Numpy Crash Course

In [1]:
```
import numpy as np
```

In [2]:
```
import sys
sys.version
```

Out[2]:
```
'3.12.7 | packaged by Anaconda, Inc. | (main, Oct  4 2024, 13:17:27) [MSC v.1929 6
4 bit (AMD64)]'
```

In [3]:
```
np.__version__
```

Out[3]:
```
'1.26.4'
```

create a list

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

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

In [5]:
```
type(my_list)
```

Out[5]:
```
list
```

converting list to array

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

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

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

Out[7]:
```
numpy.ndarray
```

In [8]:
```
print(type(arr))
print(type(my_list))
```

```
<class 'numpy.ndarray'>
<class 'list'>
```

arange() on numpy---> it accepts atmost 3 arguments

In [9]:
```
np.arange(10)
```

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

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

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

```
In [11]:  np.arange(10,20,5)
```

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

```
In [12]:  np.arange(10,30,3)
```

```
Out[12]:  array([10, 13, 16, 19, 22, 25, 28])
```

```
In [13]:  np.arange(8,20)
```

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

```
In [14]:  np.arange(20,8)
```

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

```
In [15]:  np.arange(-20,8)  # 1st Argument < 2nd Argument
```

```
Out[15]:  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])
```

```
In [16]:  n=np.arange(-10,8)
          n
```

```
Out[16]:  array([-10,  -9,  -8,  -7,  -6,  -5,  -4,  -3,  -2,  -1,   0,   1,   2,
                   3,   4,   5,   6,   7])
```

zeros() on numpy

```
In [17]:  np.zeros(3) # by default it gives float
```

```
Out[17]:  array([0., 0., 0.])
```

```
In [18]:  np.zeros(3, dtype=int) # to get int type zeros
```

```
Out[18]:  array([0, 0, 0])
```

```
In [19]:  z=np.zeros(5)
          z
```

```
Out[19]:  array([0., 0., 0., 0., 0.])
```

```
In [20]:  np.zeros((2,2))
```

Out[20]:
```
array([[0., 0.],
       [0., 0.]])
```

In [21]:
```
np.zeros((2,2),dtype=int)
```

Out[21]:
```
array([[0, 0],
       [0, 0]])
```

In [22]:
```
np.zeros((3,3),dtype=int)
```

Out[22]:
```
array([[0, 0, 0],
       [0, 0, 0],
       [0, 0, 0]])
```

In [23]:
```
nd=np.zeros((5,9),dtype=int)
nd
```

Out[23]:
```
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]])
```

In [24]:
```
len(nd)
```

Out[24]:
```
5
```

one() in NUMPY

In [25]:
```
np.ones(3)
```

Out[25]:
```
array([1., 1., 1.])
```

In [26]:
```
np.ones(3,dtype=int)
```

Out[26]:
```
array([1, 1, 1])
```

In [27]:
```
nd1=np.ones((10,10),dtype=int)
nd1
```

Out[27]:
```
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],
       [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]])
```

rand()

```
In [28]: rand(3,2)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[28], line 1
----> 1 rand(3,2)

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

```
In [30]: np.random.rand(2) # here, np-->package,random-->module, rand()-->function

Out[30]: array([0.00425928, 0.96522183])
```

```
In [33]: np.random.rand(3)

Out[33]: array([0.66128979, 0.44730702, 0.58286601])
```

```
In [34]: np.random.rand(2,3)

Out[34]: array([[0.91989137, 0.67074746, 0.08561321],
               [0.27442912, 0.5086701 , 0.35805054]])
```

randint()

```
In [38]: np.random.randint(3) ## generate one random number from 0 to <3

Out[38]: 1
```

```
In [40]: np.random.randint(2,10)  # generate one random number b/w 2 to 9

Out[40]: 3
```

```
In [41]: np.random.randint(2,10,3)  # generate 3 random numbers b/w 2 to 9

Out[41]: array([9, 4, 5])
```

```
In [42]: np.random.randint(2,10,5)

Out[42]: array([5, 8, 6, 8, 7])
```

```
In [43]: np.random.randint(-30,20,10)

Out[43]: array([-26,  16, -21, -23, -23, -18,  15,  19,  14, -23])
```

```
In [44]: np.random.randint(10,20,(10,10)) # generates 10x10 matrix between numbers 10 and 10
```

```
Out[44]:  array([[18, 18, 11, 12, 18, 16, 16, 10, 16, 10],
                 [14, 11, 13, 17, 16, 14, 10, 11, 19, 12],
                 [16, 16, 13, 11, 10, 17, 10, 18, 18, 16],
                 [18, 10, 17, 13, 17, 15, 16, 14, 19, 13],
                 [17, 13, 15, 11, 13, 12, 12, 19, 18, 18],
                 [13, 10, 14, 10, 14, 10, 14, 19, 16, 16],
                 [14, 13, 10, 15, 12, 18, 19, 18, 13, 12],
                 [19, 16, 16, 15, 15, 11, 12, 15, 17, 13],
                 [19, 13, 13, 12, 18, 19, 16, 18, 12, 19],
                 [18, 15, 16, 18, 18, 17, 12, 18, 17, 15]])
```

```
In [45]:  np.random.randint(10,20,(3,3))
```

```
Out[45]:  array([[18, 17, 12],
                 [15, 11, 13],
                 [19, 13, 15]])
```

```
In [46]:  m= np.random.randint(10,20,(10,10))
           m
```

```
Out[46]:  array([[18, 12, 12, 12, 19, 12, 14, 19, 15, 18],
                 [15, 11, 11, 14, 14, 17, 17, 11, 10, 15],
                 [10, 13, 19, 10, 13, 10, 11, 12, 10, 15],
                 [11, 13, 15, 11, 16, 16, 11, 13, 17, 11],
                 [15, 16, 17, 10, 11, 16, 19, 13, 16, 12],
                 [18, 17, 16, 14, 17, 19, 18, 10, 11, 16],
                 [11, 10, 17, 17, 13, 12, 11, 17, 15, 14],
                 [18, 19, 12, 15, 13, 14, 17, 15, 11, 17],
                 [16, 12, 15, 15, 12, 18, 12, 14, 14, 15],
                 [11, 12, 10, 11, 17, 12, 15, 12, 19, 18]])
```

reshape()

```
In [47]:  arr
```

```
Out[47]:  array([0, 1, 2, 3, 4, 5])
```

```
In [48]:  arr.reshape(2,3)
```

```
Out[48]:  array([[0, 1, 2],
                 [3, 4, 5]])
```

```
In [50]:  arr.reshape(3,3)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[50], line 1
----> 1 arr.reshape(3,3)

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

```
In [51]:  arr.reshape(6,1)
```

```
Out[51]: array([[0],
                [1],
                [2],
                [3],
                [4],
                [5]])
```

```
In [52]: arr.reshape(1,6)
```

```
Out[52]: array([[0, 1, 2, 3, 4, 5]])
```

Slicing in matrix

```
In [53]: m
```

```
Out[53]: array([[18, 12, 12, 12, 19, 12, 14, 19, 15, 18],
                [15, 11, 11, 14, 14, 17, 17, 11, 10, 15],
                [10, 13, 19, 10, 13, 10, 11, 12, 10, 15],
                [11, 13, 15, 11, 16, 16, 11, 13, 17, 11],
                [15, 16, 17, 10, 11, 16, 19, 13, 16, 12],
                [18, 17, 16, 14, 17, 19, 18, 10, 11, 16],
                [11, 10, 17, 17, 13, 12, 11, 17, 15, 14],
                [18, 19, 12, 15, 13, 14, 17, 15, 11, 17],
                [16, 12, 15, 15, 12, 18, 12, 14, 14, 15],
                [11, 12, 10, 11, 17, 12, 15, 12, 19, 18]])
```

```
In [54]: b=np.random.randint(10,20,(5,4))
```

```
In [55]: b
```

```
Out[55]: array([[16, 19, 13, 15],
                [17, 18, 18, 14],
                [10, 14, 18, 12],
                [14, 15, 16, 11],
                [13, 12, 12, 11]])
```

```
In [56]: b[:]
```

```
Out[56]: array([[16, 19, 13, 15],
                [17, 18, 18, 14],
                [10, 14, 18, 12],
                [14, 15, 16, 11],
                [13, 12, 12, 11]])
```

```
In [57]: b[1:4]
```

```
Out[57]: array([[17, 18, 18, 14],
                [10, 14, 18, 12],
                [14, 15, 16, 11]])
```

```
In [59]: b[-1:]    # it prints last row
```

```
Out[59]: array([[13, 12, 12, 11]])
```

```
In [62]: b[:-1]   # starting to -2(-1-1)
```

```
Out[62]: array([[16, 19, 13, 15],
                [17, 18, 18, 14],
                [10, 14, 18, 12],
                [14, 15, 16, 11]])
```

```
In [61]: b
```

```
Out[61]: array([[16, 19, 13, 15],
                [17, 18, 18, 14],
                [10, 14, 18, 12],
                [14, 15, 16, 11],
                [13, 12, 12, 11]])
```

```
In [64]: b[:-2]   #staring to -3(-2-1) row
```

```
Out[64]: array([[16, 19, 13, 15],
                [17, 18, 18, 14],
                [10, 14, 18, 12]])
```

```
In [66]: b[1:4]    # row1 to row3(4-1)
```

```
Out[66]: array([[17, 18, 18, 14],
                [10, 14, 18, 12],
                [14, 15, 16, 11]])
```

```
In [68]: b[1,3]  # sencond row fourth column
```

```
Out[68]: 14
```

```
In [69]: b
```

```
Out[69]: array([[16, 19, 13, 15],
                [17, 18, 18, 14],
                [10, 14, 18, 12],
                [14, 15, 16, 11],
                [13, 12, 12, 11]])
```

```
In [70]: b[1,-1]
```

```
Out[70]: 14
```

numpy operations

```
In [71]: arr
```

```
Out[71]: array([0, 1, 2, 3, 4, 5])
```

max()

In [72]:
```
arr.max()
```

Out[72]:
```
5
```

min()

In [73]:
```
arr.min()
```

Out[73]:
```
0
```

median()

In [75]:
```
from numpy import * # it imports all functions of numpy internally
a=array([1,2,3,4,9])
median(a)
```

Out[75]:
```
3.0
```

indexing in numpy

In [76]:
```
mat=np.arange(0,100)
mat
```

Out[76]:
```
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 [79]:
```
mat=mat.reshape(10,10)
mat
```

Out[79]:
```
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 [81]:
```
mat[4,6]
```

Out[81]:
```
46
```

In [82]:
```
mat[2]  # displays row 2
```

```
Out[82]: array([20, 21, 22, 23, 24, 25, 26, 27, 28, 29])
```

```
In [83]: mat[:,6]  # displays 6th colum values
```

```
Out[83]: array([ 6, 16, 26, 36, 46, 56, 66, 76, 86, 96])
```

```
In [84]: mat[::-1]
```

```
Out[84]: 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 [85]: mat[::-2]
```

```
Out[85]: array([[90, 91, 92, 93, 94, 95, 96, 97, 98, 99],
                [70, 71, 72, 73, 74, 75, 76, 77, 78, 79],
                [50, 51, 52, 53, 54, 55, 56, 57, 58, 59],
                [30, 31, 32, 33, 34, 35, 36, 37, 38, 39],
                [10, 11, 12, 13, 14, 15, 16, 17, 18, 19]])
```

```
In [87]: mat[2:6,2:4]   # displays part of matrix
```

```
Out[87]: array([[22, 23],
                [32, 33],
                [42, 43],
                [52, 53]])
```

```
In [88]: mat[1:2,2:4]
```

```
Out[88]: array([[12, 13]])
```

```
In [89]: mat[3:5,2:4]
```

```
Out[89]: array([[32, 33],
                [42, 43]])
```

masking or filter

```
In [90]: mat
```

```
Out[90]: 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 [91]: mat>50
```

```
Out[91]: 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 [92]: mat[mat>50]
```

```
Out[92]: 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 [93]: mat[mat==50]
```

```
Out[93]: array([50])
```

```
In [94]: mat[mat<=50]
```

```
Out[94]: 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 [95]: mat[mat!=50]
```

Out[95]:
```
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 [96]:
```
range(10)
```

Out[96]:
```
range(0, 10)
```

In [97]:
```
for i in range(10):
    print(i)
```

```
0
1
2
3
4
5
6
7
8
9
```

In [98]:
```
list(range(6))
```

Out[98]:
```
[0, 1, 2, 3, 4, 5]
```

In [99]:
```
arr
```

Out[99]:
```
array([0, 1, 2, 3, 4, 5])
```

In [100…
```
arr.reshape(3,2)
```

Out[100…
```
array([[0, 1],
       [2, 3],
       [4, 5]])
```

In [101…
```
arr.reshape(1,6)
```

Out[101…
```
array([[0, 1, 2, 3, 4, 5]])
```

In [102…
```
arr.reshape(6,1)
```

Out[102…
```
array([[0],
       [1],
       [2],
       [3],
       [4],
       [5]])
```

In [103…
```
arr
```

```
Out[103…  array([0, 1, 2, 3, 4, 5])
```

```
In [104…  arr.reshape(2,4)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[104], line 1
----> 1 arr.reshape(2,4)

ValueError: cannot reshape array of size 6 into shape (2,4)
```

```
In [106…  arr.reshape(2,3,order='C')
```

```
Out[106…  array([[0, 1, 2],
                 [3, 4, 5]])
```

```
In [107…  arr.reshape(2,3,order='F')
```

```
Out[107…  array([[0, 2, 4],
                 [1, 3, 5]])
```

```
In [109…  arr.reshape(2,3,order='A')
```

```
Out[109…  array([[0, 1, 2],
                 [3, 4, 5]])
```

```
In [111…  arr
```

```
Out[111…  array([0, 1, 2, 3, 4, 5])
```

```
In [112…  arr.reshape(3,3)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[112], line 1
----> 1 arr.reshape(3,3)

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

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```