# Introduction

## What Is Matplotlib?

**matplotlib** is a plotting library used for 2D graphics in python programming language.

## What is Matplotlib used for?

Matploitlib is a Python Library used for plotting, this python library provides and objected-oriented APIs for integrating plots into applications.

## Is Matplotlib Included in Python?

Matplotlib is not a part of the Standard Libraries which is installed by default when Python, there are several toolkits which are available that extend python matplotlib functionality.

## Python Matplotlib : Types of Plots

There are various plots which can be created using python matplotlib.


## Download and Install Matplotlib

The matplotlib installation process is very simple. If you are using Anaconda Navigator, then no need to install matplotlib. If you are using Python idle, PyCharm, Sublime Text, etc, then follow the below steps.

Open the command prompt or terminal and enter

```
pip install matplotlib
```

**Introduction**

# Verify the Installation

To verify that matplotlib is installed properly or not, type the following command includes calling .__version __ in the terminal.

```
import matplotlib
matplotlib.__version__
```

For plotting graph, need to import matplotlib in the python file. Here is two way to import matplotlib given below.

```
from matplotlib import pyplot as plt   #or
import matplotlib.pyplot as plt #  importing pyplot module from
```

```
import matplotlib.pyplot as plt
x=[1,3]
y=[3,5]
plt.plot(x,y)
plt.show()
```

`matplotlib.pyplot` is a collection of functions that make matplotlib work like MATLAB.

Each `pyplot` function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.

## Title, Labels, Legends

```
import matplotlib.pyplot as plt
x=[1,3,8,5,9]
y=[2,5,2,6,8]
plt.plot(x,y)
plt.title('creative')
plt.ylabel('shailesh')
plt.xlabel('gautam')
plt.show()
```

# CREATIVE INSTITUTE DATA SCIENCE

## Introduction

### Lengends

```
import matplotlib.pyplot as plt
x=[1,3,8,5,9]
y=[2,5,2,6,8]

x2=[2,3,5,7,6]
y2=[3,6,2,6,8]
plt.plot(x,y,label='line 1')
plt.plot(x2,y2,label='line 2')
plt.title('creative institute of data science')
plt.ylabel('Y axis')
plt.xlabel('X axis')
plt.legend()

plt.show()
```

### Color in Graph

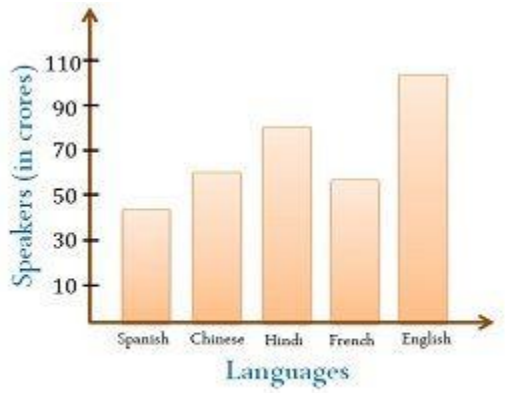
```
import matplotlib.pyplot as plt
x=[1,2,5,3]
y=[2,4,6,8]
x1=[3,5,7,9]
y1=[3,4,5,6]
plt.plot(x,y,label="line 1",color="salmon")
plt.plot(x1,y1,label='line 2',color="black")
plt.legend()
plt.show()
```

### BAR CHARTS

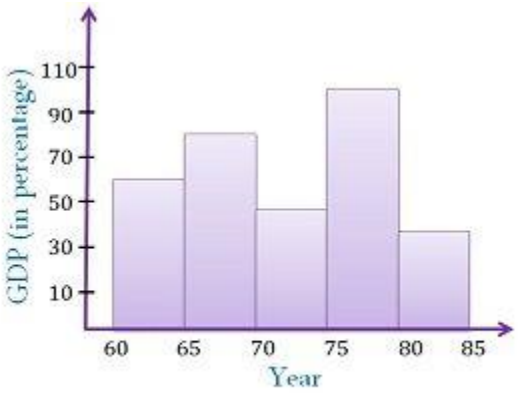
```
import matplotlib.pyplot as plt
x=[1,3,8,5,9]
y=[2,5,2,6,8]

plt.bar(x,y,label='line 1',color='red')

plt.title('creative institute of data science')
plt.ylabel('Y axis')
plt.xlabel('X axis')
plt.legend()

plt.show()
```

# CREATIVE INSTITUTE DATA SCIENCE

## Introduction




**Bar Graph**




**Histogram**

```
import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5]
y = [6, 2, 1, 1.5, 9]

plt.bar(x, y, label="Bar 1")

plt.xlabel("Month")
plt.ylabel("Sales")

plt.title("Sales Per Month")
plt.legend()

plt.show()
```

**Dual Bar Graph**

```
import matplotlib.pyplot as plt

x = [1, 3, 5, 7, 10]
y = [6, 2, 1, 1.5, 9]
x2 = [2, 4, 4, 8, 9]
y2 = [3, 3, 6, 1, 6]

plt.bar(x, y, label="Store A")
plt.bar(x2, y2, label="Store B")

plt.xlabel("Month")
plt.ylabel("Sales")
```

## Introduction

```
plt.title("Sales Per Month Of Store A and B")
plt.legend()

plt.show()
```

## HISTOGRAMS

Difference between a bar graph and a histogram. Histograms are used to show a distribution whereas a bar chart is used to compare different entities.

Histograms are useful when you have arrays or a very long list.

This example where I have to plot the age of population with respect to bin.

Now, bin refers to the range of values that are divided into series of intervals.

Bins are usually created of the same size

I have created the bins in the interval of 10 which means the first bin contains elements from 0 to 9, then 10 to 19 and so on.

```
import matplotlib.pyplot as plt

age = [21, 32, 69, 56, 85, 45, 50, 30, 10, 15, 16, 42, 66, 77, 11, 2, 32, 9, 91,
85, 81, 92, 27]


bin = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

plt.hist(age, bin, histtype="bar")

plt.xlabel("X-Axis SIde")
plt.ylabel("y-Axis SIde")

plt.title("A Simple Histogram")
#plt.legend()
plt.show()
```

for spacing between graph

```
plt.hist(age, bin, histtype="bar",rwidth=0.7)
```

# CREATIVE INSTITUTE DATA SCIENCE

## Introduction

### Scatter Plots / Scatter Graph

```
import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5, 6, 7, 8, 9]
y = [5, 7, 3, 4, 4, 6, 1, 7, 9]


plt.scatter(x, y, label="Creative Institute", color="maroon")
plt.xlabel("X-Axis Name")
plt.xlabel("Y-Axis Name")
plt.title("Scatter Plots / Scatter Graph")
plt.legend()
plt.show()
```

### matplotlib.markers

```
import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5, 6, 7, 8, 9]
y = [5, 7, 3, 4, 4, 6, 1, 7, 9]

plt.scatter(x,y,label='creative institute',color='maroon',marker='x')
plt.xlabel('x axis')
plt.ylabel('y axis')
plt.title('creative institue multimeieda')
plt.legend()
plt.show()
```

### Stack Plots

```
import matplotlib.pyplot as plt

days = [1, 2, 3, 4, 5]

sleeping = [7, 8, 7.5, 9, 12]
working = [3, 6, 5, 4, 2]
exercise = [1, 0.5, 1, 0.5, 0.5]
study = [2, 3, 2, 4, 6]

plt.stackplot(days,sleeping,working,exercise,study,labels=['days','sleeping','wor
king','exercise','study'])
plt.legend()
plt.show()
```

## Introduction

# Pie Charts

pie charts are similar to stack plots/ stack graphs .the only difference is that pie charts represent data in a circle. In some cases pie charts are preferred over stack plots. both have their own uses. You must know when to use what graph !

```
import matplotlib.pyplot as plt

chart = [7, 2, 3, 7, 6]
label = ["Sleeping", "playing", "music", "coding", "eating"]

plt.pie(chart, labels=label,)

plt.title("PIE CHARTS exercise")
plt.show()
```

### percentage value in pie chart

use attribut autopct=’%f%%’

```
plt.pie(chart, labels=label,autopct='%.2f%%')
```

increase radius

```
plt.pie(chart, labels=label,autopct='%.2f%%',radius=1.5)
```

# CREATIVE INSTITUTE DATA SCIENCE

**Introduction**

## ## Loading Data From File

```
import matplotlib.pyplot as plt
import csv

x=[]
y=[]

import csv
import matplotlib.pyplot as plt

x = []
y = []

with open('data.txt', 'r') as csvfile:
    plots = csv.reader(csvfile, delimiter=",")
    for row in plots:
        x.append(int(row[0]))
        y.append(int(row[1]))

plt.plot(x, y)
plt.xlabel("X-Axis Name")
plt.ylabel("Y-Axis Name")
plt.title("Loading Data Form FIle")
plt.show()
```

Using Numpy Library to Load Txt File

```
import matplotlib.pyplot as plt
import numpy as np

x = []
y = []

x,y=np.loadtxt('data.txt',delimiter=',',unpack=True)
plt.scatter(x,y)
plt.show()
```