
Out[29]:  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 [30]: `mat>50`

```
Out[30]: 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]])
```

using fliter(masking)

```
In [33]: mat[mat>50]
```

```
Out[33]: 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 [34]: mat[mat!=50]
```

```
Out[34]: 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, 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 [35]: mat[mat<50]
```

```
Out[35]: 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 [36]: mat[mat==50]
```

```
Out[36]: array([50])
```

```
In [ ]:
```

```
In [39]: s=np.arange(0,25).reshape(5,5)
         s
```

Out[39]: 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]])

In [40]: ```
          s[s<5]
          ```

Out[40]:  array([0, 1, 2, 3, 4])

In [41]:  ```
          s[s>5]
          ```

Out[41]: array([6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22,
                 23, 24])

In [ ]: