Python

Matplotlib Library

Installing Matplotlib

```
pip install matplotlib
```

Importing Libraries

1. Import matplotlib.pyplot as plt
2. From matplotlib import pyplot as plt
3. import numpy as np
4. import pandas as pd
5. import csv

Different Styles Available

1.Solarize_Light2

2._classic_test_patch

3.bmh

4.classic

5.dark_background

6.fast

7.fivethirtyeight

8.ggplot

9.grayscale

10.seaborn

11.seaborn-bright

12.seaborn-colorblind

13.seaborn-dark

14.seaborn-dark-palette

15.seaborn-darkgrid

16.seaborn-deep

17.seaborn-muted

18.seaborn-notebook

19.seaborn-paper

20.seaborn-pastel

21.seaborn-poster

22.seaborn-talk

23.seaborn-ticks

24.seaborn-white

25.seaborn-whitegrid

26.tableau-colorblind10

To use :

```
plt.style.use(‘Name_of_the_style’)
```

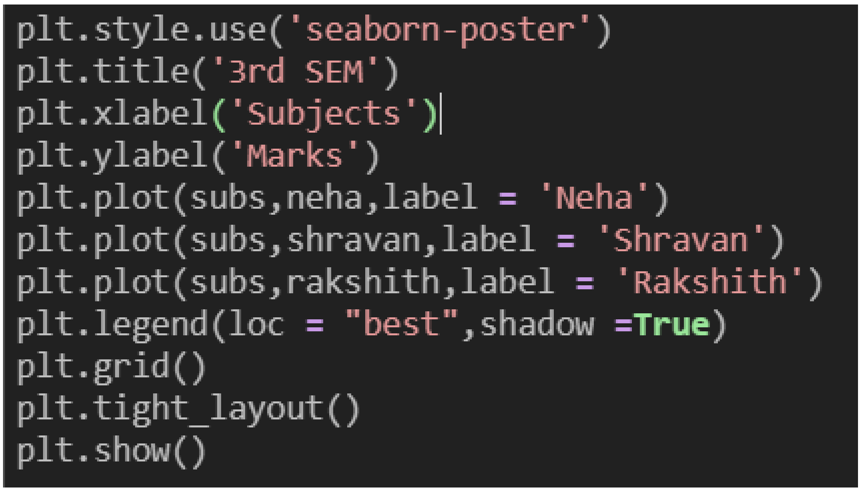
Different Types of Plots

1.Line Plots

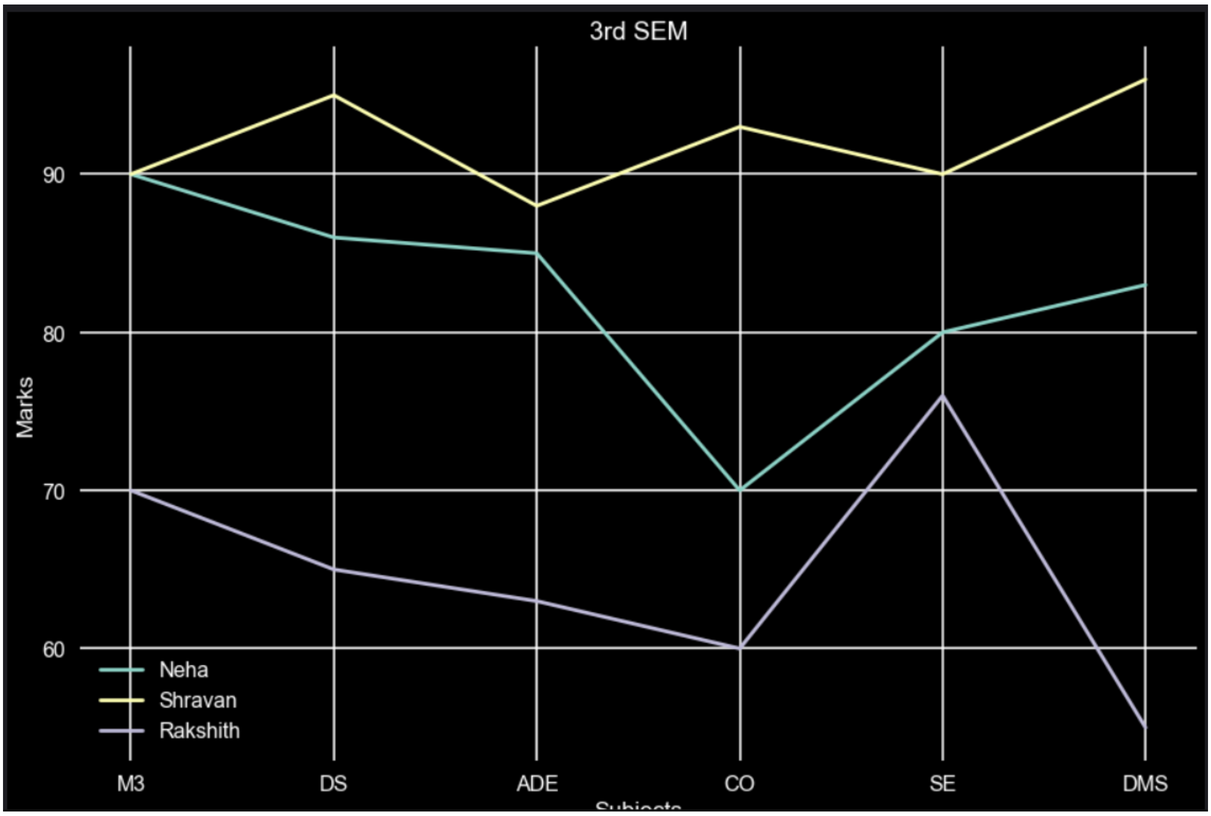
Syntax:
plt.plot(xplotvalue, yplotvalue,label = ‘name_of_the_line’,color = ‘color_name’, linestyle = ‘style_name’,marker = ‘marker_type’,linewidth= ‘any_desired_number’,)

Notes
1.color can be any hexadecimal number
2.linestyle can be for eg “--”
3.further information at
https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.plot.html#matplotlib.pyplot.plot

Example



```
plt.style.use('seaborn-poster')
plt.title('3rd SEM')
plt.xlabel('Subjects')
plt.ylabel('Marks')
plt.plot(subs,neha,label = 'Neha')
plt.plot(subs,shravan,label = 'Shravan')
plt.plot(subs,rakshith,label = 'Rakshith')
plt.legend(loc = "best",shadow =True)
plt.grid()
plt.tight_layout()
plt.show()
```

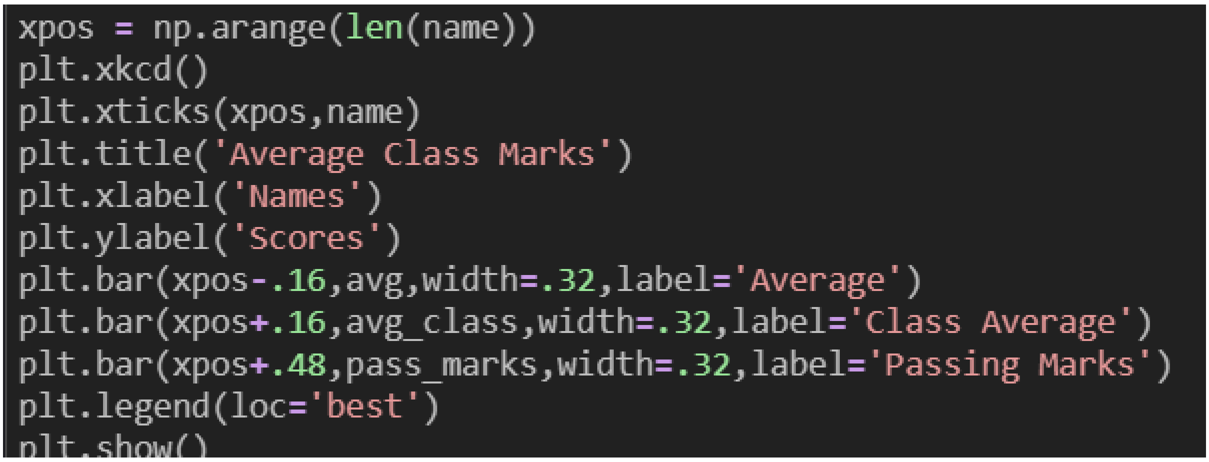


2.Bar Graph

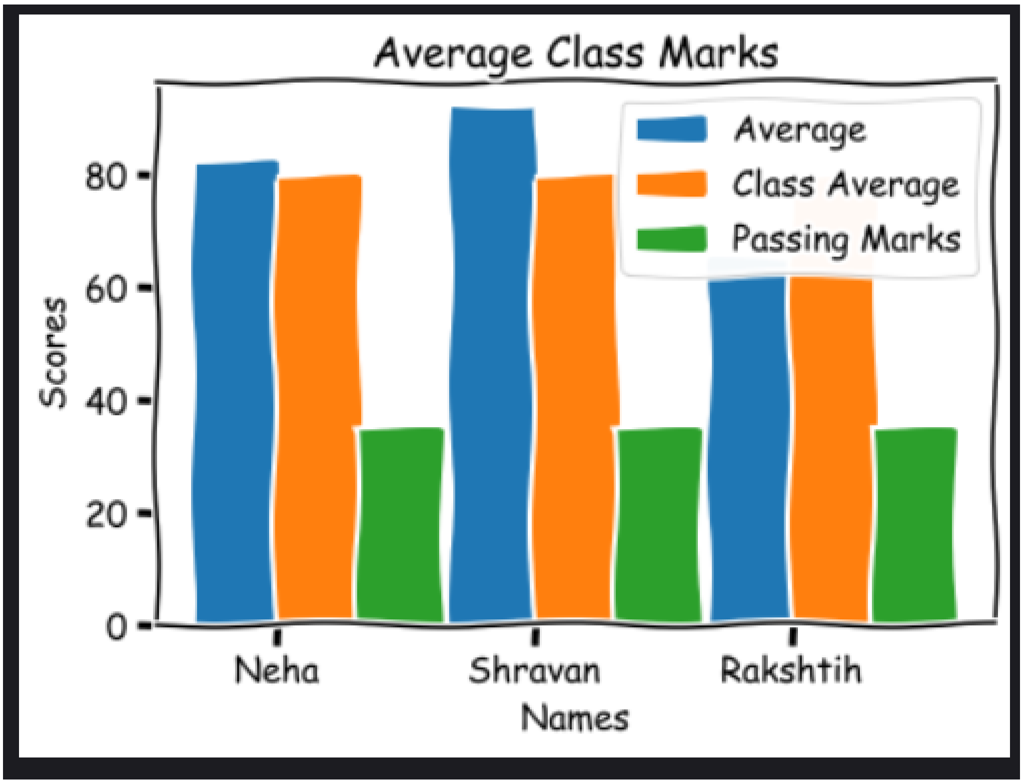
Syntax:
plt.bar(xplotvalue, yplotvalue,label = ‘name_of_the_line’,color = ‘color_name’,width = ‘any_number’)

Notes
1.color can be any hexadecimal number
2.numpy should be imported for bargraphs
3.further information at https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.bar.html#matplotlib.pyplot.bar

Example



```
xpos = np.arange(len(name))
plt.xkcd()
plt.xticks(xpos,name)
plt.title('Average Class Marks')
plt.xlabel('Names')
plt.ylabel('Scores')
plt.bar(xpos-.16,avg,width=.32,label='Average')
plt.bar(xpos+.16,avg_class,width=.32,label='Class Average')
plt.bar(xpos+.48,pass_marks,width=.32,label='Passing Marks')
plt.legend(loc='best')
plt.show()
```

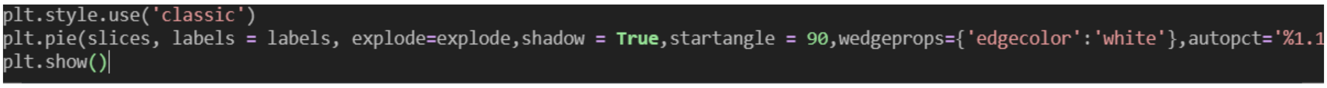


3.Pie Plots

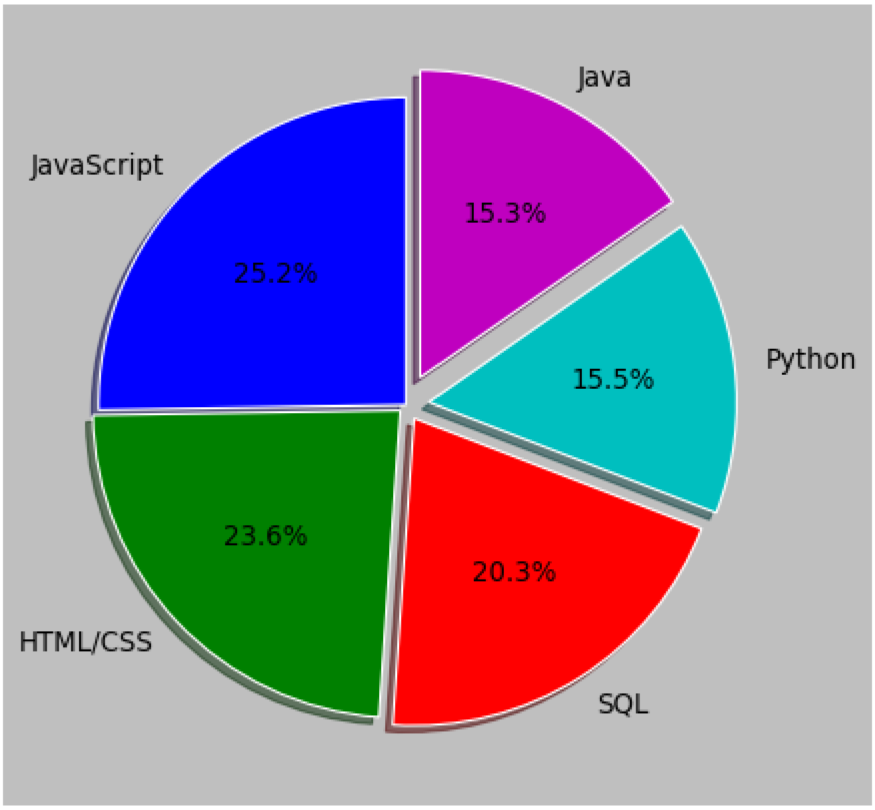
Syntax:
plt.pie(data_to_be_plotted, labels = [labels in list format], explode=[values in list format],shadow = ‘True/False’,startangle = any_angle ,wedgeprops={'edgecolor':'any_color'},autopct='%1.1f%%')

Notes
1.color can be any hexadecimal number
2.further information at
https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.pie.html#matplotlib.pyplot.pie

Example



```
plt.style.use('classic')
plt.pie(slices, labels = labels, explode=explode,shadow = True,startangle = 90,wedgeprops={'edgecolor':'white'},autopct='%1.1
plt.show()
```

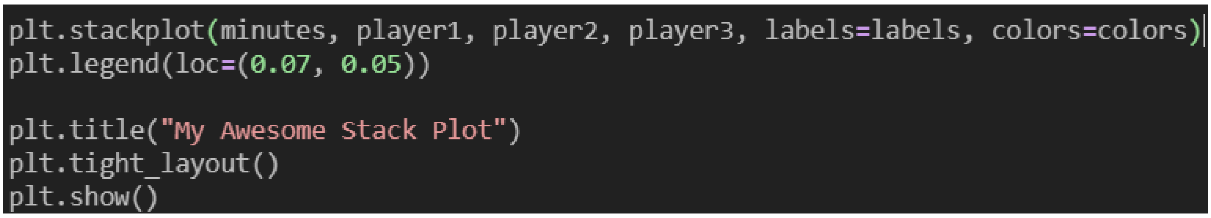


4.Stack Plots

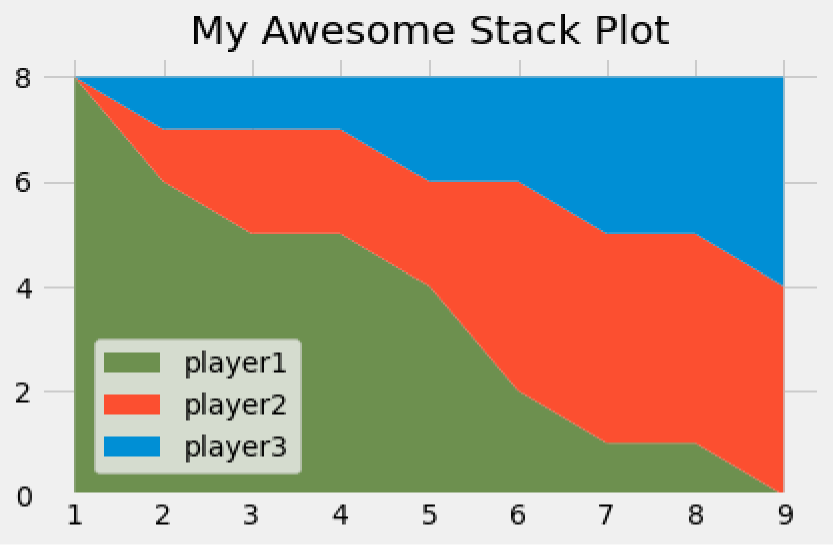
Syntax:
plt.stackplot(Xvalue, Yvalue 1,Yvalue 2……,Yvalue n, labels=labels, colors=colors)

Notes
1.further information at
https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.stackplot.html#matplotlib.pyplot.stackplot

Example



```
plt.stackplot(minutes, player1, player2, player3, labels=labels, colors=colors)
plt.legend(loc=(0.07, 0.05))

plt.title("My Awesome Stack Plot")
plt.tight_layout()
plt.show()
```



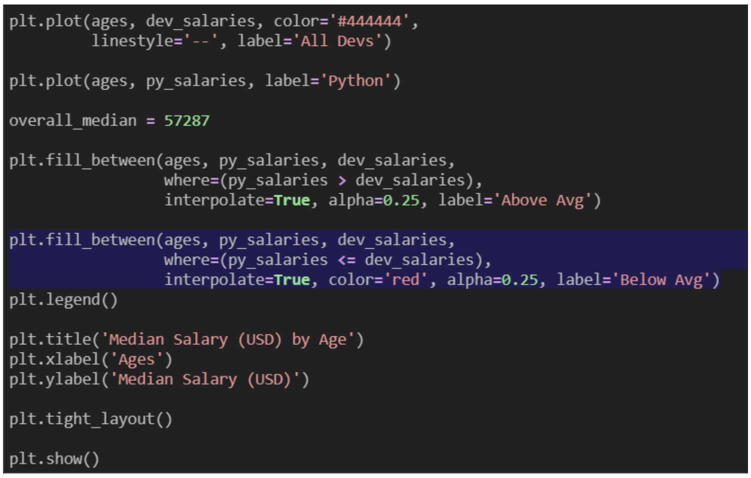
5.Filling Between Line Plots

Syntax:
plt.fill_between(Xvalue, Yvalue1,Yvalue2,
where=(place_you_need_to_fill),
interpolate=True/False, color='Any_Color', alpha=0-1, label='Label')

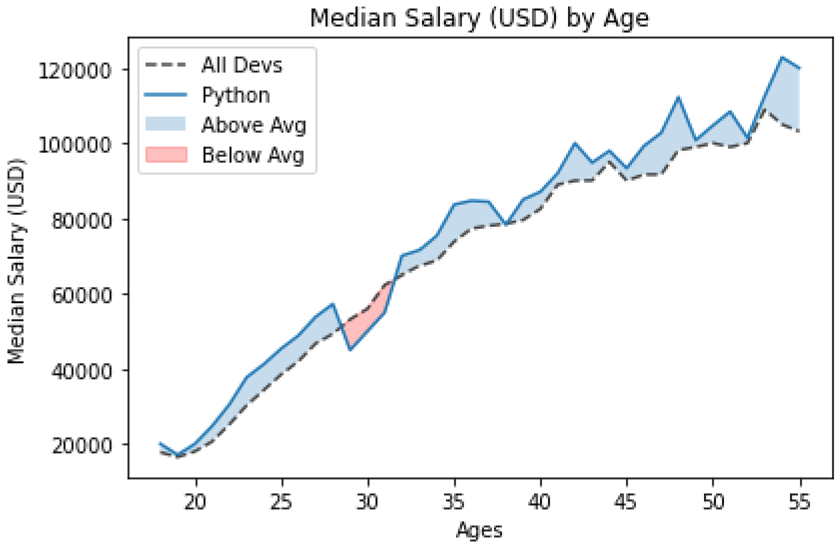
Notes

1.further information at
https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.fill_between.html#matplotlib.pyplot.fill_between

Example



```
plt.plot(ages, dev_salaries, color='#444444',
         linestyle='--', label='All Devs')

plt.plot(ages, py_salaries, label='Python')

overall_median = 57287

plt.fill_between(ages, py_salaries, dev_salaries,
                 where=(py_salaries > dev_salaries),
                 interpolate=True, alpha=0.25, label='Above Avg')

plt.fill_between(ages, py_salaries, dev_salaries,
                 where=(py_salaries <= dev_salaries),
                 interpolate=True, color='red', alpha=0.25, label='Below Avg')
plt.legend()

plt.title('Median Salary (USD) by Age')
plt.xlabel('Ages')
plt.ylabel('Median Salary (USD)')

plt.tight_layout()

plt.show()
```



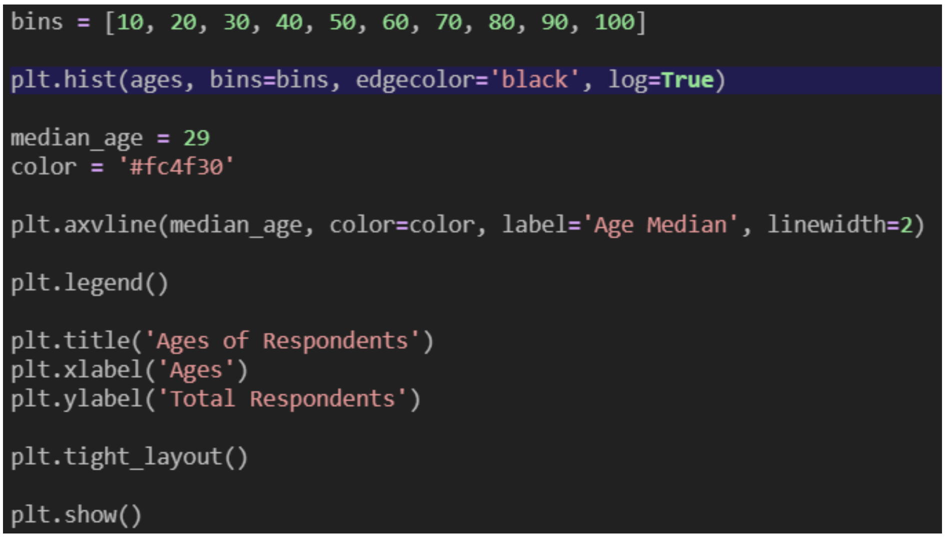
6.Histogram

Syntax:
plt.hist(ages, bins='bin_values', edgecolor='color', log=True/false)

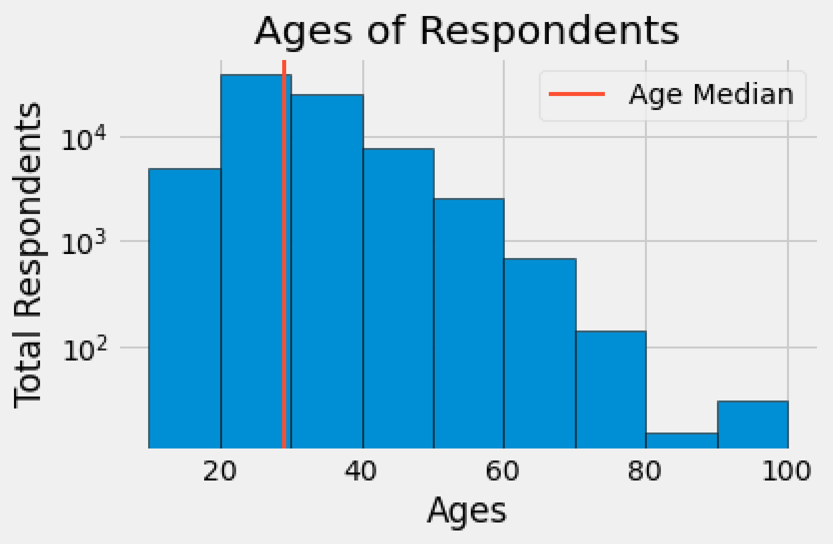
Notes

1.further information at https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.hist.html#matplotlib.pyplot.hist

Example



```
bins = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

plt.hist(ages, bins=bins, edgecolor='black', log=True)

median_age = 29
color = '#fc4f30'

plt.axvline(median_age, color=color, label='Age Median', linewidth=2)

plt.legend()

plt.title('Ages of Respondents')
plt.xlabel('Ages')
plt.ylabel('Total Respondents')

plt.tight_layout()

plt.show()
```



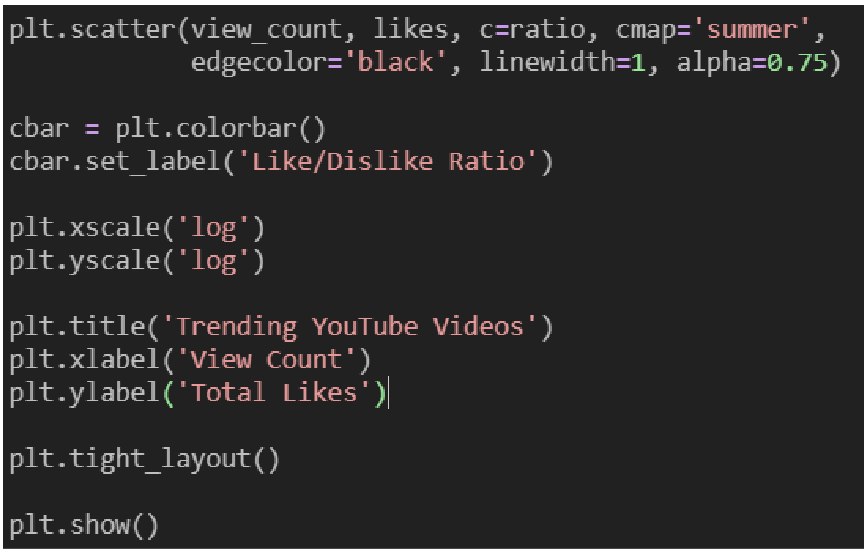
7.Scatter Plot

Syntax:
plt.scatter(xvalue, yvalue, c=color, cmap='color type',
edgecolor='Any_color', linewidth=any_number, alpha=0-1)

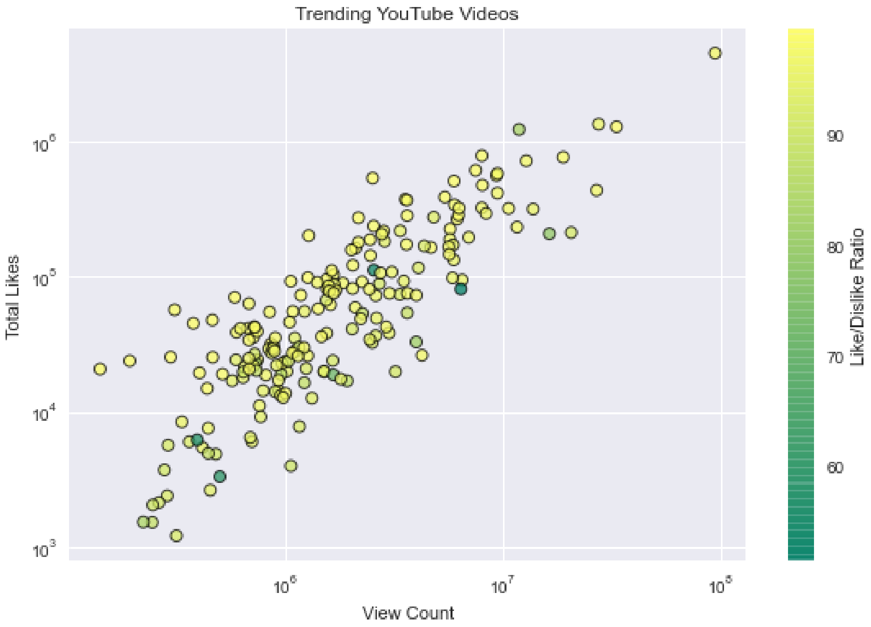
Notes

1.further information at
https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.scatter.html#matplotlib.pyplot.scatter

Example



```
plt.scatter(view_count, likes, c=ratio, cmap='summer',
            edgecolor='black', linewidth=1, alpha=0.75)

cbar = plt.colorbar()
cbar.set_label('Like/Dislike Ratio')

plt.xscale('log')
plt.yscale('log')

plt.title('Trending YouTube Videos')
plt.xlabel('View Count')
plt.ylabel('Total Likes')

plt.tight_layout()

plt.show()
```



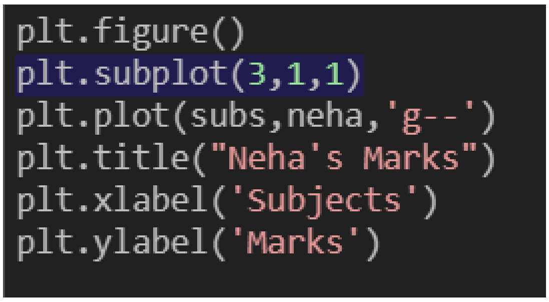
8.Sub Plot

Syntax:
plt.subplot(rows,coloums,index)

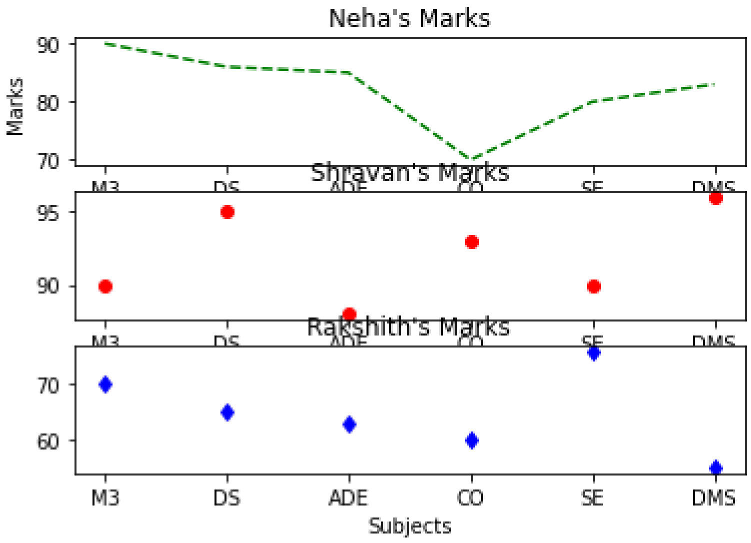
Notes

1.further information at https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.subplot.html#matplotlib.pyplot.subplot

Example



```
plt.figure()
plt.subplot(3,1,1)
plt.plot(subs,neha,'g--')
plt.title("Neha's Marks")
plt.xlabel('Subjects')
plt.ylabel('Marks')
```



IMPORTING FILES

1.Csv method

Step:1

import csv

Step:2

with open('filename.csv') as csv_file:

2.Using pandas

Step:1

import pandas as pd

Step:2

variable_name = pd.read_csv('filename.csv')