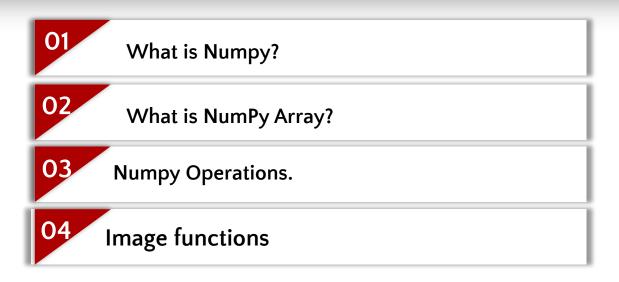
# NumPy

Numerical Python a powerful n-dimensional array tool / library

#### Agenda





# After successfully installing Numpy library we shall create a numpy array

```
#Creating a single dimensional array
import numpy as np #importing the numpy library as np
a = np.array([1,2,3,4]) #creating a one dimensional array using
numpy
print(a) #printing the array
```

[1 2 3 4]

## Now, let's create a ndarray having elements of

#### float data type

```
#creating a ndarray having elements of float data type import numpy as np a = np.array([0.2, 5.2, 3]) print(a)
```

[0.2 5.2 3.]

## Creating a two dimensional array

```
#Creating a 2 dimensional array import numpy as np b = np.array([(1,2,3), (4,5,6)]) print(b)
```

```
[ [1 2 3]
[4 5 6] ]
```

# NumPy Operations It help us finding the:

#1

Dimension of array.

#2

Byte size each element.

#3

Data.

List item.

#### ndim()

This function helps us to find the dimension of the array, whether it is a multidimensional array or a single dimensional array.

```
#find the dimension of array import numpy as np a = np.array([(1,2,3), (2,3,4)]) print(a.ndim)
```

2

#### itemsize()

```
#To check the size of a single element in the array import numpy as np
A = np.array([1, 2, 3, 4, 5, 6], dtype = np.int8)
print(A.itemsize)
```

1

#### dtype()

dtype(): Using this function we can find the data type of the elements stored in an array

To find the data type of the elements of the array:

```
import numpy as np
a = np.array([4.5, 22.8, 6.2, 1.1])
print("the data type of the elements of array is: ", a.dtype)
```

Output: the data type of the elements of array is: float64

#### size

Size of array: The size of array means the total number of elements present in the array.

To check the size of array

import numpy as np
a = np.array([(1,2,3,4,5,6)])
print(a.size)

Output: 6

#### Slicing and Indexing

Slicing in python means taking elements from one given index to another given index.

We can slice instead of index like this [start:end].

We can also define the step, like this: [start:end:step].

Note: The result includes the start index, but excludes the end index.

### Reading and Showing images

To use the OpenCV library in python, we need to install these libraries as a prerequisite:

1. Numpy Library: The computer processes images in the form of a matrix for which NumPy is used and OpenCV uses it in the background.

2. OpenCV python: OpenCV library previously it was cv but the updated version is cv2. It is used to manipulate images and videos.

### **Image Functions**

Function	Description
Imread()	Read an image from a file into an array.
Imsave()	Save an array as in image file.
Imshow()	Display an image on the axes.

The steps to read and display an image in OpenCV are:

- 1.Import Numpy and cv2
- 2.Read an image using imread() function.
- 3. Create a GUI window and display image using imshow() function.
- 4. Use function waitkey(0) to hold the image window on the screen by the specified number of seconds, o means till the user closes it, it will hold GUI window on the screen.
- 5. Delete image window from the memory after displaying using destroyAllWindows() function.

#### Basic code

```
import cv2
import numpy as np
img=cv2.imread("path",cv2.flag)
cv2.imshow("name to show(window",img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

#### Reading an Image

For reading an image, use the imread() function in OpenCV.

Here's the syntax:imread(filename, flags)

It takes two arguments:

- 1. The first argument is the image name, which requires a fully qualified pathname to the file.
- 2. The second argument is an optional flag that lets you specify how the image should be represented. OpenCV offers several options for this flag

#### Displaying an image

In OpenCV, you display an image using the imshow() function.

Here's the syntax: cv2.imshow(window\_name, image)

This function also takes two arguments:

- 1. The first argument is the window name that will be displayed on the window
- 2. The second argument is the image that you want to display.

#### Writing an image

How to write/save an image into the file directory, using the imwrite() function.

Syntax: cv2.imwrite(filename, image).

- 1. The first argument is the filename, which must include the filename extension (for example.png.jpg etc). OpenCV uses this filename extension to specify the format of the file.
- 2. The second argument is the image you want to save. The function returns True if the image is saved successfully.

Function	Syntax
Grayscale	<pre>image = cv2.imread('colourful.jpg') grayscale = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY) cv2.imshow('Grayscale', grayscale)</pre>
Shape	<pre>img = cv2.imread("test.jpeg") print(img.shape) Output : (height, width, channel)</pre>
Cropping	img = cv2.imread("test.jpeg") crop = img[50:180, 100:300] cv2.imshow('cropped', crop)