**MATPLOTLIB**

**MATPLOT:**

Matplotlib provide us to create variety of plots such as :

* + Line Charts
  + Scattered Plots
  + Bar Plots

Its built on top of NumPy Library.

**Line Charts:**

**Code:**

import matplotlib.pyplot as plt  
import numpy as np  
  
xpoints = np.array([2019,2020,2021,2023,2024])  
ypoints = np.array([60,62,65,67,75])  
  
plt.title('Last 5 Yrs Salary')  
plt.xlabel('Year')  
plt.ylabel('Salary in Euro(k)')  
plt.grid(color='LightBlue', linestyle='dotted', linewidth=2, axis='y')  
plt.plot(xpoints, ypoints, color='Blue', alpha=0.75, linewidth=5)  
plt.show()

**Output:**

A screenshot of a computer

Description automatically generated

After adding the labels

A screenshot of a computer

Description automatically generated

After adding colour, grid colour, line style and line width

A screenshot of a computer

Description automatically generated

Changing the alpha value, which is default 1 and the value has to be between 0 and 1.

Also, added the linewidth value for the plot.

A screenshot of a computer

Description automatically generated

**Scattered Plot :** Can be used to establish the relationship between two variables

**Code:**

import matplotlib.pyplot as plt  
import numpy as np  
  
xpoints = np.array([2019,2020,2021,2023,2024])  
ypoints = np.array([60,62,65,67,75])  
  
plt.title('Last 5 Yrs Salary')  
plt.xlabel('Year')  
plt.ylabel('Salary in Euro(k)')  
plt.grid(color='LightBlue', linestyle='dotted', linewidth=2, axis='y')  
*#plt.plot(xpoints, ypoints, color='Blue', alpha=0.75, linewidth=5)*plt.scatter(xpoints, ypoints, color='Blue', alpha=0.75, linewidth=5)  
plt.show()

**Output:**

A screen shot of a computer

Description automatically generated

**Bar Graph:**

**Code:**

import matplotlib.pyplot as plt  
import numpy as np  
  
xpoints = np.array([2020,2021,2022,2023,2024])  
ypoints = np.array([60,62,65,67,75])  
  
plt.title('Last 5 Yrs Salary')  
plt.xlabel('Year')  
plt.ylabel('Salary in Euro(k)')  
plt.grid(color='LightBlue', linestyle='dotted', linewidth=2, axis='y')  
*#plt.plot(xpoints, ypoints, color='Blue', alpha=0.75, linewidth=5)  
#plt.scatter(xpoints, ypoints, color='Blue', alpha=0.75, linewidth=5)*plt.bar(xpoints, ypoints, color='Green', alpha=0.75, linewidth=2)  
plt.show()

**Output:**

A screenshot of a computer

Description automatically generated

**Histogram:** It’s a frequency distribution graph. The y axis represents the count.

**Code:**

import matplotlib.pyplot as plt  
import numpy as np  
  
mobile\_price= np.array([1075.99, 1000.75, 1170.89, 1789.25, 2189.25, 1075.99, 1170.89, 1250.75, 899.99, 549, 799, 1029])  
plt.hist(mobile\_price, rwidth=0.5)  
plt.show()

**Output:**

A screenshot of a computer

Description automatically generated

**Multiple Subplots:** The matplotlib.pyplot.subplots method provides a way to plot multiple plots on a single frame.

Given the number of rows and columns, it returns a tuple.

(fig, ax), giving a single figure fig with an array of axes ax.

fig: The matplot.pyplot.figure object to be used as a container for all the subplots.

ax: Axes object if t5here is only one plot or any array of axes.

Axes objects if there are multiple plots, as specified by the nrows and ncols.

**Code:**

import matplotlib.pyplot as plt  
import numpy as np  
  
fig,ax = plt.subplots(nrows=1, ncols = 3)  
fig.suptitle('Mutliple Subplots')  
  
xpoints = np.array([2015,2016,2017,2018,2019])  
ypoints = np.array([45,50,52,57,59])  
ax[0].plot(xpoints, ypoints, color='Green')  
  
xpoints1 = np.array([2020,2021,2022,2023,2024])  
ypoints1 = np.array([60,62,65,67,75])  
ax[1].plot(xpoints1, ypoints1, color='Blue')  
  
xpoints2 = np.array([2025,2026,2027,2028,2029])  
ypoints2 = np.array([85,95,105,110,125])  
ax[2].plot(xpoints2, ypoints2, color='Orange')  
  
plt.show()

**Output:**

A screenshot of a computer

Description automatically generated