matplotlib

In [1]:

```
# importing numpy as np and matplotlib as plt
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:

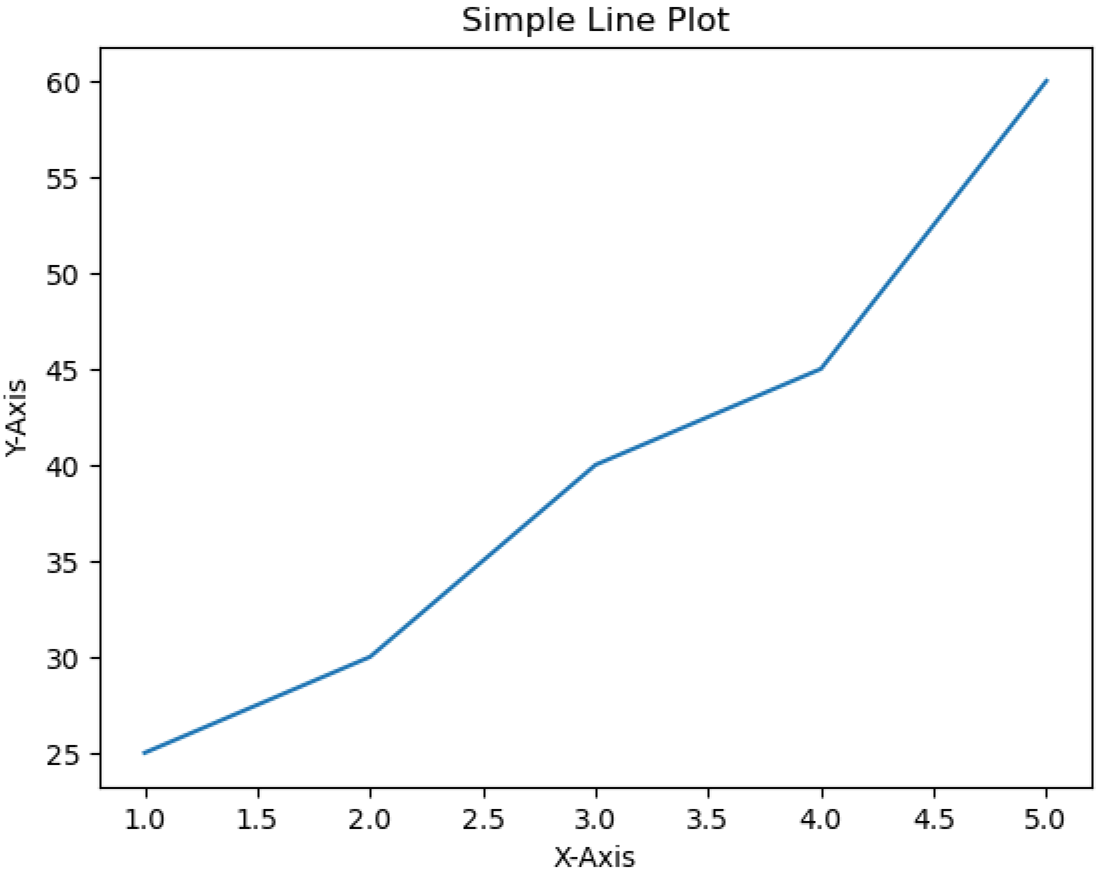
```
import warnings
#suppressing warnings
warnings.filterwarnings('ignore')
```

In [3]:

```
#It will output static images of thge plot embedded in the notebook
%matplotlib inline
```

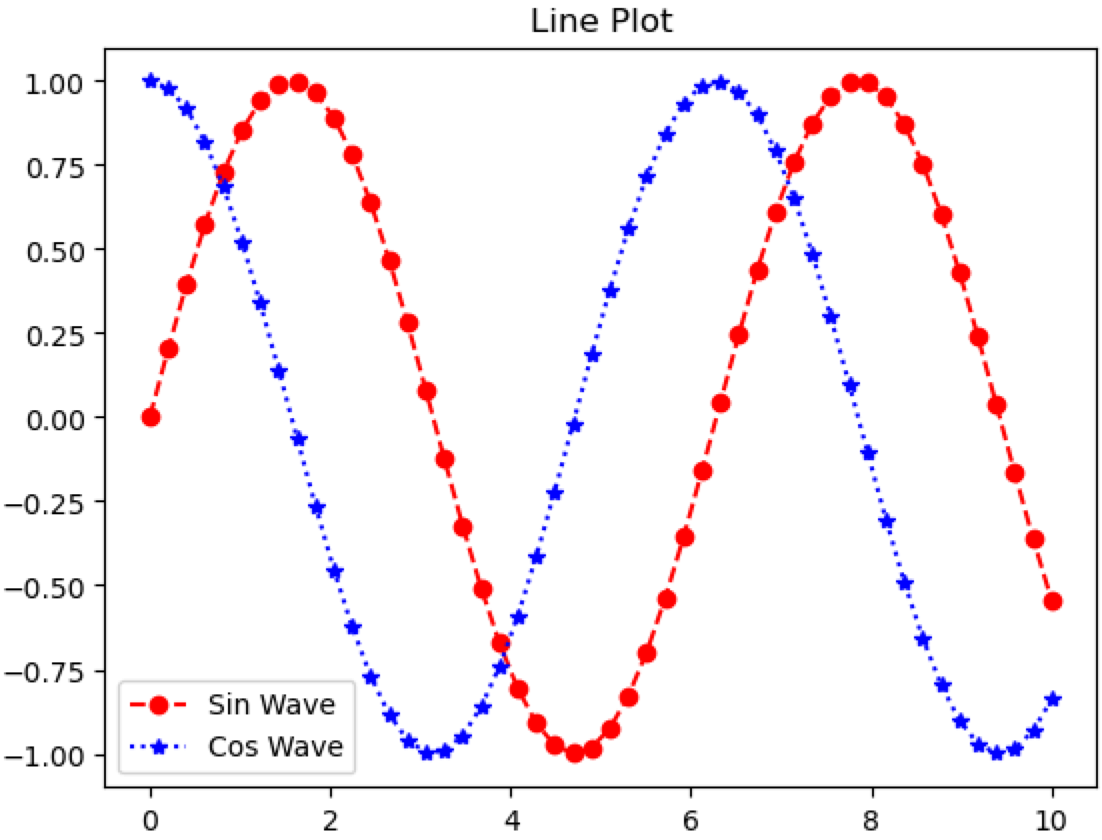
In [4]:

```
x=[1,2,3,4,5]
y=[25,30,40,45,60]
plt.plot(x,y)   # Line Plot
plt.title('Simple Line Plot')
plt.xlabel('X-Axis')
plt.ylabel('Y-Axis')
plt.show()
```



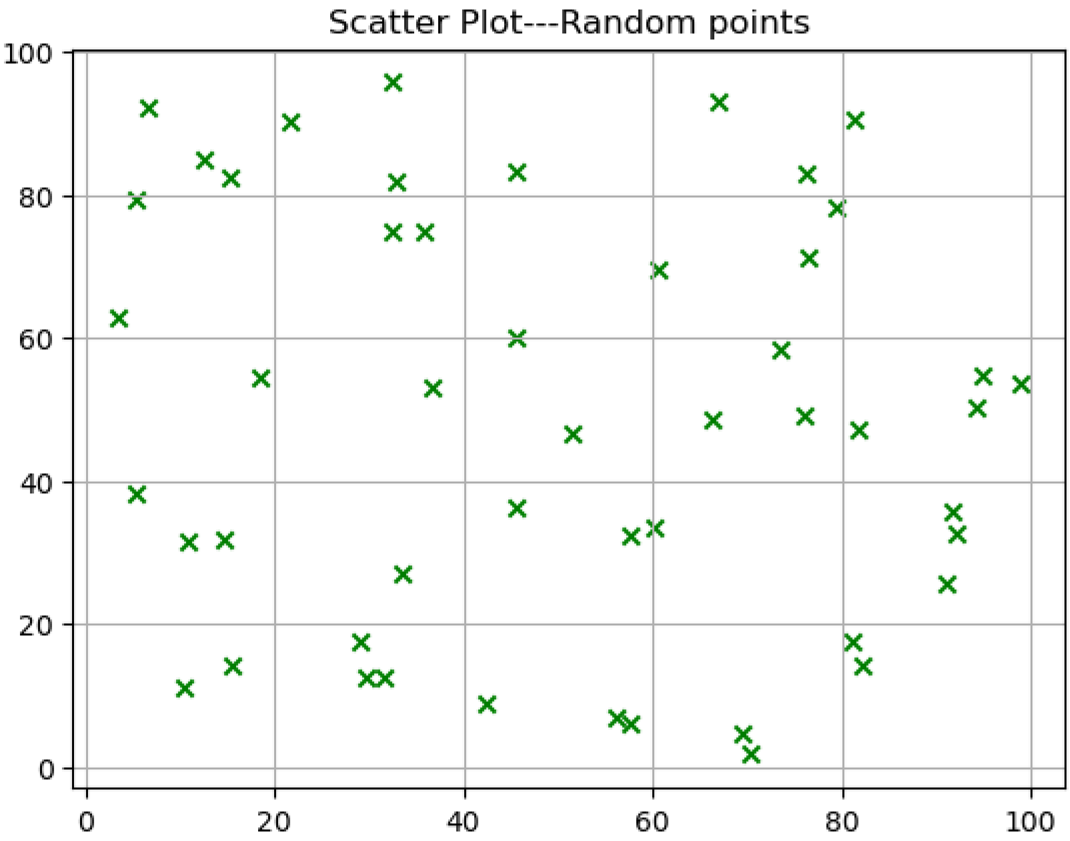
In [5]:

```
x1=np.linspace(0,10,50)
plt.plot(x1,np.sin(x1),color='red',ls='--',marker='o', label='Sin Wave')
plt.plot(x1,np.cos(x1),color='blue',ls=':',marker='*', label='Cos Wave')
plt.title('Line Plot')
plt.legend()
plt.show()
```

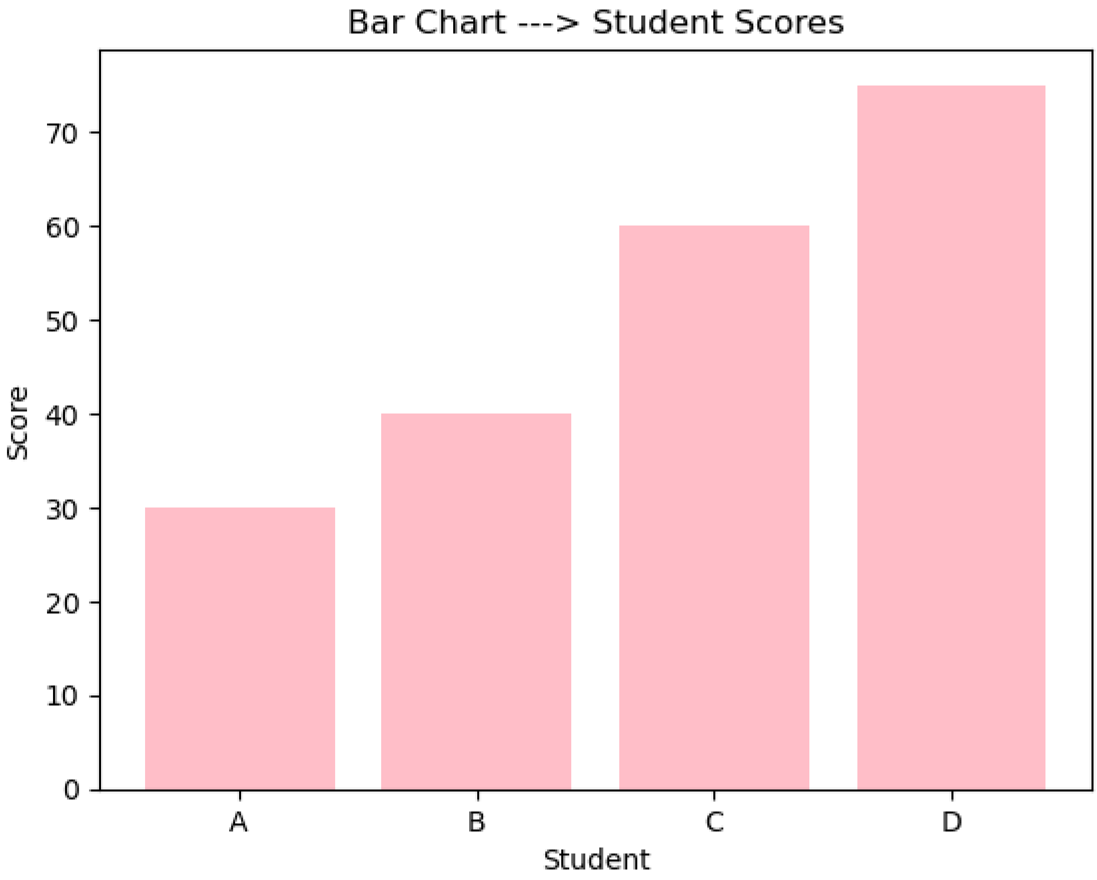


In [6]:

```
x2=np.random.random(50)*100
y2=np.random.random(50)*100
plt.scatter(x2,y2,color='green',marker='x')
plt.title('Scatter Plot---Random points')
plt.grid(True)
plt.show()
```

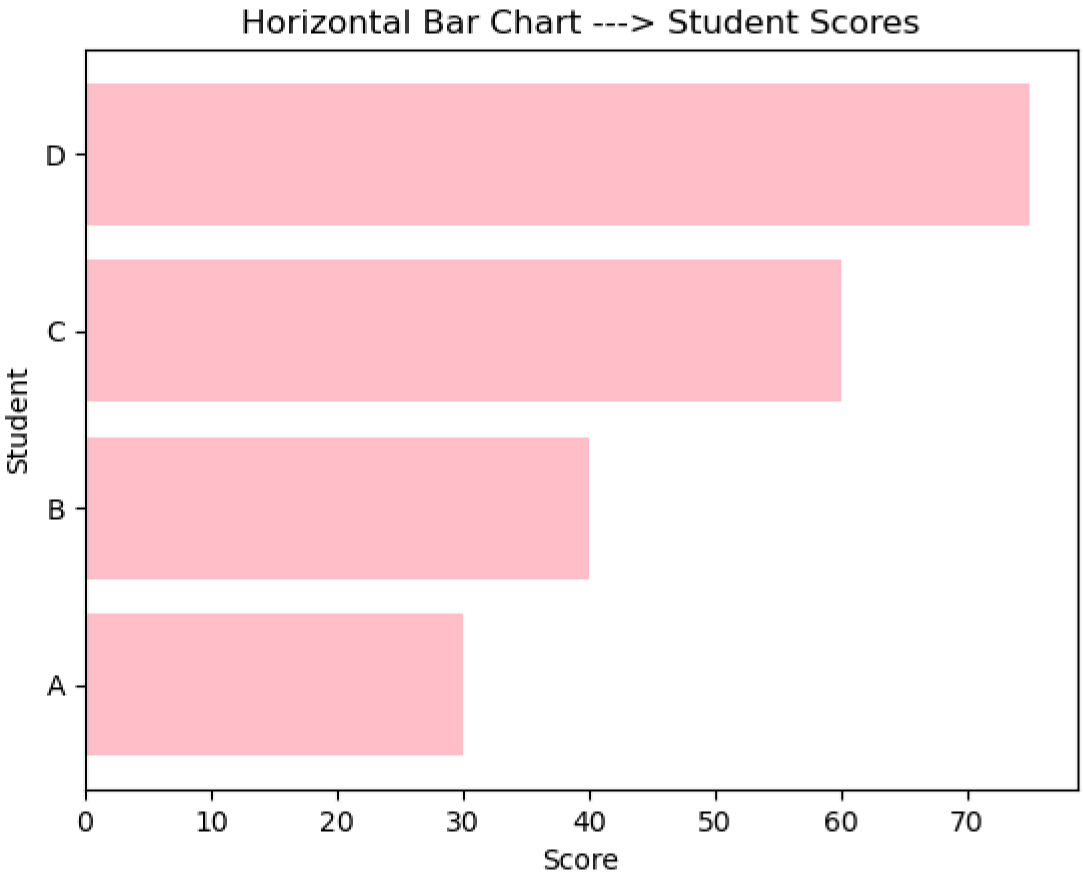


In [7]:
```
names=['A','B','C','D']
scores=[30,40,60,75]
plt.bar(names,scores,color='pink')
plt.title('Bar Chart ---> Student Scores')
plt.xlabel('Student')
plt.ylabel('Score')
plt.show()
```



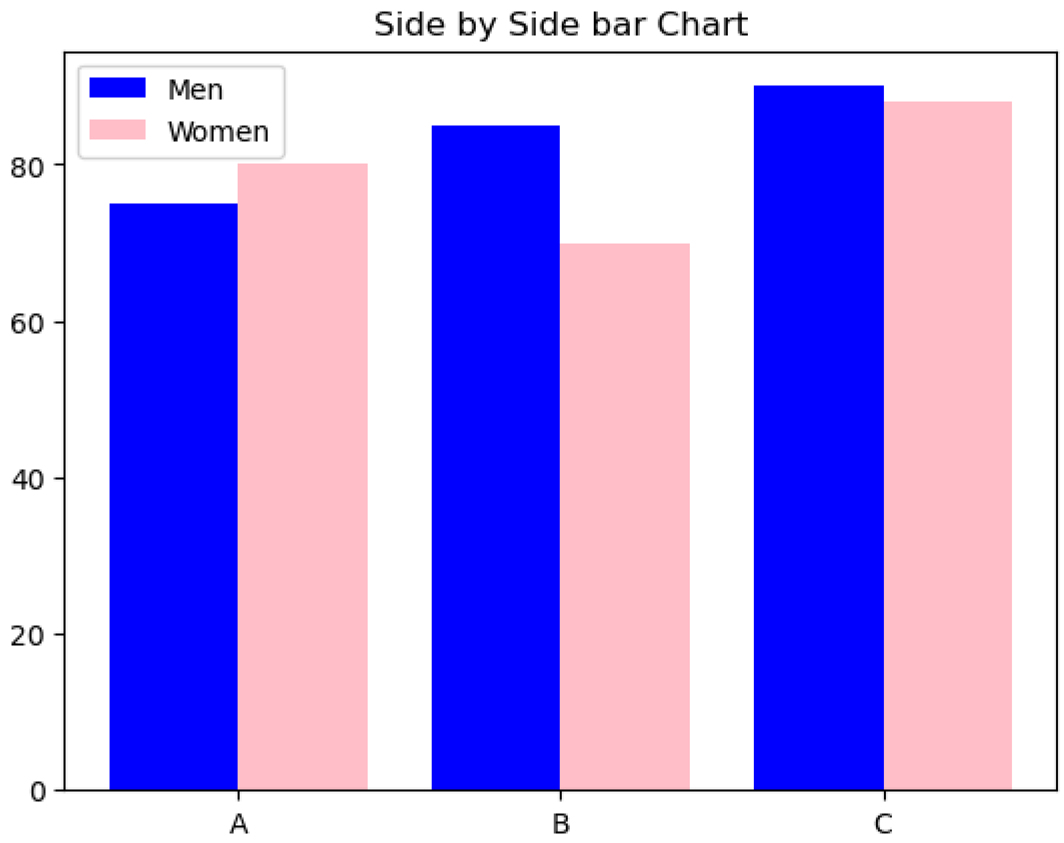
In [8]:

```
plt.barh(names,scores,color='pink')
plt.title('Horizontal Bar Chart ---> Student Scores')
plt.xlabel('Score')
plt.ylabel('Student')
plt.show()
```



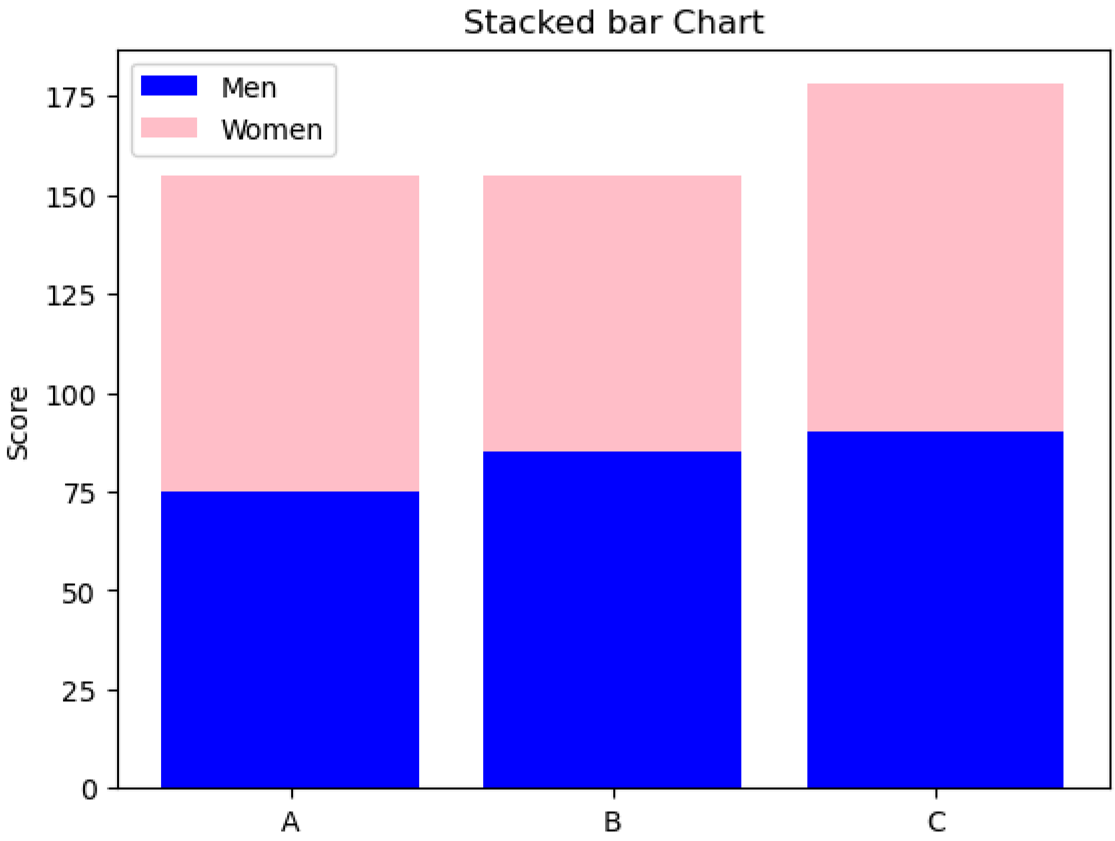
In [9]:

```
labels=['A','B','C']
men = [75, 85, 90]
women = [80, 70, 88]

x=np.arange(len(labels))
width=0.4

plt.bar(x-width/2,men,width, color='blue',label='Men')
plt.bar(x+width/2,women,width, color='pink',label='Women')
plt.title('Side by Side bar Chart')
plt.xticks(x,labels)
plt.legend()
plt.show()
```



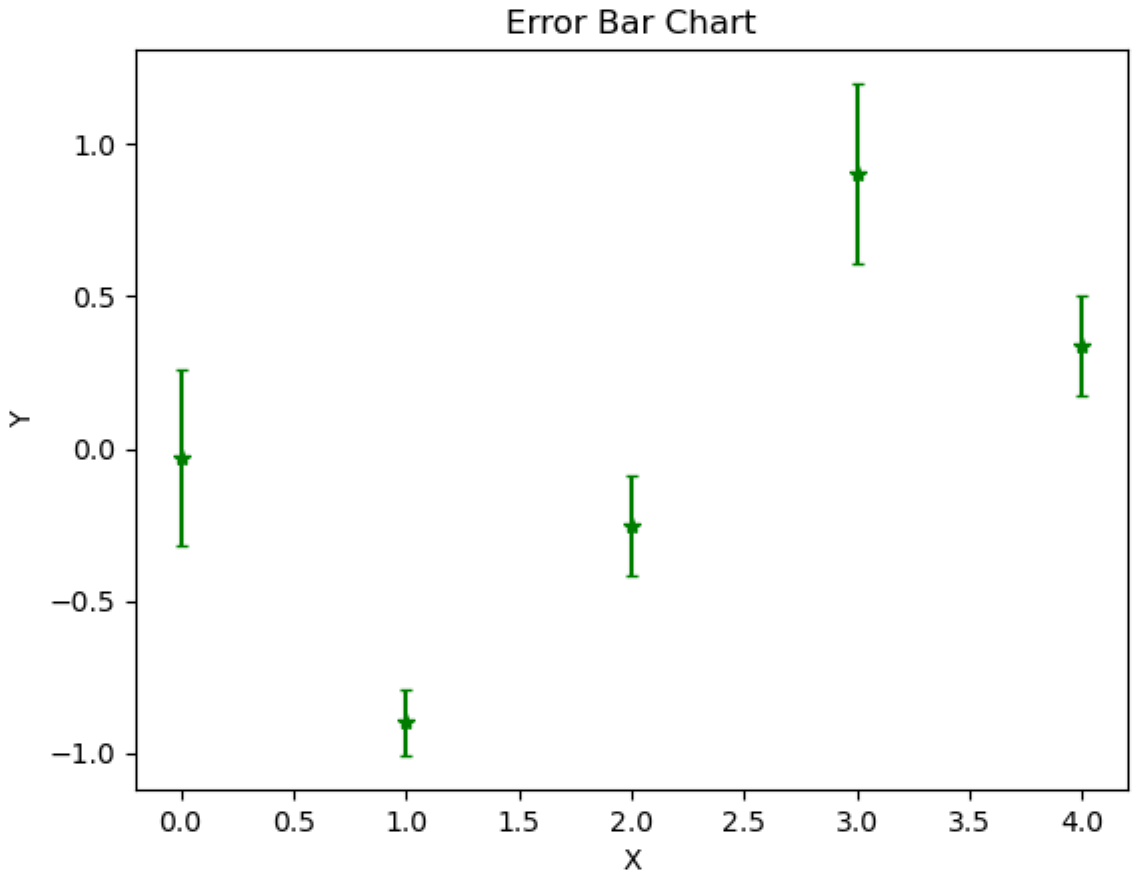
In [10]:

```
labels=['A','B','C']
men = [75, 85, 90]
women = [80, 70, 88]

x=np.arange(len(labels))


plt.bar(x,men, color='blue',label='Men')
plt.bar(x,women, bottom=men, color='pink',label='Women')
plt.title('Stacked bar Chart')
plt.xticks(x,labels)
plt.ylabel('Score')
plt.legend()
plt.show()
```



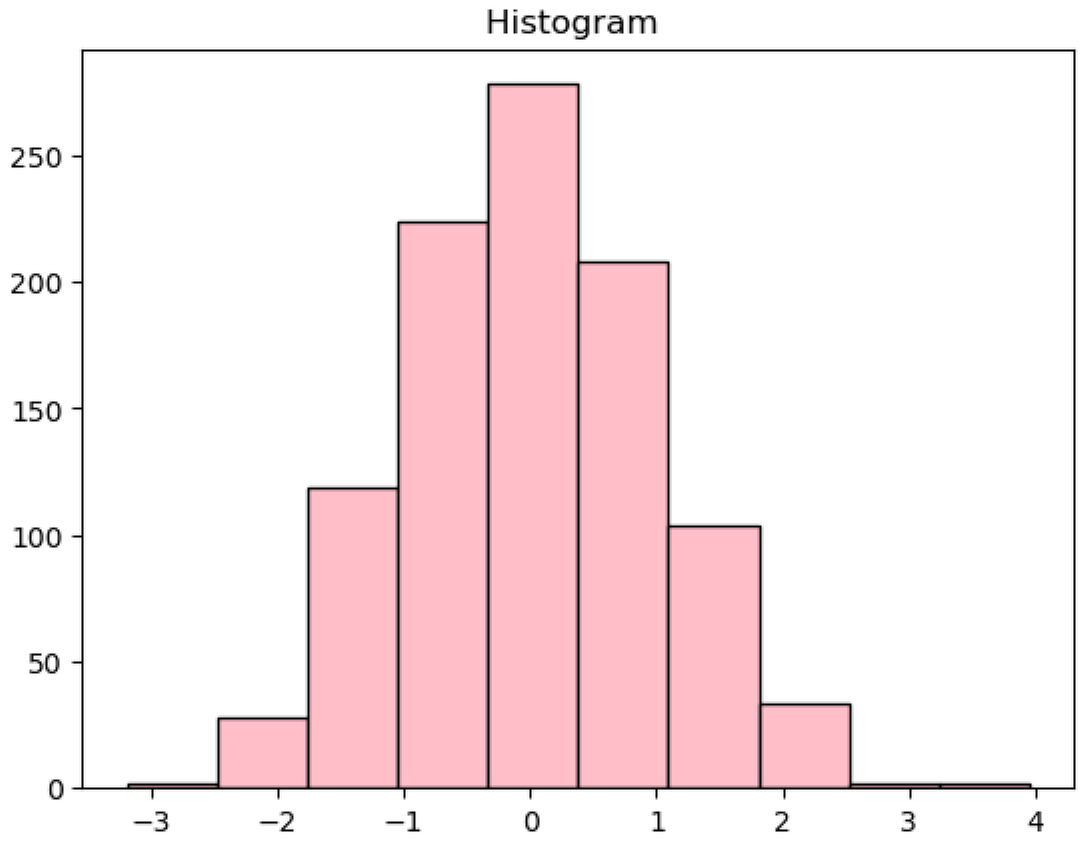
In [11]:

```
x = np.arange(5)
y = np.random.normal(0, 1, 5)
errors = np.random.rand(5) * 0.3
plt.errorbar(x, y, yerr=errors, fmt='*', capsize=2, color='green')
plt.title("Error Bar Chart")
plt.xlabel("X")
plt.ylabel("Y")
plt.show()
#  Shows uncertainty/error margins for data point
```

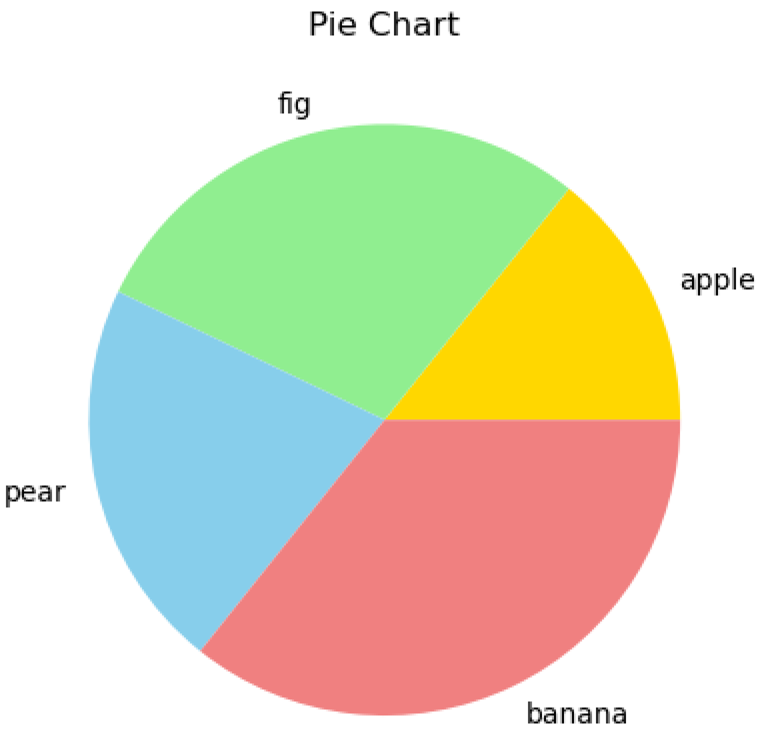


In [12]:

```
data = np.random.randn(1000)
plt.hist(data, color='pink', edgecolor='black')
plt.title("Histogram ")
plt.show()
```

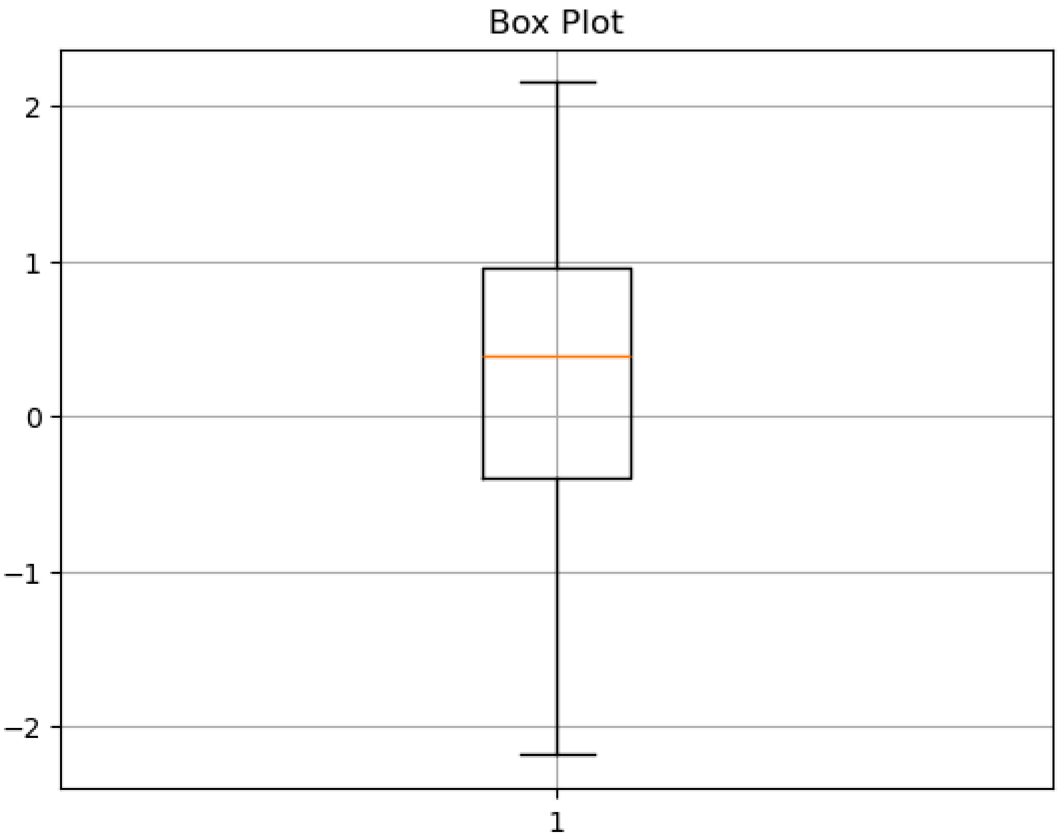


In [13]:
```
labels=['apple','fig','pear','banana']
value=[10,20,15,25]
colors = ['gold', 'lightgreen', 'skyblue', 'lightcoral']
plt.pie(value, labels=labels, colors=colors)
plt.title("Pie Chart")
plt.show()
```

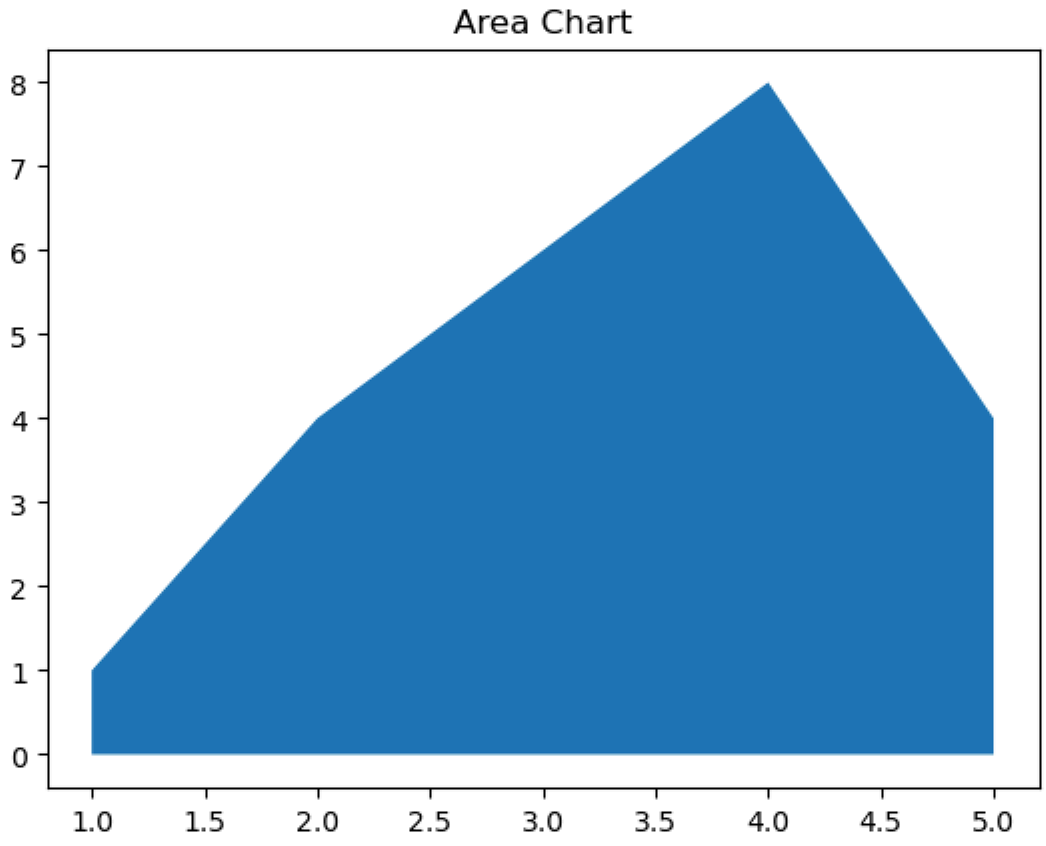


In [14]:

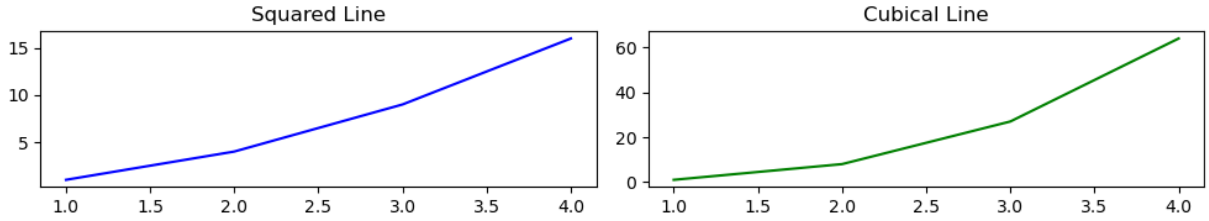
```
data = np.random.randn(100)
plt.boxplot(data)
plt.title("Box Plot")
plt.grid(True)
plt.show()
```



```
In [15]: x=range(1,6)
         y=[1,4,6,8,4]
         plt.fill_between(x,y)
         plt.title('Area Chart')
         plt.show()
```



In [16]:
```
x = [1, 2, 3, 4]
y1 = [1, 4, 9, 16]
y2 = [1, 8, 27, 64]
plt.figure(figsize=(10, 2))
plt.subplot(1, 2, 1)
plt.plot(x, y1, color='blue')
plt.title("Squared Line")
plt.subplot(1, 2, 2)
plt.plot(x, y2, color='green')
plt.title("Cubical Line")
plt.tight_layout()
plt.show()
```



In [ ]: