Numpy crash course

```
import numpy as np
```

```
np.__version__
```

```
'1.26.4'
```

```
import sys
sys.version
```

```
'3.12.4 | packaged by Anaconda, Inc. | (main, Jun 18 2024, 10:07:17)
[Clang 14.0.6 ]'
```

Creating Arrays

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

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

```
type(my_list)
```

```
list
```

```
arr = np.array(my_list)
arr
```

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

```
type(arr)
```

```
numpy.ndarray
```

```
type(my_list)
```

```
list
```

Functions

```
np.arange(15)
```

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

```
np.arange(3.0)
```

```
array([0., 1., 2.])
```

```
np.arange(10)
```

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

```
np.arange(0,5)
```

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

```
np.arange(10,20)
```

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

```
np.arange(20,10)     #1st arg should always< 2nd arg
```

```
array([], dtype=int64)
```

```
np.arange(-20,10)
```

```
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])
```

```
np.arange(-16,10)
```

```
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])
```

```
np.arange(-20,-10)
```

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

```
np.arange(30,20)
```

```
array([], dtype=int64)
```

```
ar = np.arange(-30,20)
ar
```

```
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])
```

```
np.arange(10,10)
```

```
array([], dtype=int64)
```

```
np.arange()
```

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

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

```
np.arange(10,30,5) # start,stop,step
```

```
array([10, 15, 20, 25])
```

```
np.arange(0,10,3)
```

```
array([0, 3, 6, 9])
```

```
np.arange(10,30,5,6)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[62], line 1
----> 1 np.arange(10,30,5,6)

TypeError: Cannot interpret '6' as a data type
```

```
np.zeros(5)    # by default it will print the zeroes in float format.
# Parameter tunning
```

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

```
np.zeros(5, dtype=int)  # hyper parameter
```

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

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

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

```
np.zeros((2,2), dtype=int)
```

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

```
zero = np.zeros([2,2])
```

```
type(zero)
```

```
numpy.ndarray
```

```
print(zero)
print(type(zero))
```

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

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

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

```
n=(6,7)    # tuple
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.]]
```

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

```
n=(6,7)    # tuple
n1 = (6,8)
print(np.zeros(n1, 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 0 0 0 0 0 0]]
```

```
n
```

```
(6, 7)
```

```
n1
```

```
(6, 8)
```

```
np.ones(3)
```

```
array([1., 1., 1.])
```

```
np.ones(3,dtype=int)
```

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

```
np.ones(4,dtype = int)
```

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

```
np.ones(n)
```

```
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.]])
```

```
np.ones((4,5),dtype=int)
```

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

```
np.twos((3,2), dtype=int) # system understand only ones and zeroes
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[123], line 1
----> 1 np.twos((3,2), dtype=int)

File /opt/anaconda3/lib/python3.12/site-packages/numpy/__init__.py:333, in __getattr__(attr)
    330     "Removed in NumPy 1.25.0"
    331     raise RuntimeError("Tester was removed in NumPy 1.25.")
--> 333 raise AttributeError("module {!r} has no attribute "
    334                      "{!r}".format(__name__, attr))

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

```
range(5)
```

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

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

```
r
```

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

```
for i in (r):    #i - iteration
    print(i)
```

```
0
1
2
3
4
```

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

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

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

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

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

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

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

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

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

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

```
rand(3,2)     # rand - random number
```

```
---------------------------------------------------------------------------
-----
NameError                                 Traceback (most recent call
last)
Cell In[148], line 1
----> 1 rand(3,2)

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

```
rand(3,2)
random.rand(3,2)     # random - module name
```

```
---------------------------------------------------------------------------
-----
```

```
NameError                                 Traceback (most recent call
last)
Cell In[150], line 1
----> 1 rand(3,2)
      2 random.rand(3,2)

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

```
np.random.rand(5)     # package.module name. function name
```

```
array([0.32753623, 0.78475799, 0.84826847, 0.88727188, 0.53508119])
```

```
np.rand(4)    # module is missing
```

```
---------------------------------------------------------------------
-----
AttributeError                            Traceback (most recent call
last)
Cell In[154], line 1
----> 1 np.rand(4)

File
/opt/anaconda3/lib/python3.12/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 'rand'
```

```
np.random.rand(2,3)   # prints 2 rows and 3 columns with random
numbers, by default with float datatype
```

```
array([[0.4909457 , 0.50971826, 0.95401961],
       [0.09722143, 0.52034218, 0.63950499]])
```

```
np.random.randint(2,5)   # return random numbers including the lower
value and excluding the higher value
                         # shift+tab for info
```

```
2
```

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

```
16
```

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

```
0
```

```
np.random.randint(10,20,3)
```

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

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

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

np.random.rand(3)

array([0.81662337, 0.96175485, 0.25829767])

np.random.randint(30,20,10)   # first argument should be less than the
2nd argument

---------------------------------------------------------------------------
-----
ValueError                                Traceback (most recent call
last)
Cell In[210], line 1
----> 1 np.random.randint(30,20,10)

File numpy/random/mtrand.pyx:782, in
numpy.random.mtrand.RandomState.randint()

File numpy/random/_bounded_integers.pyx:1334, in
numpy.random._bounded_integers._rand_int64()

ValueError: low >= high

np.random.randint(20,30,3)

array([22, 21, 26])

np.random.randint(1,12,10)

array([ 4, 10,  1,  9,  7, 10,  1,  8,  8,  6])

np.random.randint(10,40,(10,10))      # start=10, stop=40, step =
10*10 matrix

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

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

```
array([[11, 16, 12, 16],
       [17, 17, 17, 10],
       [10, 13, 11, 16],
       [13, 11, 16, 13],
       [18, 15, 11, 18]])
```

```
type(b)
```

```
numpy.ndarray
```

```
b[:]
```

```
array([[11, 16, 12, 16],
       [17, 17, 17, 10],
       [10, 13, 11, 16],
       [13, 11, 16, 13],
       [18, 15, 11, 18]])
```

```
b[1:3]    # 3=3-1=2, so row no 1 & 2 will print
```

```
array([[17, 17, 17, 10],
       [10, 13, 11, 16]])
```

```
b
```

```
array([[11, 16, 12, 16],
       [17, 17, 17, 10],
       [10, 13, 11, 16],
       [13, 11, 16, 13],
       [18, 15, 11, 18]])
```

```
b[2,3]
```

```
16
```

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

```
16
```

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

```
17
```

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

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

Operations

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

```
array([12, 12, 18, 11, 15])
```

```
id(a)
```

```
4435107952
```

```
arr
```

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

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

```
arr2
```

```
array([[33, 84, 23, 43, 24, 61, 41, 23, 81, 51],
       [19, 12, 83, 37, 61, 34, 50, 15,  2, 16],
       [23, 74,  8, 42, 71, 45, 30, 96, 24, 54],
       [30, 28, 47, 67, 31, 44, 97, 95, 90, 34],
       [44, 89, 69, 94,  9, 86, 46, 69, 30, 20],
       [65, 77, 75,  6, 53, 77, 15, 42, 20, 15],
       [80, 44,  1, 67, 67, 59, 67, 10, 49, 45],
       [ 1, 27, 26, 37, 22, 78, 79, 43, 94, 53],
       [35,  1, 22, 36, 27, 34, 41, 34, 91, 98],
       [29, 37, 25, 78, 98, 83, 63,  5,  2, 59]])
```

```
arr
```

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

```
arr[:]
```

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

```
arr[:4]
```

```
array([0, 1, 2, 3])
```

```
arr2[:]
```

```
array([[33, 84, 23, 43, 24, 61, 41, 23, 81, 51],
       [19, 12, 83, 37, 61, 34, 50, 15,  2, 16],
       [23, 74,  8, 42, 71, 45, 30, 96, 24, 54],
       [30, 28, 47, 67, 31, 44, 97, 95, 90, 34],
       [44, 89, 69, 94,  9, 86, 46, 69, 30, 20],
       [65, 77, 75,  6, 53, 77, 15, 42, 20, 15],
       [80, 44,  1, 67, 67, 59, 67, 10, 49, 45],
       [ 1, 27, 26, 37, 22, 78, 79, 43, 94, 53],
```

```
       [35,  1, 22, 36, 27, 34, 41, 34, 91, 98],
       [29, 37, 25, 78, 98, 83, 63,  5,  2, 59]])
```

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

```
array([[33, 84, 23, 43, 24, 61, 41, 23, 81, 51],
       [19, 12, 83, 37, 61, 34, 50, 15,  2, 16],
       [23, 74,  8, 42, 71, 45, 30, 96, 24, 54],
       [30, 28, 47, 67, 31, 44, 97, 95, 90, 34],
       [44, 89, 69, 94,  9, 86, 46, 69, 30, 20]])
```

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

```
61
```

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

```
2
```

```
arr2
```

```
array([[33, 84, 23, 43, 24, 61, 41, 23, 81, 51],
       [19, 12, 83, 37, 61, 34, 50, 15,  2, 16],
       [23, 74,  8, 42, 71, 45, 30, 96, 24, 54],
       [30, 28, 47, 67, 31, 44, 97, 95, 90, 34],
       [44, 89, 69, 94,  9, 86, 46, 69, 30, 20],
       [65, 77, 75,  6, 53, 77, 15, 42, 20, 15],
       [80, 44,  1, 67, 67, 59, 67, 10, 49, 45],
       [ 1, 27, 26, 37, 22, 78, 79, 43, 94, 53],
       [35,  1, 22, 36, 27, 34, 41, 34, 91, 98],
       [29, 37, 25, 78, 98, 83, 63,  5,  2, 59]])
```

```
arr2[::-2]
```

```
array([[29, 37, 25, 78, 98, 83, 63,  5,  2, 59],
       [ 1, 27, 26, 37, 22, 78, 79, 43, 94, 53],
       [65, 77, 75,  6, 53, 77, 15, 42, 20, 15],
       [30, 28, 47, 67, 31, 44, 97, 95, 90, 34],
       [19, 12, 83, 37, 61, 34, 50, 15,  2, 16]])
```

```
arr2[::2]
```

```
array([[33, 84, 23, 43, 24, 61, 41, 23, 81, 51],
       [23, 74,  8, 42, 71, 45, 30, 96, 24, 54],
       [44, 89, 69, 94,  9, 86, 46, 69, 30, 20],
       [80, 44,  1, 67, 67, 59, 67, 10, 49, 45],
       [35,  1, 22, 36, 27, 34, 41, 34, 91, 98]])
```

```
arr2
```

```
array([[33, 84, 23, 43, 24, 61, 41, 23, 81, 51],
       [19, 12, 83, 37, 61, 34, 50, 15,  2, 16],
```

```
       [23, 74,  8, 42, 71, 45, 30, 96, 24, 54],
       [30, 28, 47, 67, 31, 44, 97, 95, 90, 34],
       [44, 89, 69, 94,  9, 86, 46, 69, 30, 20],
       [65, 77, 75,  6, 53, 77, 15, 42, 20, 15],
       [80, 44,  1, 67, 67, 59, 67, 10, 49, 45],
       [ 1, 27, 26, 37, 22, 78, 79, 43, 94, 53],
       [35,  1, 22, 36, 27, 34, 41, 34, 91, 98],
       [29, 37, 25, 78, 98, 83, 63,  5,  2, 59]])
```

```
arr2[:-3]   # -3-1=-4
```

```
array([[33, 84, 23, 43, 24, 61, 41, 23, 81, 51],
       [19, 12, 83, 37, 61, 34, 50, 15,  2, 16],
       [23, 74,  8, 42, 71, 45, 30, 96, 24, 54],
       [30, 28, 47, 67, 31, 44, 97, 95, 90, 34],
       [44, 89, 69, 94,  9, 86, 46, 69, 30, 20],
       [65, 77, 75,  6, 53, 77, 15, 42, 20, 15],
       [80, 44,  1, 67, 67, 59, 67, 10, 49, 45]])
```

```
arr
```

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

```
arr.max()
```

```
5
```

```
arr.min()
```

```
0
```

```
arr.mean()
```

```
2.5
```

```
arr.median()
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[295], line 1
----> 1 arr.median()

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

```
from numpy import *
a=([1,2,4,4,6])
median(a)
```

```
4.0
```

```
arr
```

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

arr.reshape(3,2)       # 3*2=6 elements

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

arr.reshape(6,1)      # 6*1=6 elements

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

arr.reshape(1,6)         # 1*6=6 elements

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

arr.reshape(2,4)     # 2*4=8 elements but arr have only 6 elements

---------------------------------------------------------------------------
-----
ValueError                                Traceback (most recent call
last)
Cell In[311], line 1
----> 1 arr.reshape(2,4)

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

arr

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

arr.reshape(2,3,order='c')

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

arr.reshape(2,3,order='F')    # Fortran

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

arr.reshape(2,3,order='A')

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

indexing

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

```
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]])
```

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

```
row
```

```
4
```

```
col
```

```
5
```

```
mat
```

```
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]])
```

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

```
45
```

```
mat[:]
```

```
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]])
```

```
mat[6]   # refers to row no by default, specific row
```

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

```
mat[:,6]  # to print specific column with slices
```

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

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

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

```
mat[:,8]   # n=n-1 formula not applicable here, formula applies only
in slicing, here we are using ,
```

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

```
mat[:col]
```

```
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]])
```

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

```
array([ 5, 15, 25, 35, 45, 55, 65, 75, 85, 95])
```

```
mat
```

```
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]])
```

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

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

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

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

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

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

Masking

```
mat
```

```
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]])
```

```
id(mat)
```

```
4444644976
```

```
mat<50
```

```
array([[ True,  True,  True,  True,  True,  True,  True,  True,  True,
         True],
       [ True,  True,  True,  True,  True,  True,  True,  True,  True,
         True],
       [ True,  True,  True,  True,  True,  True,  True,  True,  True,
         True],
       [ True,  True,  True,  True,  True,  True,  True,  True,  True,
         True],
       [ True,  True,  True,  True,  True,  True,  True,  True,  True,
         True],
       [False, False, False, False, False, False, False, False, False,
        False],
       [False, False, False, False, False, False, False, False, False,
        False],
       [False, False, False, False, False, False, False, False, False,
        False],
       [False, False, False, False, False, False, False, False, False,
        False],
       [False, False, False, False, False, False, False, False, False,
        False]])
```

```
mat[mat<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])
```

```
mat[mat<=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])
```

```
mat[mat>50]
```

```
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])
```

```
mat[mat>=50]
```

```
array([50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65,
66,
       67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82,
83,
       84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98,
99])
```

```
mat[mat!=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,
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])
```

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

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

```
mat==50
```

```
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]])
```