

Load Necessary Libraries

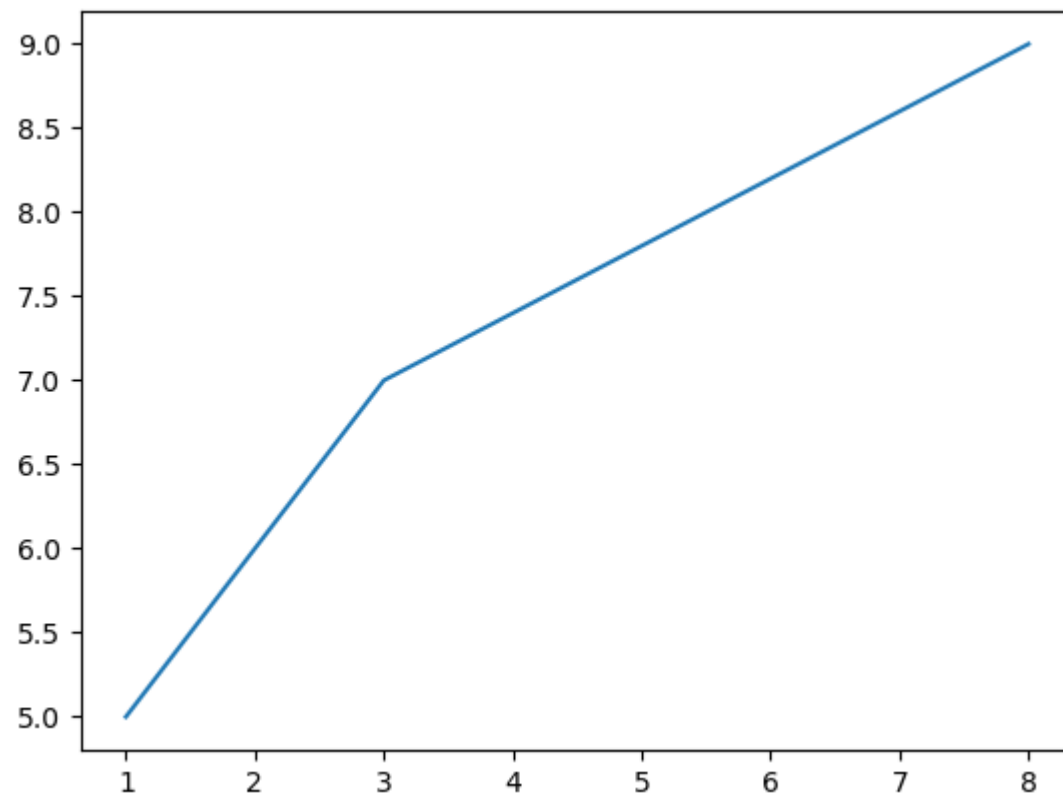
```
In [1]: 1 import matplotlib.pyplot as plt
        2 import pandas as pd
        3 import numpy as np
```

Refer the documentation for reference

https://matplotlib.org/3.5.3/api/_as_gen/matplotlib.pyplot.html (https://matplotlib.org/3.5.3/api/_as_gen/matplotlib.pyplot.html)

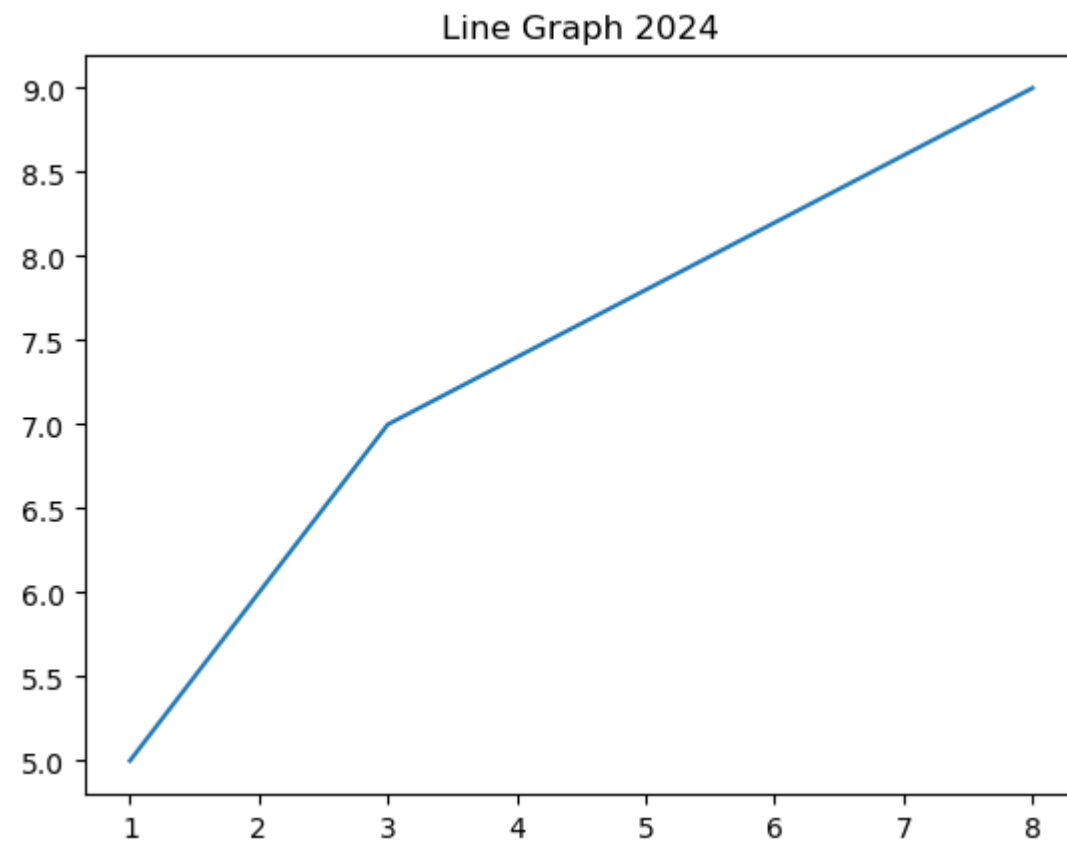
Basic Graph Plot

```
In [2]: 1 x=[1,2,3,8]
        2 y=[5,6,7,9]
        3
        4 plt.plot(x,y) # This will plot the Graph
        5 plt.show()
```



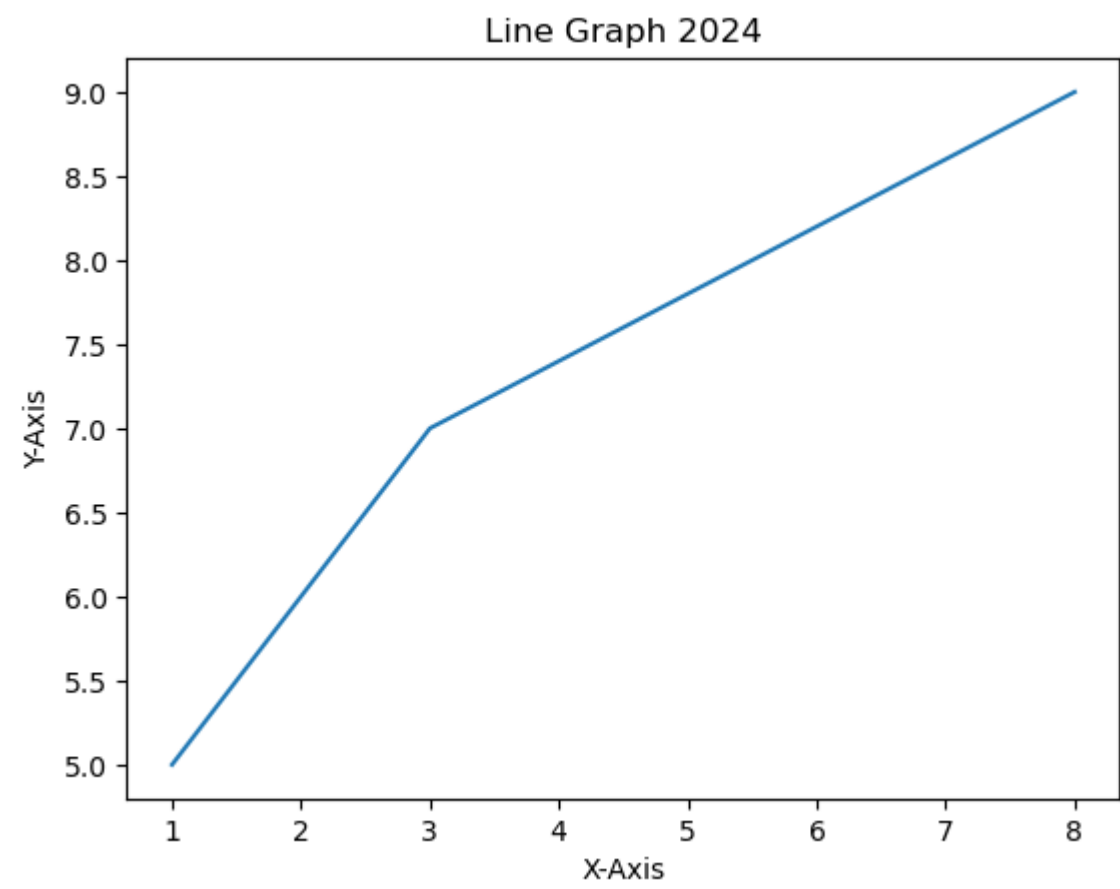
Title to the Graph

```
In [3]: 1 x=[1,2,3,8]
        2 y=[5,6,7,9]
        3
        4 plt.plot(x,y)
        5
        6 plt.title("Line Graph 2024") # This will give label to the Graph
        7
        8 plt.show()
```



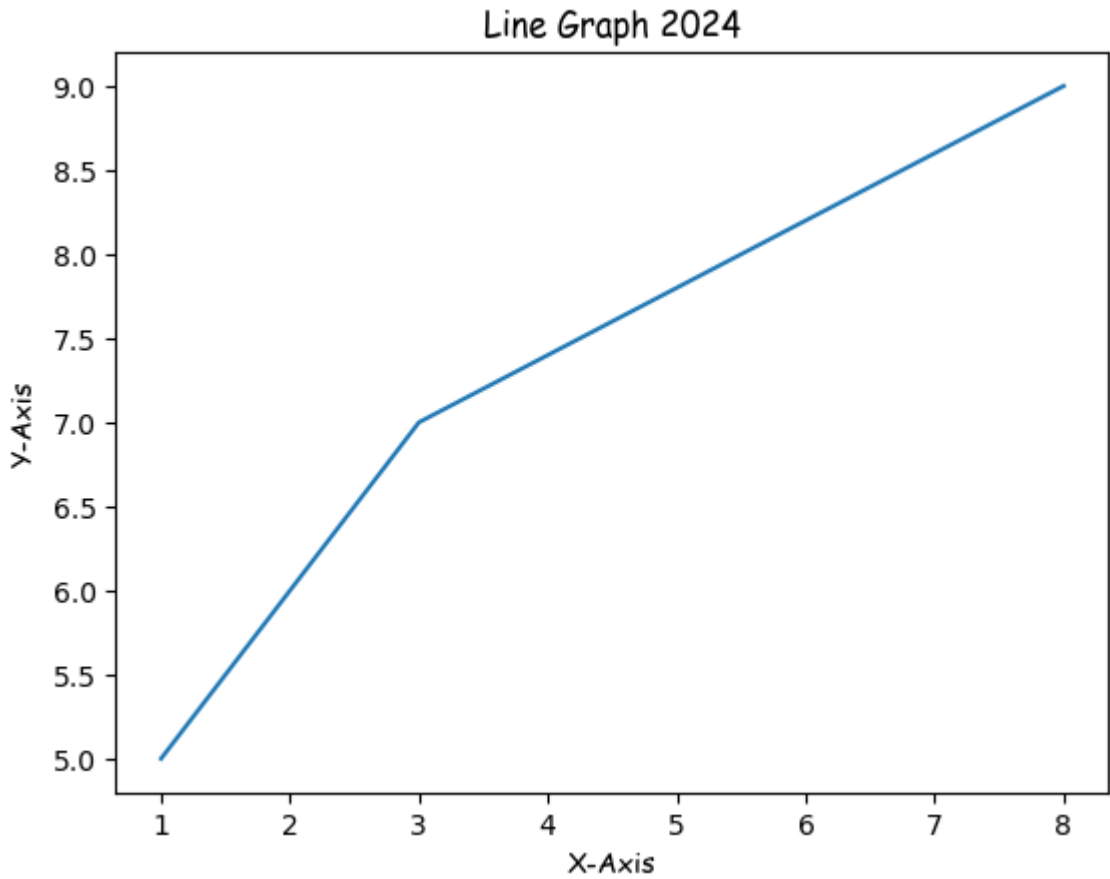
Labels to X and Y axis

```
In [4]: 1 x=[1,2,3,8]
        2 y=[5,6,7,9]
        3
        4 plt.plot(x,y)
        5
        6 plt.title("Line Graph 2024")
        7 plt.xlabel("X-Axis") # This provides Label to X-axis
        8 plt.ylabel("Y-Axis") # This provides Label to Y-axis
        9
       10 plt.show()
```



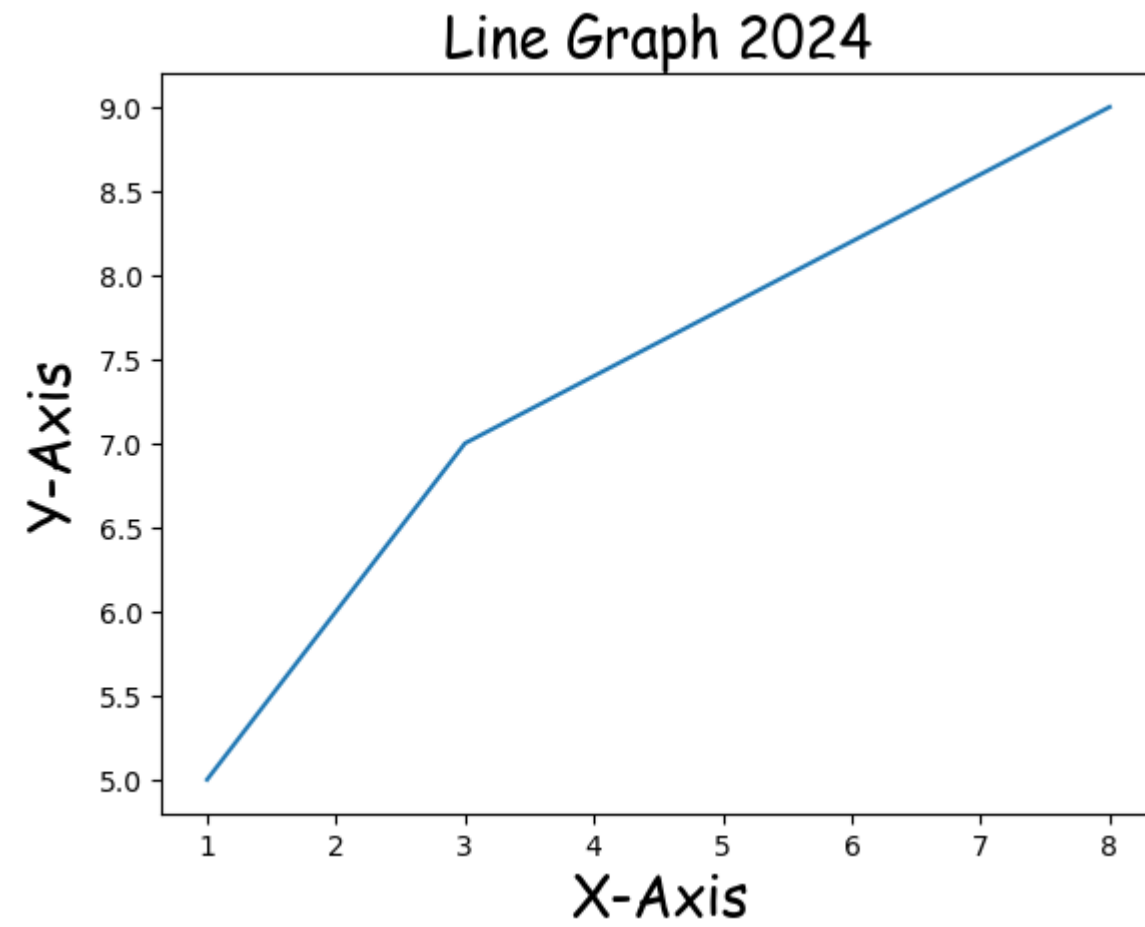
Font Name

```
In [5]: 1 x=[1,2,3,8]
2 y=[5,6,7,9]
3
4 plt.plot(x,y)
5
6 plt.title("Line Graph 2024",fontdict={'fontname':'Comic Sans MS'}) # This will change the font type
7 plt.xlabel("X-Axis",fontdict={'fontname':'Comic Sans MS'})
8 plt.ylabel("Y-Axis",fontdict={'fontname':'Comic Sans MS'})
9
10 plt.show()
```



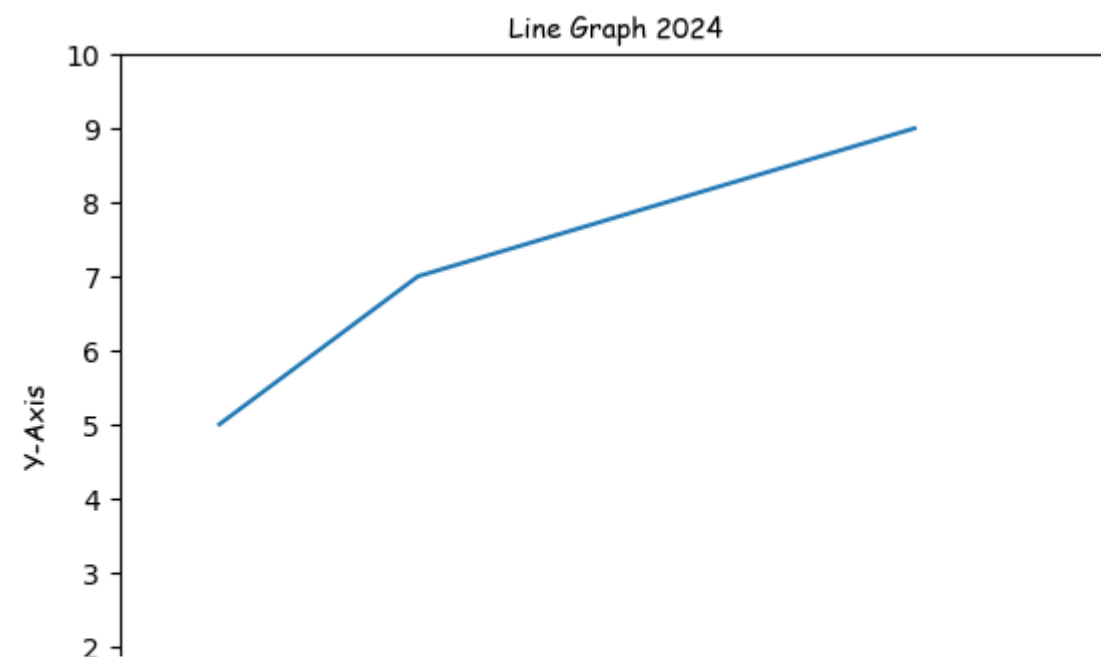
Font Size

```
In [6]: 1 x=[1,2,3,8]
2 y=[5,6,7,9]
3
4 plt.plot(x,y)
5
6 plt.title("Line Graph 2024",fontdict={'fontname':'Comic Sans MS','fontsize':20}) # This will change the font size
7 plt.xlabel("X-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':20})
8 plt.ylabel("Y-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':20})
9
10 plt.show()
```



Scale/Ticks of the Graph

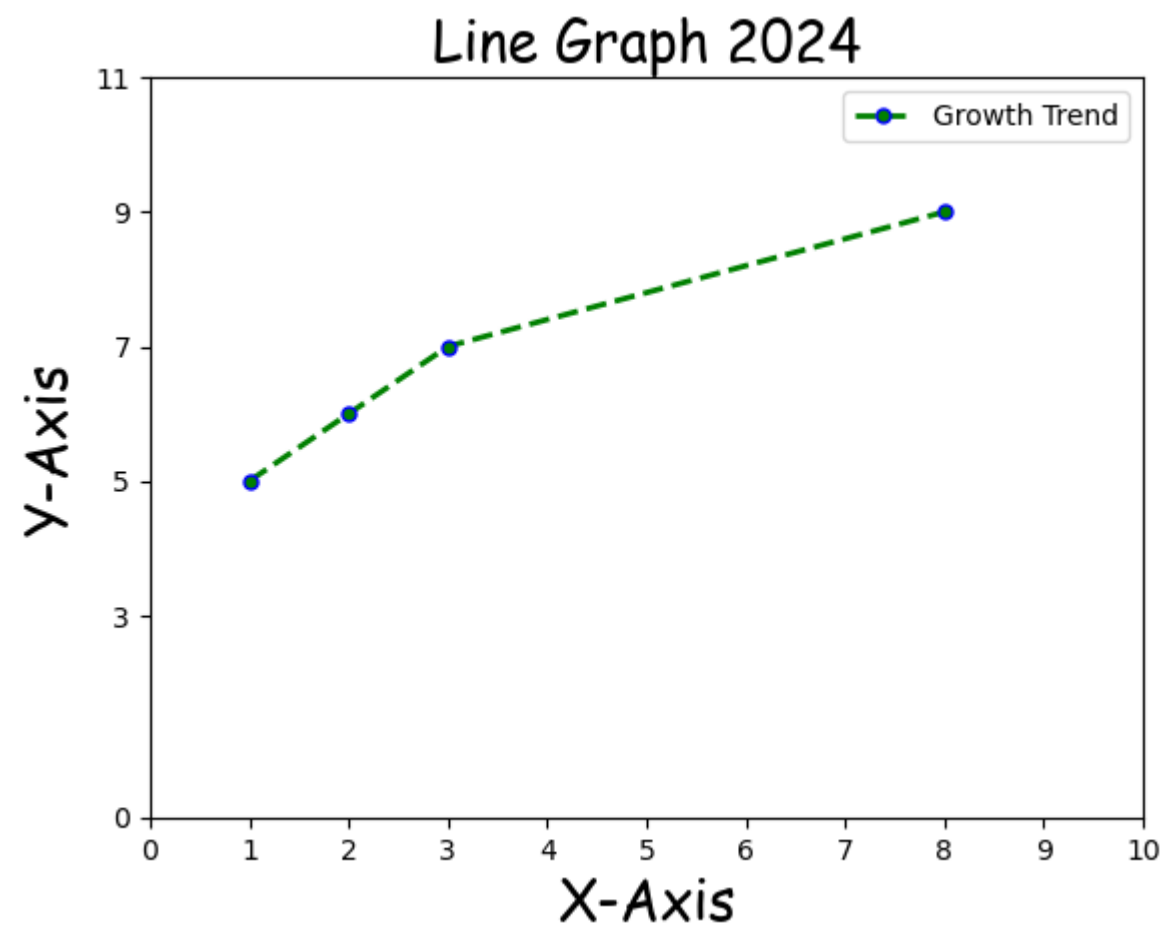
```
In [7]: 1 x=[1,2,3,8]
2 y=[5,6,7,9]
3
4 plt.plot(x,y)
5
6 plt.title("Line Graph 2024",fontdict={'fontname':'Comic Sans MS','fontsize':10})
7 plt.xlabel("X-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':10})
8 plt.ylabel("Y-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':10})
9
10
11 plt.xticks([0,1,2,3,4,5,6,7,8,9,10]) # This will scale the X and Y axis as per choice
12 plt.yticks([0,1,2,3,4,5,6,7,8,9,10])
13
14 plt.show()
```



Legend

Parameters: label, color, linewidth,linestyle,marker,markersize,markeredgcolor

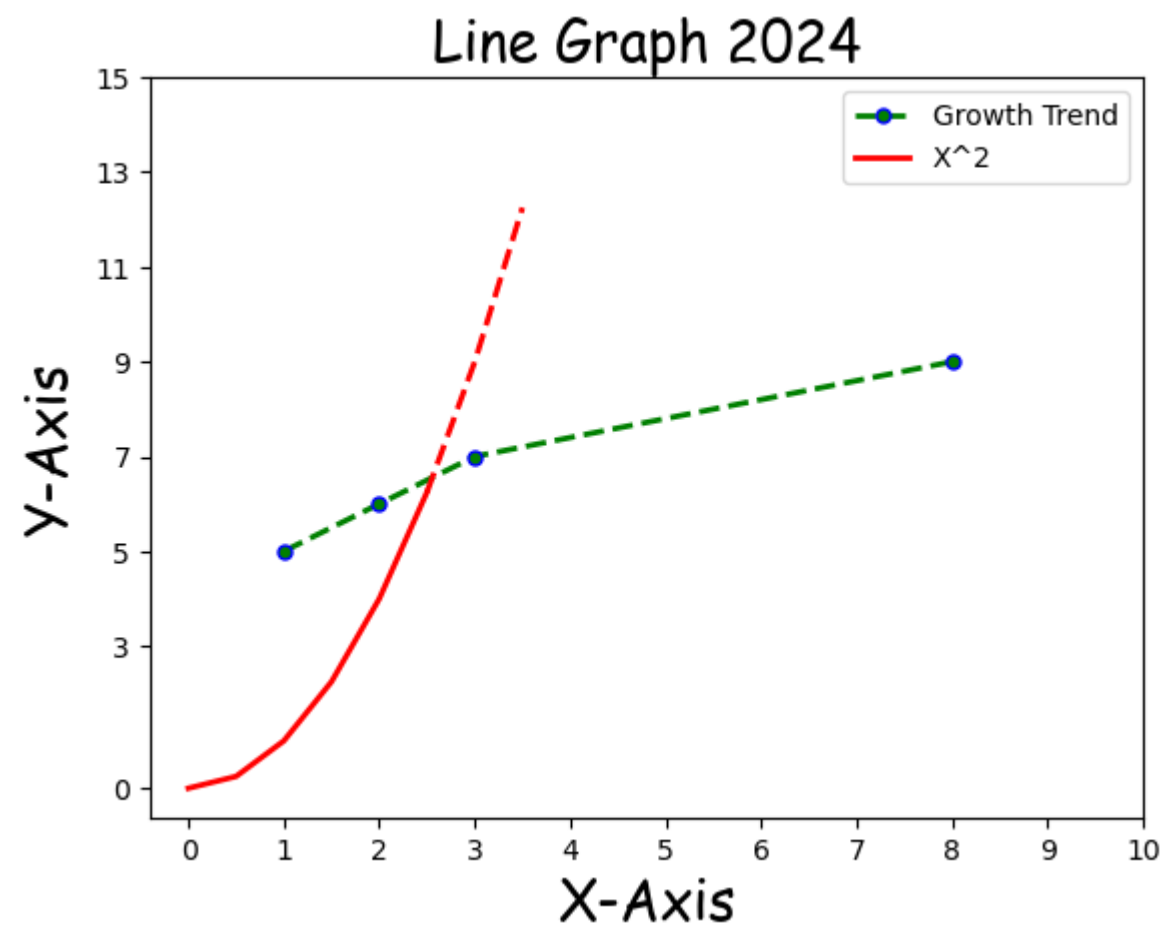
```
In [8]: 1 x=[1,2,3,8]
2 y=[5,6,7,9]
3
4 # Defining the Legend parameters
5
6 plt.plot(x,y,label="Growth Trend",color='green',linewidth=2,marker='.',linestyle='--', markersize=10,markeredgcolor='blue')
7
8 plt.title("Line Graph 2024",fontdict={'fontname':'Comic Sans MS','fontsize':20})
9 plt.xlabel("X-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':20})
10 plt.ylabel("Y-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':20})
11 plt.xticks([0,1,2,3,4,5,6,7,8,9,10])
12 plt.yticks([0,3,5,7,9,11])
13
14 plt.legend()
15
16 plt.show()
```



2 Lines

In [22]:

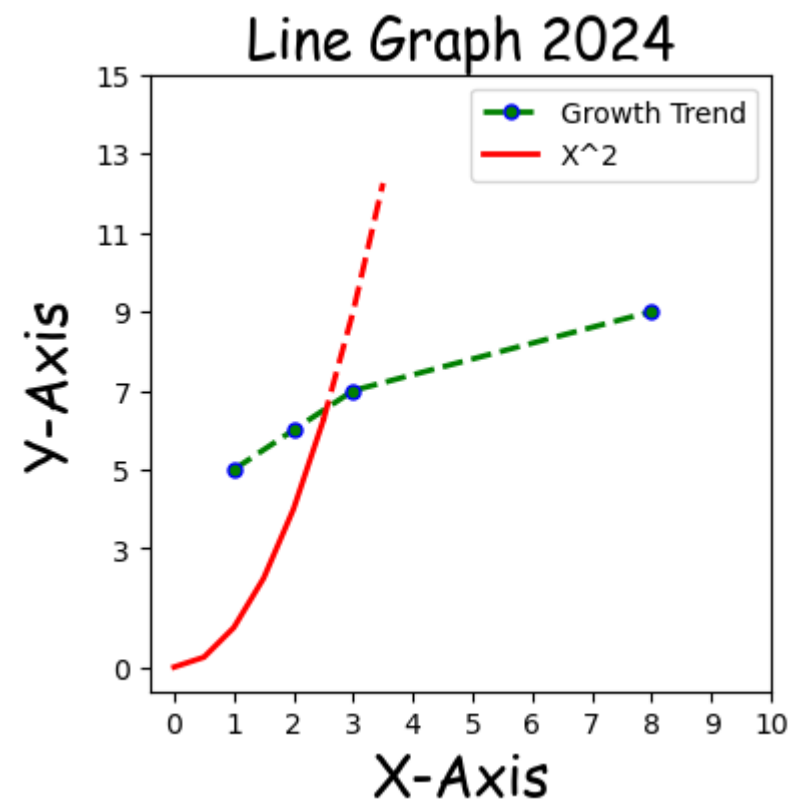
```
1 x=[1,2,3,8]
2 y=[5,6,7,9]
3
4 plt.plot(x,y,label="Growth Trend",color='green',linewidth=2,marker='.',linestyle='--', markersize=10,markeredgecolor='blue')
5
6 # Line Number 2 and parameters
7
8 x2=np.arange(0,4,0.5)
9 plt.plot(x2[:6],x2[:6]**2,label="X^2",color='red',linewidth=2)
10 plt.plot(x2[5:],x2[5:]**2,color='red',linewidth=2,linestyle='--')
11
12 plt.title("Line Graph 2024",fontdict={'fontname':'Comic Sans MS','fontsize':20})
13 plt.xlabel("X-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':20})
14 plt.ylabel("Y-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':20})
15 plt.xticks([0,1,2,3,4,5,6,7,8,9,10])
16 plt.yticks([0,3,5,7,9,11,13,15])
17
18 plt.legend()
19
20 plt.show()
```



Resizing your Graph

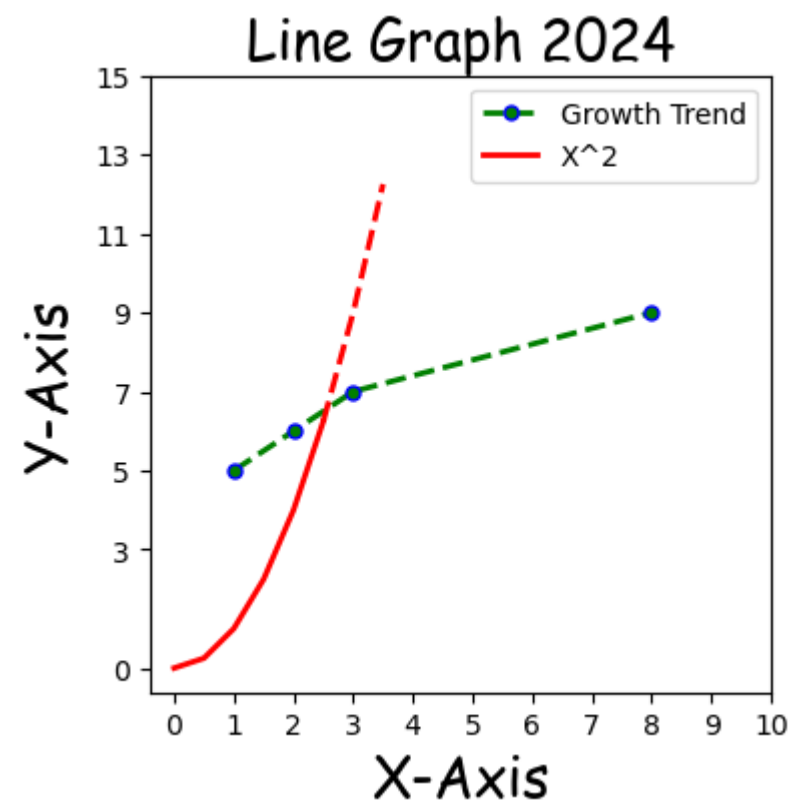
Parameters:figsize,dpi

```
In [29]: 1 x=[1,2,3,8]
2 y=[5,6,7,9]
3
4 # Resizing the Graph
5
6 plt.figure(figsize=(4,4),dpi=100) # Here the fig=figsize X dpi , 400 X400 .
7
8
9 plt.plot(x,y,label="Growth Trend",color='green',linewidth=2,marker='.',linestyle='--', markersize=10,markeredgecolor='blue')
10
11 # Line Number 2 and parameters
12
13 x2=np.arange(0,4,0.5)
14 plt.plot(x2[:6],x2[:6]**2,label="X^2",color='red',linewidth=2)
15 plt.plot(x2[5:],x2[5:]**2,color='red',linewidth=2,linestyle='--')
16
17 plt.title("Line Graph 2024",fontdict={'fontname':'Comic Sans MS','fontsize':20})
18 plt.xlabel("X-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':20})
19 plt.ylabel("Y-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':20})
20 plt.xticks([0,1,2,3,4,5,6,7,8,9,10])
21 plt.yticks([0,3,5,7,9,11,13,15])
22
23 plt.legend()
24
25 plt.show()
```



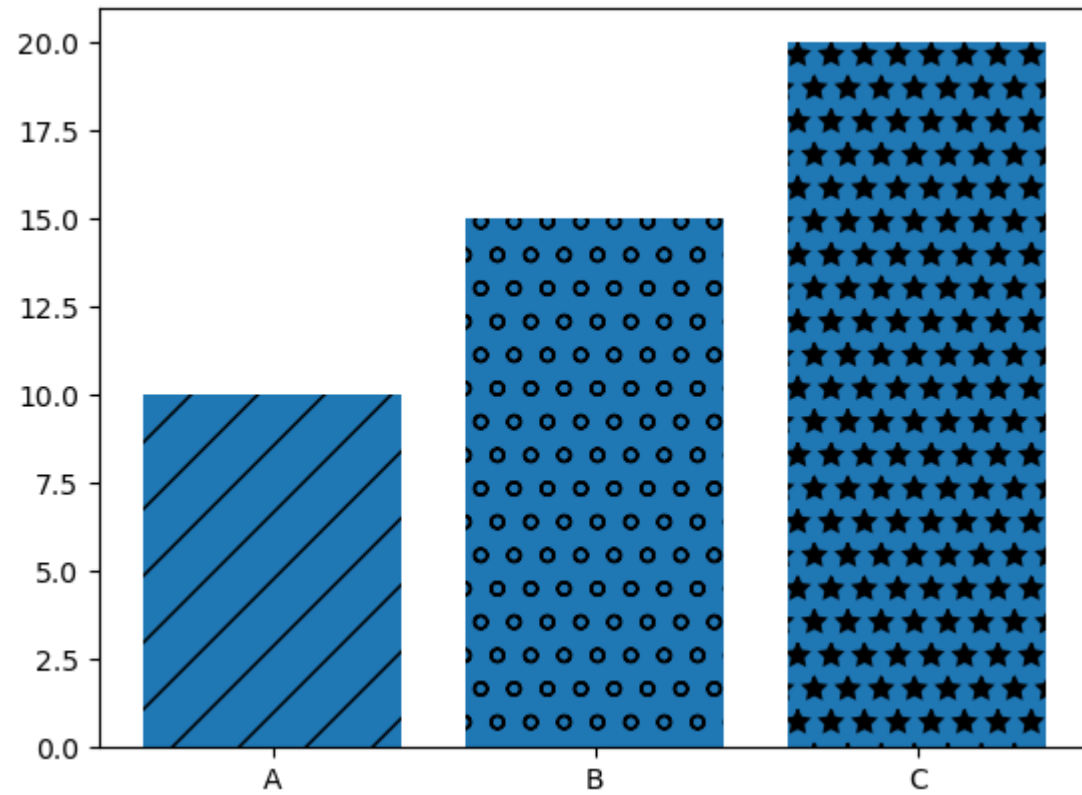
Saving the Graph into .png format : savefig

```
In [35]: 1 x=[1,2,3,8]
2 y=[5,6,7,9]
3
4
5 plt.figure(figsize=(4,4),dpi=100) # Here the fig=figsize X dpi , 400 X400 .
6
7 plt.plot(x,y,label="Growth Trend",color='green',linewidth=2,marker='.',linestyle='--', markersize=10,markeredgecolor='blue')
8
9 # Line Number 2 and parameters
10
11 x2=np.arange(0,4,0.5)
12 plt.plot(x2[:6],x2[:6]**2,label="X^2",color='red',linewidth=2)
13 plt.plot(x2[5:],x2[5:]**2,color='red',linewidth=2,linestyle='--')
14
15 plt.title("Line Graph 2024",fontdict={'fontname':'Comic Sans MS','fontsize':20})
16 plt.xlabel("X-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':20})
17 plt.ylabel("Y-Axis",fontdict={'fontname':'Comic Sans MS','fontsize':20})
18 plt.xticks([0,1,2,3,4,5,6,7,8,9,10])
19 plt.yticks([0,3,5,7,9,11,13,15])
20
21 plt.legend()
22
23 # saving the Graph
24 plt.savefig('line_graph.png',dpi=500)
25
26 plt.show()
```



Bar Chart

```
In [62]: 1 X=["A","B","C"]
2 Y=[10,15,20]
3
4 #Plotting the values
5
6 bars=plt.bar(X,Y)
7
8 #Giving Pattern of Hatch to the Bar
9
10 bars[0].set_hatch("/")
11 bars[1].set_hatch("o")
12 bars[2].set_hatch("*")
13
14 # Adjusting the figure size
15
16 plt.figure(figsize=(1,6))
17
18 #This command shows the graph
19 plt.show()
```



<Figure size 100x600 with 0 Axes>

Gas Dataset : Line Graph

Dataset: https://github.com/KeithGalli/matplotlib_tutorial/blob/master/gas_prices.csv (https://github.com/KeithGalli/matplotlib_tutorial/blob/master/gas_prices.csv)

In [84]:

1

gas=pd.read_csv("gas_prices.csv")

2

gas.tail(2)

Out[84]:

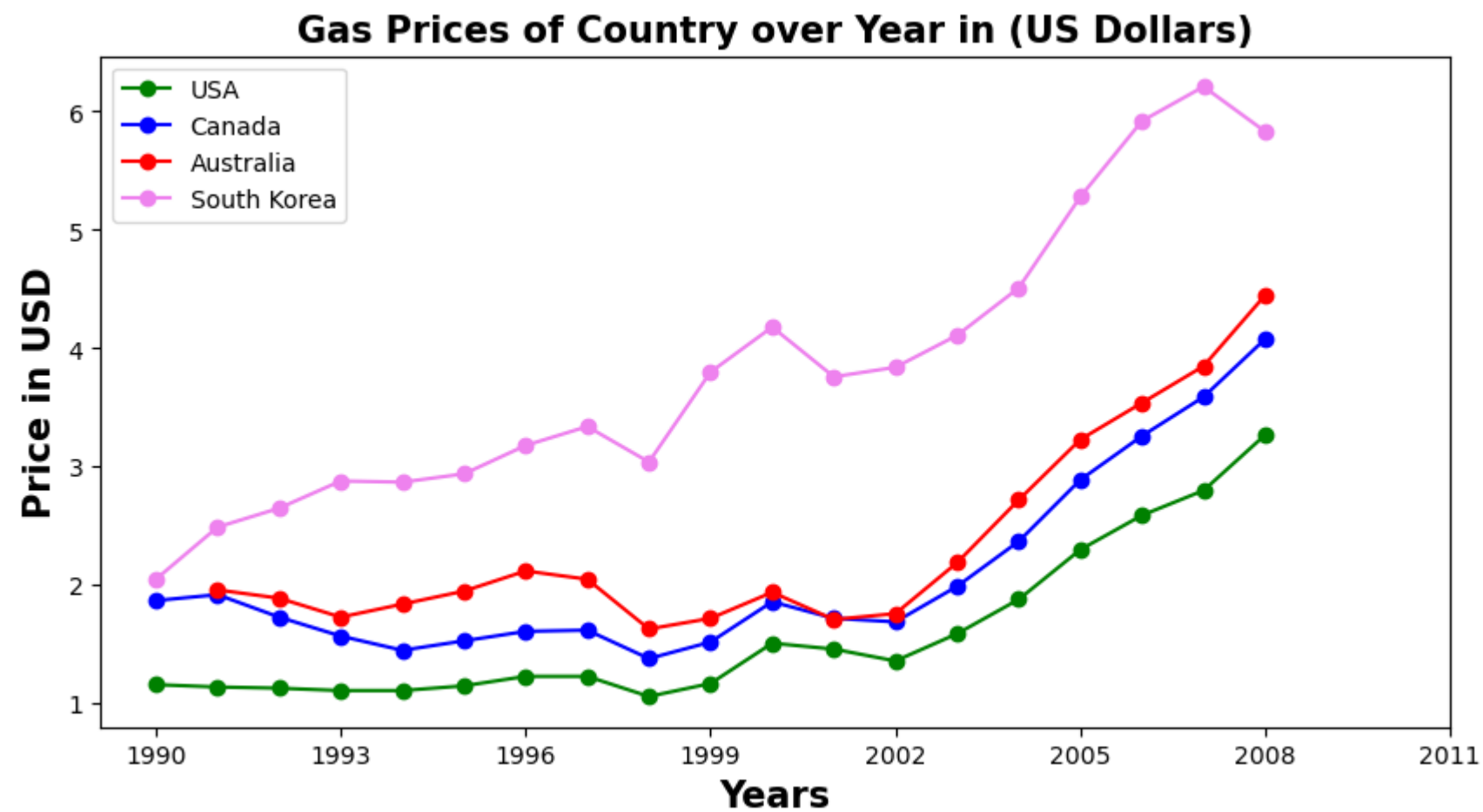
	Year	Australia	Canada	France	Germany	Italy	Japan	Mexico	South Korea	UK	USA
17	2007	3.85	3.59	6.60	6.88	6.73	4.49	2.40	6.21	7.13	2.80
18	2008	4.45	4.08	7.51	7.75	7.63	5.74	2.45	5.83	7.42	3.27

Line Graph

```

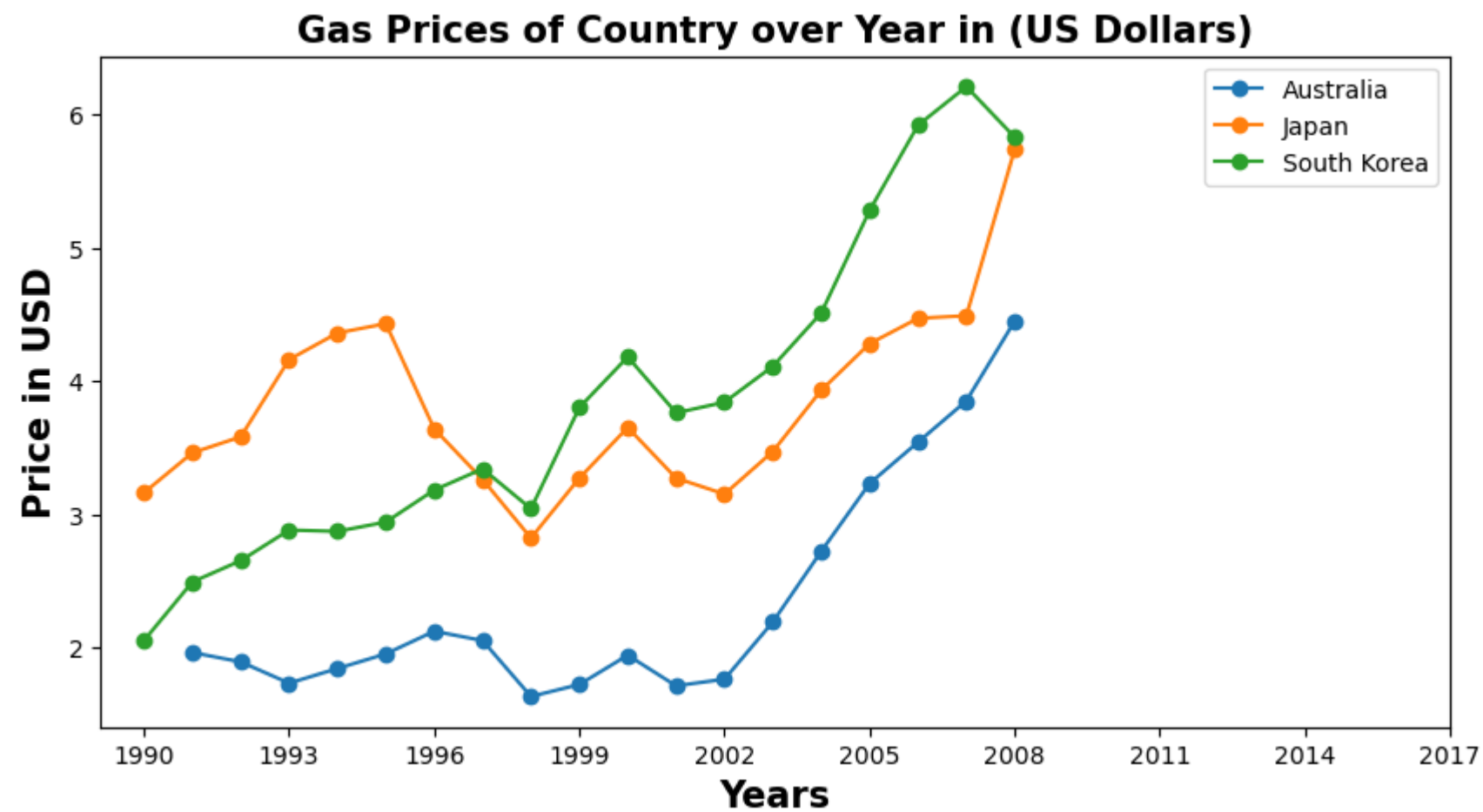
In [152]: 1 plt.figure(figsize=(10,5)) # This Scales the Size of Graph
2 plt.title("Gas Prices of Country over Year in (US Dollars)",fontdict={'fontweight':'bold','fontsize':15})
3
4 # Parameters to compare wrt different countries
5 plt.plot(gas.Year,gas.USA,label="USA",color="green",marker="o")
6 plt.plot(gas.Year,gas.Canada,label="Canada",color="blue",marker="o")
7 plt.plot(gas.Year,gas.Australia,label="Australia",color="red",marker="o")
8 plt.plot(gas.Year,gas['South Korea'],label="South Korea",color="violet",marker="o")
9
10
11 # This will plot the X-Axis years data of evry 3 years
12 plt.xticks(gas.Year[::3].tolist() + [2011]) # This will add 2011 in the X-axis
13
14 # Labelling the X and Y axis of the Graph
15 plt.xlabel("Years",fontdict={'fontweight':'bold','fontsize':15})
16 plt.ylabel("Price in USD",fontdict={'fontweight':'bold','fontsize':15})
17
18 plt.legend() # Use this command to display the labelsn or else it wont show
19
20 plt.savefig("gas_graph.png",dpi=300) ## Save this command before plt.show() or else nothing will be displayed in img
21
22 plt.show()
23
24

```



Graphing using the for loop

```
In [155]: 1 plt.figure(figsize=(10,5)) # This Scales the Size of Graph
2 plt.title("Gas Prices of Country over Year in (US Dollars)",fontdict={'fontweight':'bold','fontsize':15}) # This gives Title to your Graph
3
4 # Method 2 to check using the For Loop
5 countries_to_look_at=["Australia","Japan","South Korea"]
6 for country in gas :
7     if country in countries_to_look_at:
8         plt.plot(gas.Year,gas[country],marker='o',label=(country))
9
10
11 # This will plot the X-Axis years data of evry 3 years
12 plt.xticks(gas.Year[::3].tolist()+[2011,2014,2017]) # This tolist() function will add years in X-axis
13
14 # Labelling the X and Y axis of the Graph
15 plt.xlabel("Years",fontdict={'fontweight':'bold','fontsize':15})
16 plt.ylabel("Price in USD",fontdict={'fontweight':'bold','fontsize':15})
17
18 plt.legend() # Use this command to display the labelsn or else it wont show
19
20 plt.savefig("gas_graph2.png",dpi=200) ## Save this command before plt.show() or else nothing will be displayed in img
21
22 plt.show()
```



Fifa Dataset : Histogram

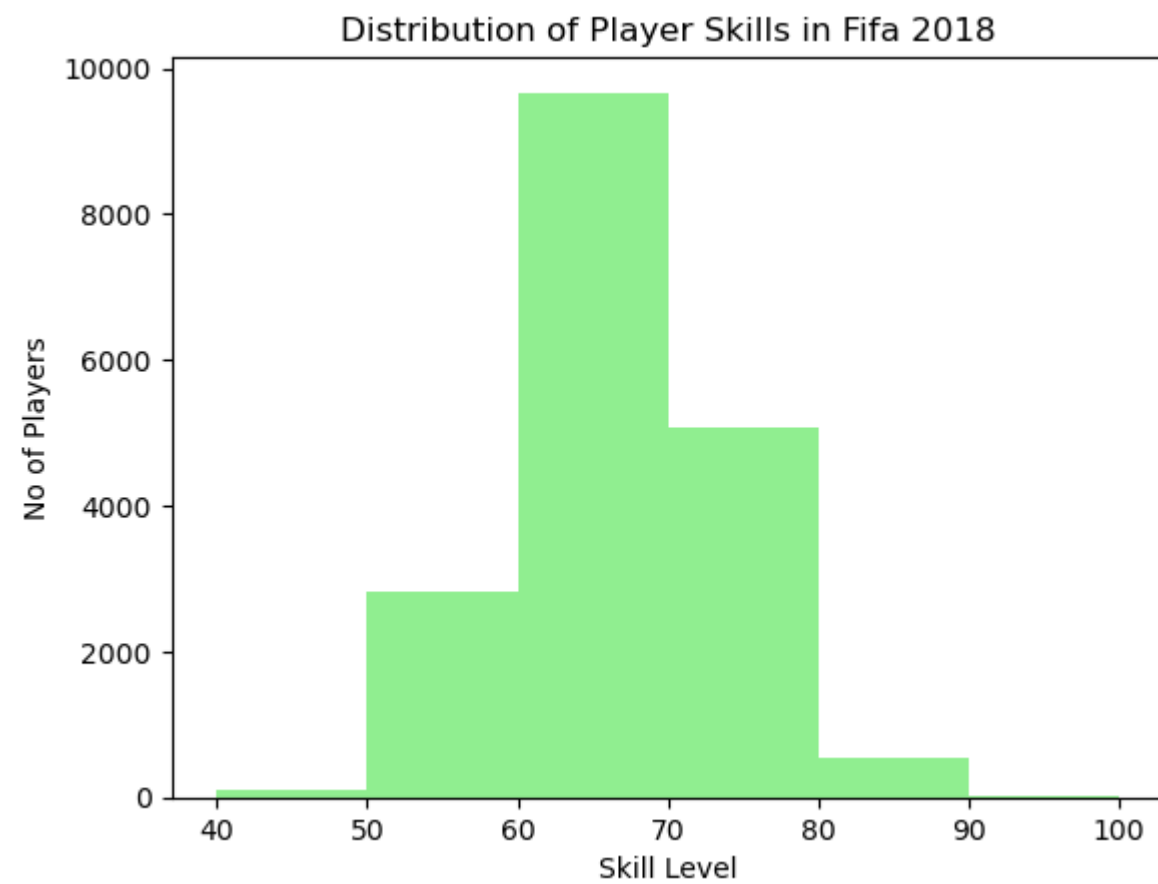
Dataset: https://github.com/KeithGalli/matplotlib_tutorial/blob/master/fifa_data.csv
(https://github.com/KeithGalli/matplotlib_tutorial/blob/master/fifa_data.csv)

```
In [181]: 1 fifa=pd.read_csv("fifa_data.csv")
          2 fifa.shape
```

Out[181]: (18207, 89)

Histograms

```
In [237]: 1 bins=[40,50,60,70,80,90,100]
          2 plt.hist(fifa.Overall,bins=bins,color="lightgreen")
          3
          4 plt.title("Distribution of Player Skills in Fifa 2018")
          5 plt.xlabel("Skill Level")
          6 plt.ylabel("No of Players")
          7 plt.xticks(bins)
          8
          9 plt.show()
```



Pie charts

In [192]:

1fifa.head(2)

Out[192]:

	Unnamed: 0	ID	Name	Age	Photo	Nationality	Flag	Overall	Potential	Club	...	Composure	Marking	StandingTackle	SlidingTackle	GKDividing	GKI
0	0	158023	L. Messi	31	https://cdn.sofifa.org/players/4/19/158023.png	Argentina	https://cdn.sofifa.org/flags/52.png	94	94	FC Barcelona	...	96.0	33.0	28.0	26.0	6.0	
1	1	20801	Cristiano Ronaldo	33	https://cdn.sofifa.org/players/4/19/20801.png	Portugal	https://cdn.sofifa.org/flags/38.png	94	94	Juventus	...	95.0	28.0	31.0	23.0	7.0	

2 rows × 89 columns

In [230]:

1fifa.columns

Out[230]:

Index(['Unnamed: 0', 'ID', 'Name', 'Age', 'Photo', 'Nationality', 'Flag', 'Overall', 'Potential', 'Club', 'Club Logo', 'Value', 'Wage', 'Special', 'Preferred Foot', 'International Reputation', 'Weak Foot', 'Skill Moves', 'Work Rate', 'Body Type', 'Real Face', 'Position', 'Jersey Number', 'Joined', 'Loaned From', 'Contract Valid Until', 'Height', 'Weight', 'LS', 'ST', 'RS', 'LW', 'LF', 'CF', 'RF', 'RW', 'LAM', 'CAM', 'RAM', 'LM', 'LCM', 'CM', 'RCM', 'RM', 'LWB', 'LDM', 'CDM', 'RDM', 'RWB', 'LB', 'LCB', 'CB', 'RCB', 'RB', 'Crossing', 'Finishing', 'HeadingAccuracy', 'ShortPassing', 'Volleys', 'Dribbling', 'Curve', 'FKAccuracy', 'LongPassing', 'BallControl', 'Acceleration', 'SprintSpeed', 'Agility', 'Reactions', 'Balance', 'ShotPower', 'Jumping', 'Stamina', 'Strength', 'LongShots', 'Aggression', 'Interceptions', 'Positioning', 'Vision', 'Penalties', 'Composure', 'Marking', 'StandingTackle', 'SlidingTackle', 'GKDividing', 'GKHandling', 'GKKicking', 'GKPositioning', 'GKReflexes', 'Release Clause'], dtype='object')

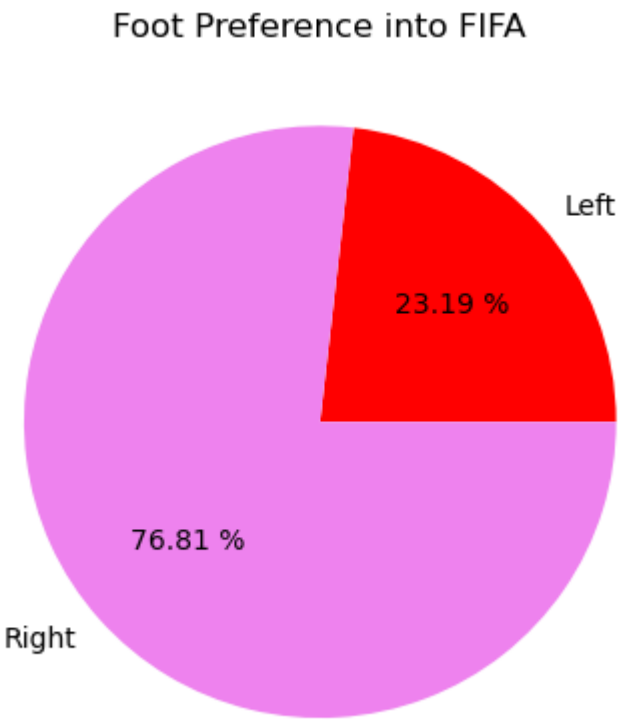
In [234]:

1fifa['Preferred Foot'].value_counts()

Out[234]:

Right 13948
Left 4211
Name: Preferred Foot, dtype: int64


```
In [274]: 1 left=fifa.loc[fifa['Preferred Foot']=="Left"].count()[0] # This gives count of Left Foot
2 right=fifa.loc[fifa['Preferred Foot']=="Right"].count()[0] # This gives count of Righth Foot
3
4 label=["Left","Right"]
5 color=["Red","Violet"]
6
7 # Plotting it over the Pie chart
8 plt.pie([left,right],labels=label,colors=color,autopct="%.2f %%")
9
10 plt.title("Foot Preference into FIFA")
11
12
13 plt.show()
14
15
```



```
In [ ]: 1
In [ ]: 1
In [ ]: 1
In [ ]: 1
```