Lab-3_Part_1

Matplotlib Exercises

- Name: Daniel Mehta
- Student ID: n01753264

Import Numpy, Panda and Matplotlib library

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:
```
import numpy as np
x = np.arange(0,50)
y = x*2
z = x**2
```

Question 1

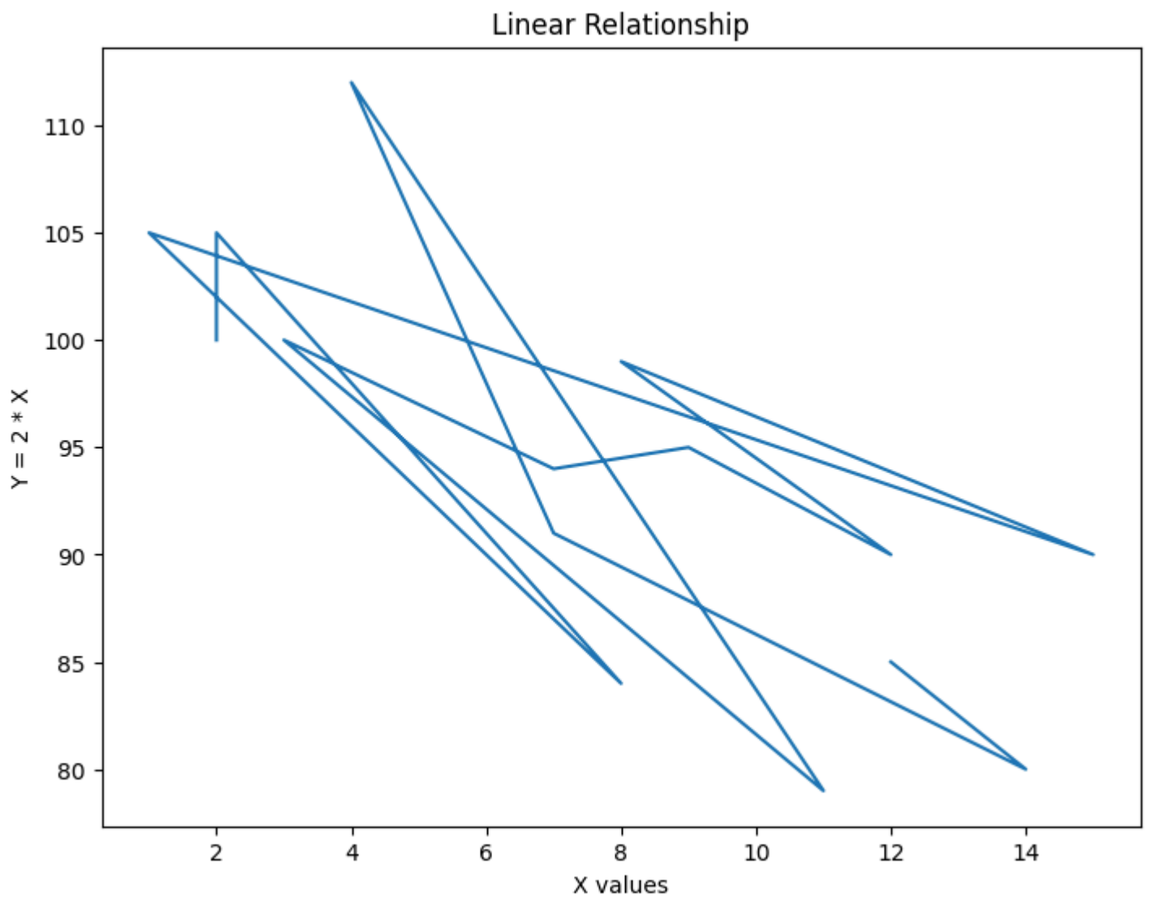
Follow steps:

- Create a figure object called fig using plt.figure() **
- Use add_axes to add an axis to the figure canvas at [0,0,1,1]. Call this new axis ax.
- Plot (x,y) on that axes and set the labels and titles to match the plot below:**

In [7]:
```
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])
ax.plot(x,y)

ax.set_title("Linear Relationship")
ax.set_xlabel("X values")
ax.set_ylabel("Y = 2 * X")
```

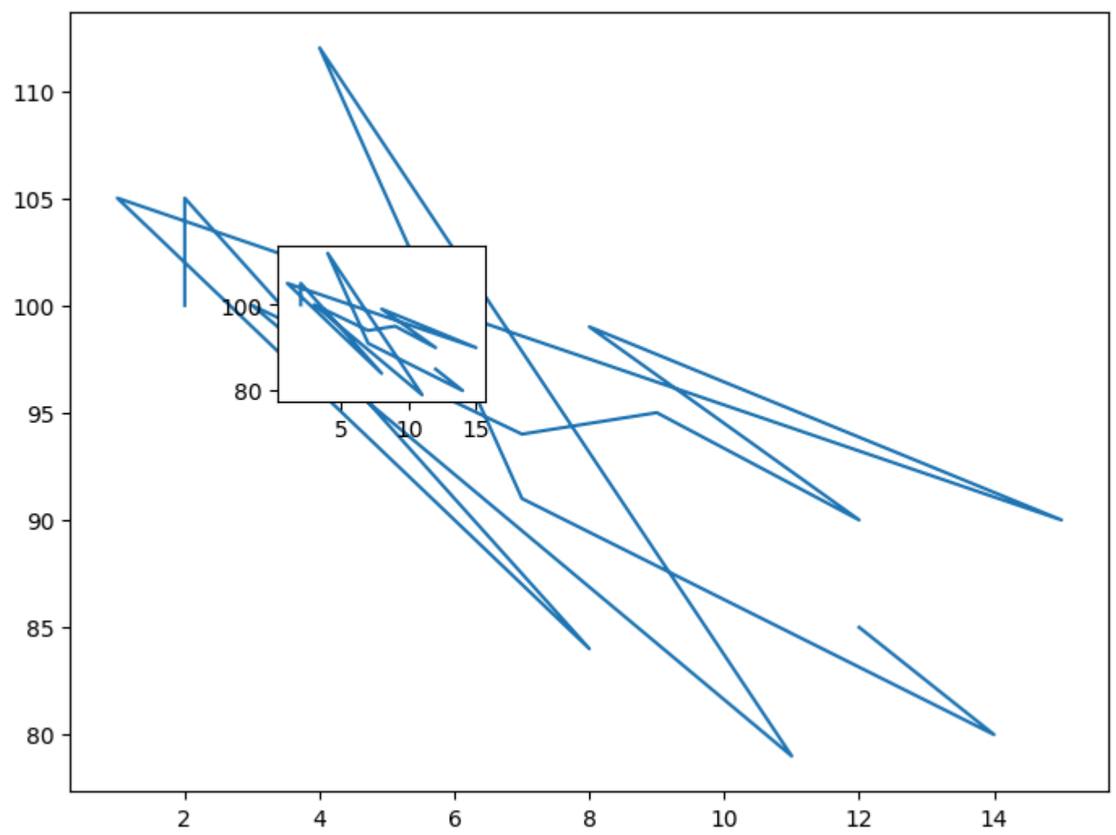
Out[7]: Text(0, 0.5, 'Y = 2 * X')



Question 2

- Create a figure object and put two axes on it, ax1 and ax2. Located at [0,0,1,1] and [0.2,0.5,.2,.2] respectively.
- plot (x,y) on both axes. And call your figure object to show it.

In [13]:
```
fig2 = plt.figure()
ax1 = fig2.add_axes([0,0,1,1])
ax2 = fig2.add_axes([0.2,0.5,.2,.2])
ax1.plot(x,y)
ax2.plot(x,y)
plt.show()
```



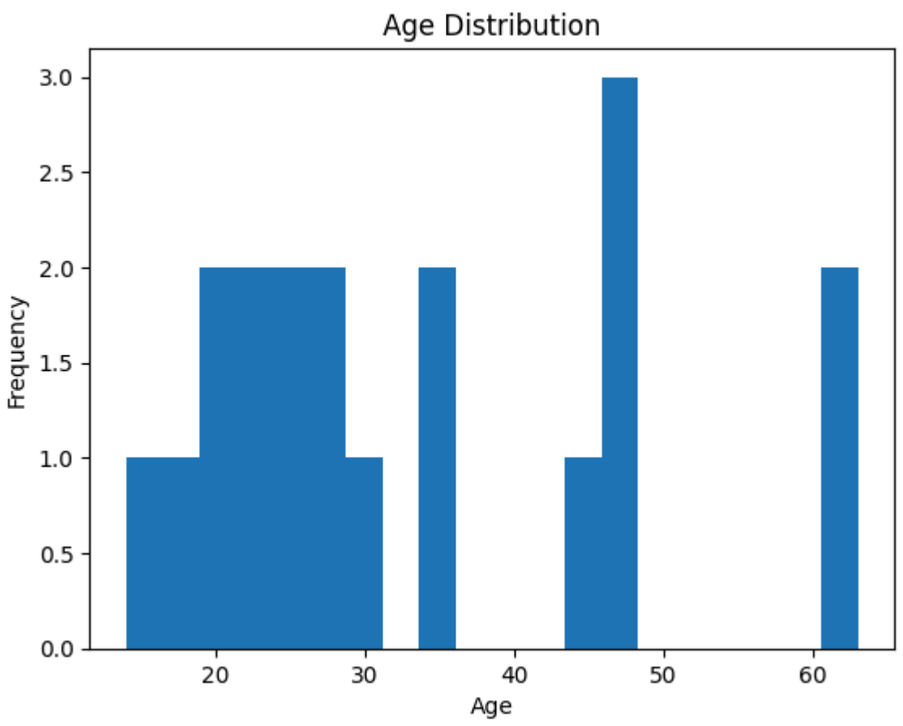
Question 3

- Read the dataset Titanic, create the dataframe and read all columns.
- Plot the Age column information

- Plot all columns information

In [25]:
```
df = pd.read_csv("Titanic_1.csv")
df.columns
#titianic_fig = plt.figure()

df['Age'].plot(kind='hist', bins=20, title='Age Distribution')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.show()
```



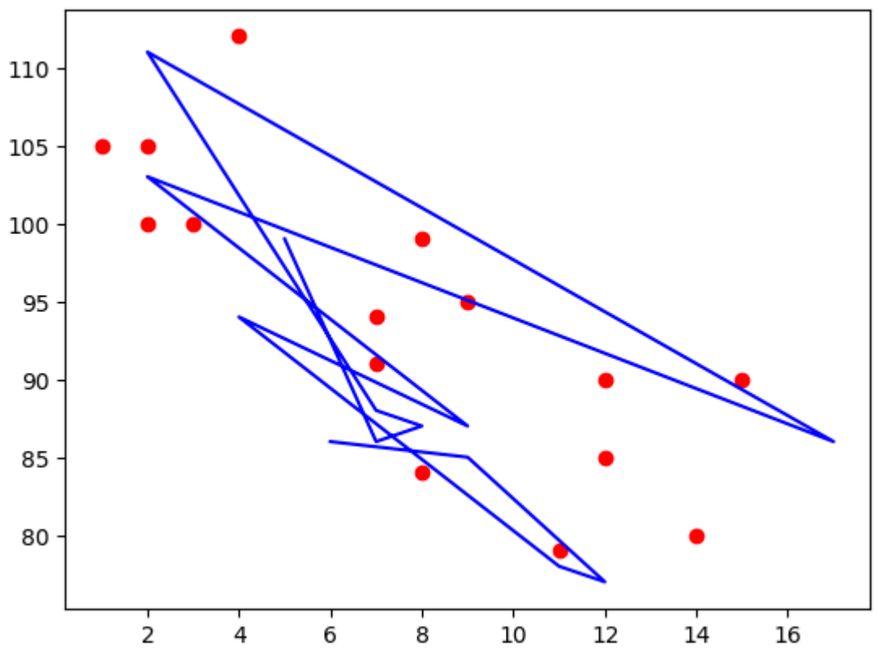
Question 4

Plot the array bellow with different line and scatterplot colors.

In [28]:
```
import matplotlib.pyplot as plt
import numpy as np

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
plt.plot(x, y, label='Line Plot', color='blue')


x = np.array([2,2,8,1,15,8,12,9,7,3,11,4,7,14,12])
y = np.array([100,105,84,105,90,99,90,95,94,100,79,112,91,80,85])
plt.scatter(x, y, label='Scatter Plot', color='Red')
plt.show()
```



Question 5

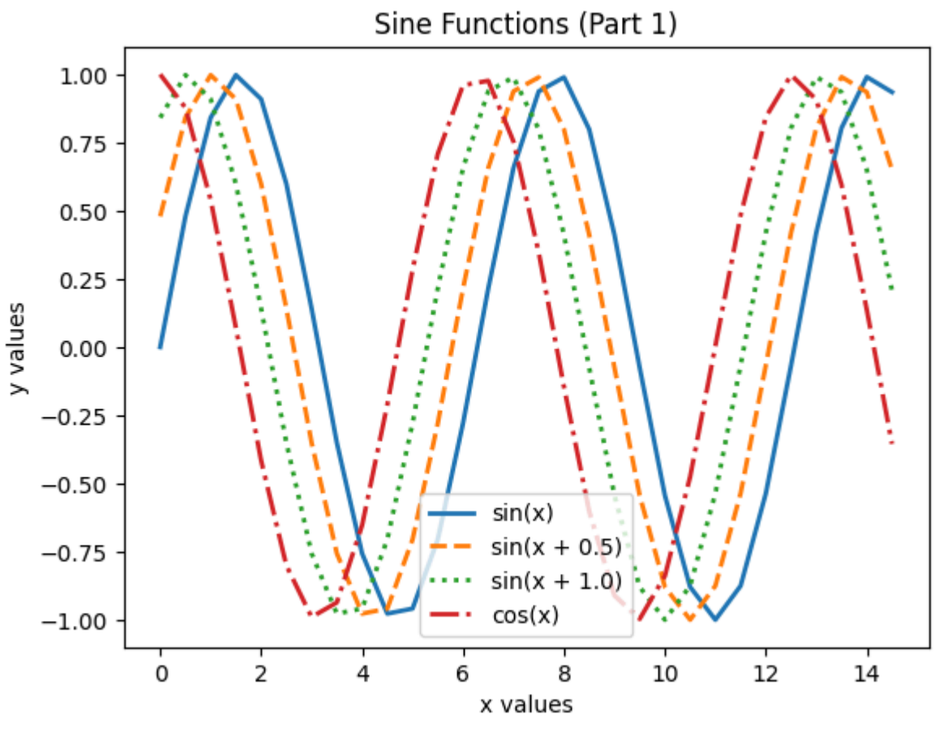
Consider the x=np.arange(0,15,0.5), them plot (np.sin(x)), (np.sin(x+0.5)), (np.sin(x+1.0)), (np.cos(x)) with different linestyle and linewidth.

In [30]:
```
x=np.arange(0,15,0.5)

plt.plot(x, np.sin(x), label='sin(x)', linestyle='-', linewidth=2)
plt.plot(x, np.sin(x + 0.5), label='sin(x + 0.5)', linestyle='--', linewidth=2)
plt.plot(x, np.sin(x + 1.0), label='sin(x + 1.0)', linestyle=':', linewidth=2)
plt.plot(x, np.cos(x), label='cos(x)', linestyle='-.', linewidth=2)

plt.title('Sine Functions (Part 1)')
plt.xlabel('x values')
plt.ylabel('y values')
plt.legend()

plt.show()
```



In [ ]: