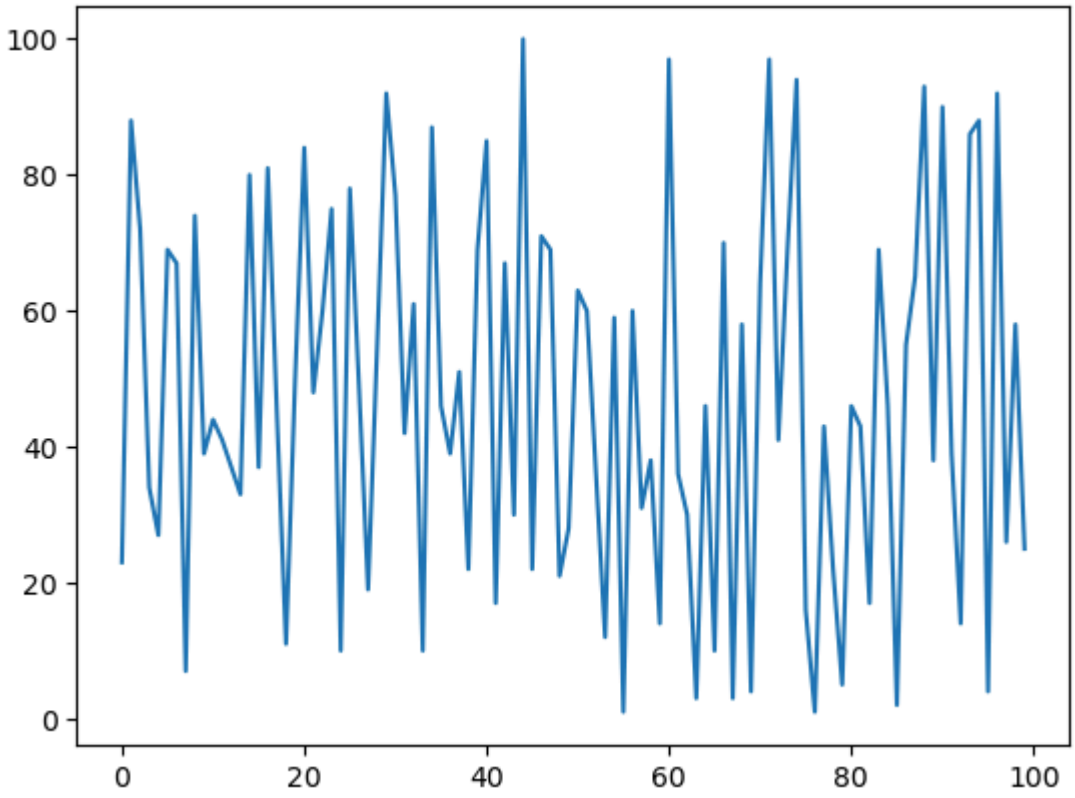
In [1]:
```
student_name = "Yasmin Pokia"  # fill your name
student_id = "s222245206"  # fill your student ID
print("Student name: " + student_name)
print("Student ID: " + student_id)
```

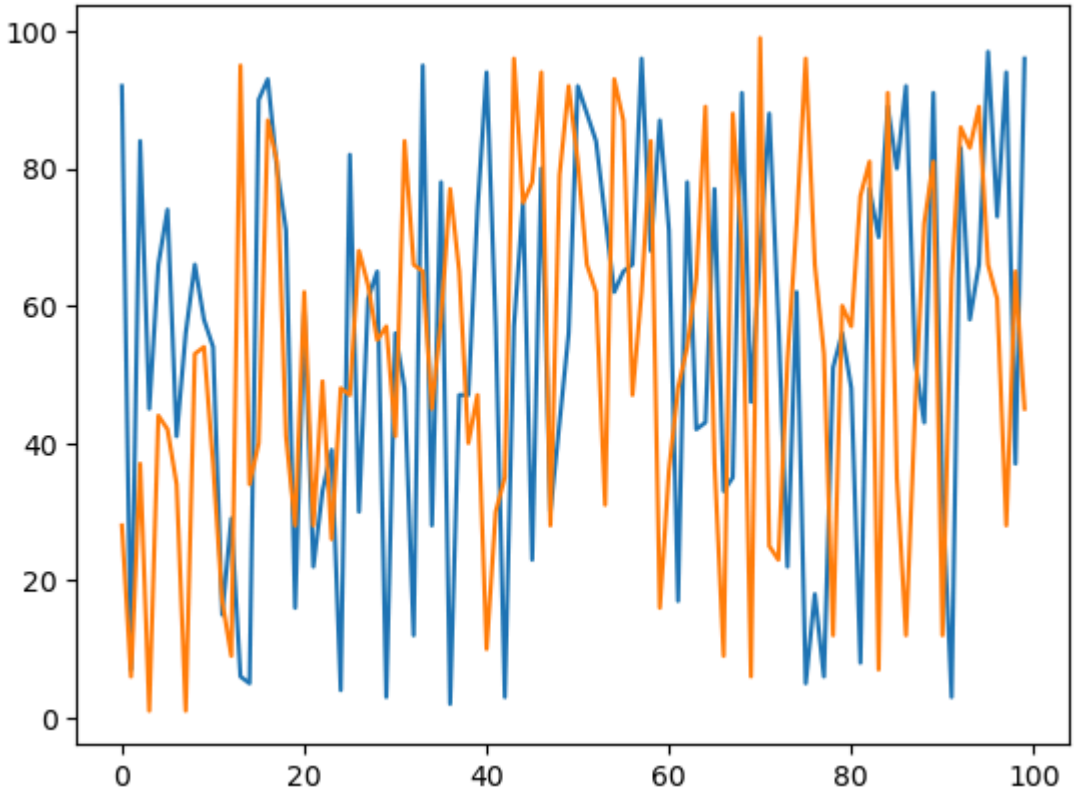
```
Student name: Yasmin Pokia
Student ID: s222245206
```

In [2]:
```
import random
import matplotlib.pyplot as plt

n_values = 100
y_values = []

# Create data (y_values) randomly between 1 and 100.
for i in range(n_values):
    y_values.append(random.randint(1, 100))

x_values = range(n_values)  # X is sequence of values 0-99
plt.plot(x_values, y_values)
plt.show()
```



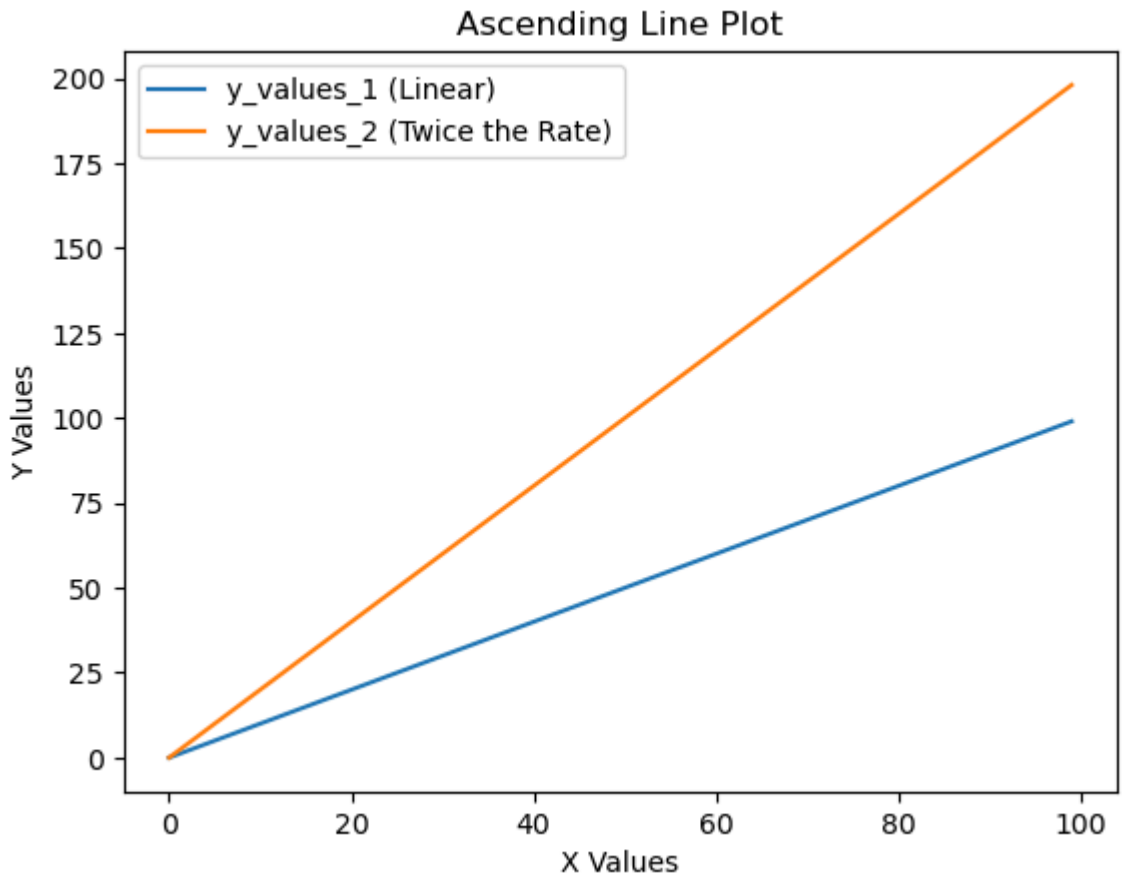
In [3]:
```
# Plot 2 variables
#

n_values = 100
y_values_1 = []
y_values_2 = []

# Create data (y_values) randomly between 1 and 100.
for i in range(n_values):
    y_values_1.append(random.randint(1, 100))
    y_values_2.append(random.randint(1, 100))

x_values = range(n_values)  # X is sequence of values 0-99
plt.plot(x_values, y_values_1)
```

```
plt.plot(x_values, y_values_2)  # call plot again draws in the same graph.
plt.show()
```

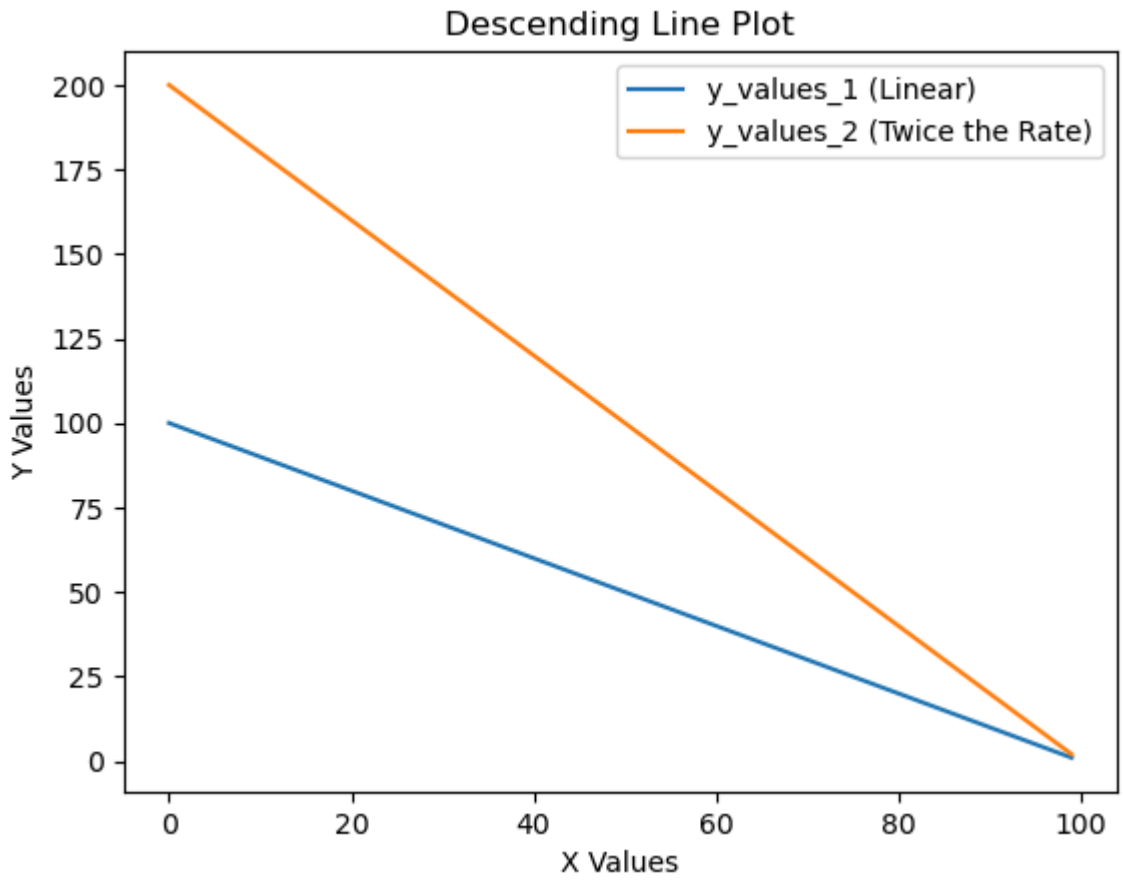


In [7]:

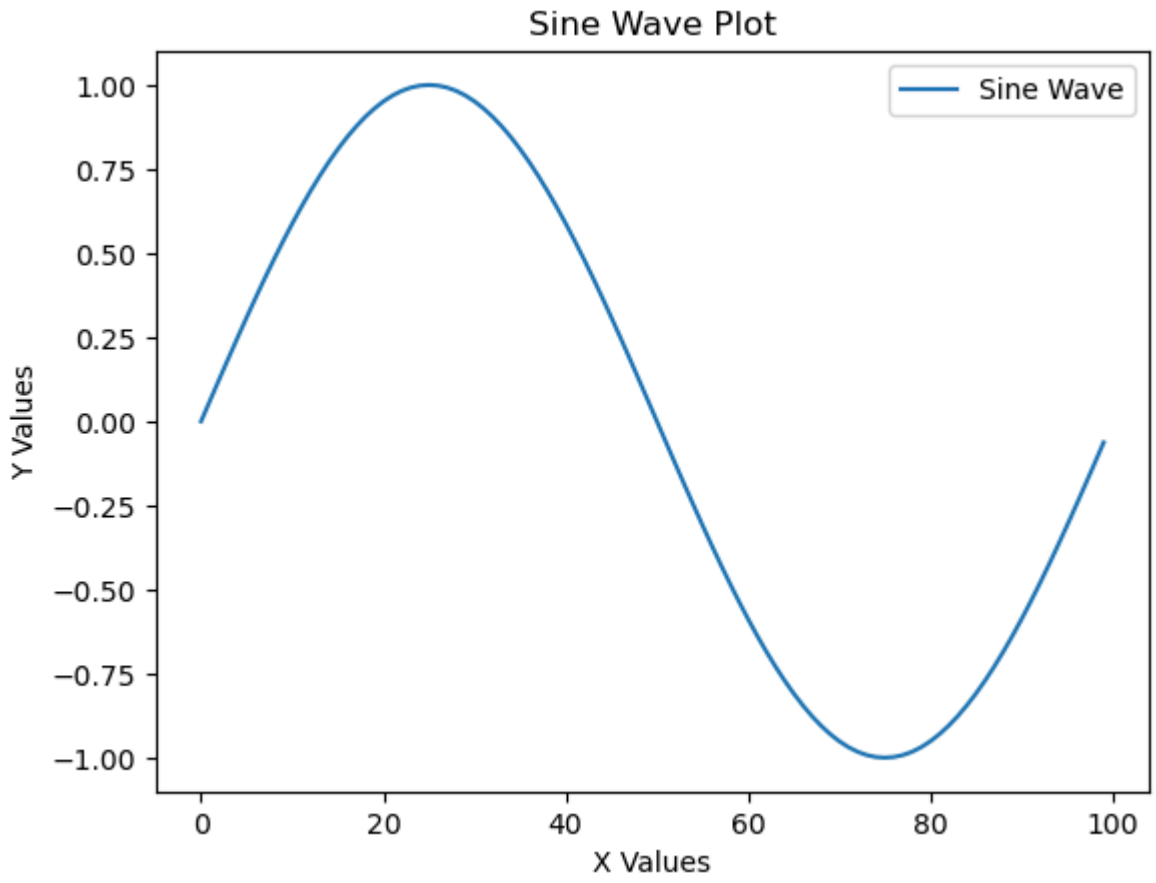
```
#
# Activity 1: Create data so that the plot draws an
# ascending line (y_values increase at any rate).
#

import matplotlib.pyplot as plt

n_values = 100
y_values_1 = []
y_values_2 = []

# Create data so that y_values increase at any rate.
for i in range(n_values):
    y_values_1.append(i)  # Linearly increasing
    y_values_2.append(i * 2)  # Increases at twice the rate of y_values_1

x_values = range(n_values)  # X is a sequence of values 0-99

plt.plot(x_values, y_values_1, label='y_values_1 (Linear)')
plt.plot(x_values, y_values_2, label='y_values_2 (Twice the Rate)')
plt.legend()  # Add a legend to distinguish the two plots
plt.xlabel('X Values')
plt.ylabel('Y Values')
plt.title('Ascending Line Plot')
plt.show()
```



In [8]:
```
#
# Activity 2: Create data so that the plot draws a
# descending line (y_values decrease at any rate).
#

import matplotlib.pyplot as plt

n_values = 100
y_values_1 = []
y_values_2 = []

# Create data so that y_values decrease at any rate.
for i in range(n_values):
    y_values_1.append(n_values - i)  # Linearly decreasing
    y_values_2.append((n_values - i) * 2)  # Decreases at twice the rate of y_value

x_values = range(n_values)  # X is a sequence of values 0-99

plt.plot(x_values, y_values_1, label='y_values_1 (Linear)')
plt.plot(x_values, y_values_2, label='y_values_2 (Twice the Rate)')
plt.legend()  # Add a legend to distinguish the two plots
plt.xlabel('X Values')
plt.ylabel('Y Values')
plt.title('Descending Line Plot')
plt.show()
```



In [9]:
```python
#
# Activity 3: Create data so that the plot draws a
# wave. You can consider using Python's math libarary, which has
# a sin function (detail https://www.w3schools.com/python/ref_math_sin.asp).
#

import math
import matplotlib.pyplot as plt

n_values = 100
y_values = []

# Create data so that y_values follow a sine wave pattern.
for i in range(n_values):
    y_values.append(math.sin(i * 2 * math.pi / n_values))  # Sine wave

x_values = range(n_values)  # X is a sequence of values 0-99

plt.plot(x_values, y_values, label='Sine Wave')
plt.legend()  # Add a legend to distinguish the plot
plt.xlabel('X Values')
plt.ylabel('Y Values')
plt.title('Sine Wave Plot')
plt.show()
```


Sine Wave Plot
Sine Wave
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
Y Values
0
20
40
60
80
100
X Values

In [ ]: