

```
# Numpy
```

```
import numpy as np
```

```
#Numpy Arrays, 0D
```

```
a = 45
print (a)
print (type(a))
```

```
↵ 45
   <class 'int'>
```

```
a = 45
arr_0 = np.array(a)
print(arr_0)
print(type(arr_0))
```

```
↵ 45
   <class 'numpy.ndarray'>
```

```
arr_0.ndim
```

```
↵ 0
```

```
arr_0.shape
```

```
↵ ()
```

```
# 1-D
```

```
a = [1,2,3]
arr_1 = np.array(a)
print(arr_1)
```

```
↵ [1 2 3]
```

```
arr_1.ndim
```

```
↵ 1
```

```
arr_1.shape
```

```
↵ (3,)
```

```
# 2D
```

```
a = [[1,2,3]]
arr_2 = np.array(a)
print(arr_2)
```

```
↵ [[1 2 3]]
```

```
arr_2.ndim
```

```
↵ 2
```

```
arr_2.shape
```

```
↵ (1, 3)
```

```
# Functions for creating 1D array
```

```
# arrange
a = np.arange(10)
```

```
print(a)
```

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

```
# linspace  
np.linspace(1,5,5)
```

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

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

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

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

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

```
#random.randint  
np.random.randint(1,1000,5)
```

```
array([511, 778, 134, 56, 238])
```

```
# Accessing elements
```

```
a = np.arange (10,20,5)  
print(a)
```

```
[10 15]
```

```
# Accessing a single value through indexing
```

```
#positive indexing  
a[1]
```

```
np.int64(15)
```

```
a[-1]
```

```
np.int64(15)
```

```
#Accessing a multiple values based on slicing
```

```
a[1:5:2]
```

```
array([15])
```

```
a[-1:-3: -1]
```

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

```
a[[0,1]]
```

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

```
a=np.array([45,46,47])
```

```
a[a%2==0]
```

```
array([46])
```

```
# Modify the values in Numpy
```

```
import numpy as np
```

```
a = np.array([45,46,47])
```

```
a1 = np.append(a,110)
print
print(a1)
```

```
↔ [ 45  46  47 110]
```

```
a = np.array([45,46,47])
a2 = np.append(a,[140,230,233])
print(a2)
```

```
↔ [ 45  46  47 140 230 233]
```

```
a [0] = 150
print(a)
```

```
↔ [150  46  47]
```

```
a[[0,1]] = [100,200]
print(a)
```

```
↔ [100 200  47]
```

```
#delete a single value in a array
```

```
a3 = np.delete(a,0)
print(a3)
```

```
↔ [200  47]
```

```
a4 = np.delete(a,[0,1])
print(a4)
```

```
↔ [47]
```

```
#Copy array
a = np.array([45,46,47])
a5 = a.copy()
print(a5)
print(a)
```

```
↔ [45 46 47]
   [45 46 47]
```

```
# sorting an Array
a = np.array([45,46,47])
a6 = np.sort(a)
print(a6)
```

```
↔ [45 46 47]
```

```
#Operations
a = np.array([1,2,3,4])
```

```
b = a+1
print(b)
```

```
↔ [2 3 4 5]
```

```
c = a +0
print(c)
```

```
↔ [1 2 3 4]
```

```
d= a **2
print(d)
```

```
↔ [ 1  4  9 16]
```

```
print (a+b)
```

```
↵ [3 5 7 9]
```

```
print (a-b)
```

```
↵ [-1 -1 -1 -1]
```

```
# Comparision operations
```

```
a == c
```

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

```
c >= a
```

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

```
np.array_equal(a,b)
```

```
↵ False
```

```
np.array_equal(a,b)
```

```
↵ False
```

```
# mathetical operations
```

```
np.exp(a)
```

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

```
import warnings  
warnings.simplefilter("ignore")
```

```
# 2D Numpy Array
```

```
b = np.array([[1,2,3],[4,5,6]])  
print(b)
```

```
↵ [[1 2 3]  
   [4 5 6]]
```

```
b.ndim
```

```
↵ 2
```

```
b.shape
```

```
↵ (2, 3)
```

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

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

```
np.zeros((2,3), dtype=int)
```

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

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

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

```
#identity matrix
np.eye(3,3)
```

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

```
#diagoal matrix
np.diag([1,2,3])
```

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

```
a = np.array([[1,2,3],[4,5,6],[7,8,9]])
print(a)
```

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

```
#indexing of 2D array
```

```
a[0]
```

```
↵ array([1, 2, 3])
```

```
a[2,2]
```

```
↵ np.int64(9)
```

```
a[1:3]
```

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

```
a[1] = [25,30,40]
a
```

```
↵ array([[ 1,  2,  3],
        [25, 30, 40],
        [ 7,  8,  9]])
```

```
a[0,1]=500
a
```

```
↵ array([[ 1, 500,  3],
        [25, 30, 40],
        [ 7,  8,  9]])
```

```
a = np.array ([[0,1,2], [3,4,5]])
print(a)
b = a.ravel()
print(b)
```

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

```
# Reshape array
```

```
arr = np.arange(6)
print(arr)
print(arr.shape)
```

```
↵ [0 1 2 3 4 5]
   (6,)
```

