1. Import numpy library

```
In [3]: import numpy as np
```

2. Create three ndarray objects: x with 100 values between 0 and 20, y with corresponding sin values of x and z with corresponding cos values of x.

Hint: You can use numpy.linspace(start, end, #values), numpy.sin(ndarray) and numpy.cos(ndarray) functions respectively for above task.

```
In [4]: x = np.linspace(0, 20, 100)
        y = np.sin(x)
        z = np.cos(x)
```

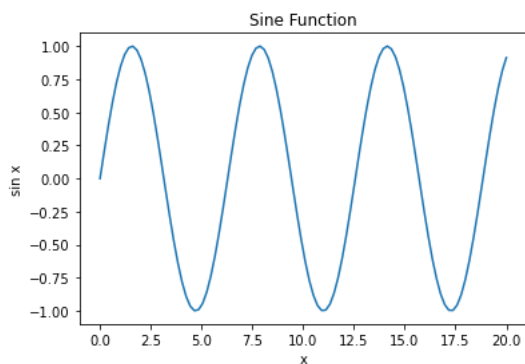### 3. Import matplotlib.pyplot module

```
In [5]: import matplotlib.pyplot as plt
```

4. Plot the values of y against x.

Hint: plot(x, y) and then show() functions from pyplot module can be used.

Hint: Additional functions title(text), xlabel(text) and ylabel(text) fcan also be used from pyplot.
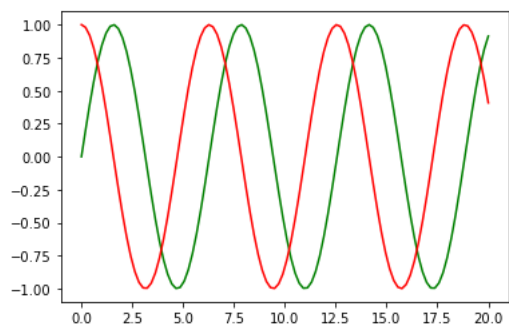
```
In [8]: plt.plot(x, y)
        plt.title("Sine Function")
        plt.xlabel("x")
        plt.ylabel("sin x")
        plt.show()
```


5. Plot the values of y and z against x with different line colors on the same window.

Hint: We can use third argument in plot function to represent color. "r" or "red" for red, "g" or "green" for green....
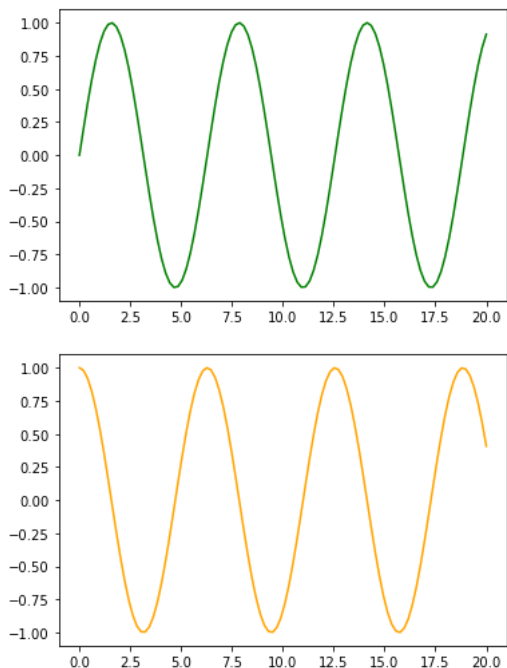
```
In [12]: plt.plot(x, y, 'green', x, z, 'r')
         plt.show()
```

6. Plot the values of y and z against x with different line colors on seperate windows.

Hint: We can use third argument in plot function to represent color. "r" for red, "g" for green....

In [10]:
```python
plt.plot(x, y, 'g')
plt.show()
plt.plot(x, z, 'orange')
plt.show()
```


**7. Create two ndarray objects each with 25 random integers.**

Hint: ndarray = numpy.random.randint(start, stop, size)

In [27]:
```python
feature1 = np.random.randint(1, 100, 25)
feature2 = np.random.randint(1, 100, 25)
```

8. Display scatter plot between above two variables.

Hint: Use scatter(x, y) function from pyplot module.

In [34]:
```python
plt.scatter(feature1, feature2)
plt.title("Scatter Plot")
plt.xlabel("feature 1")
plt.ylabel("feature 2")
plt.show()
```
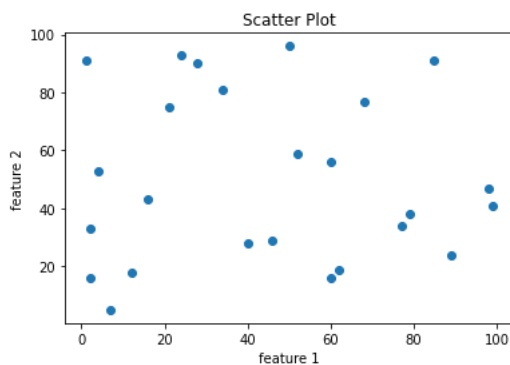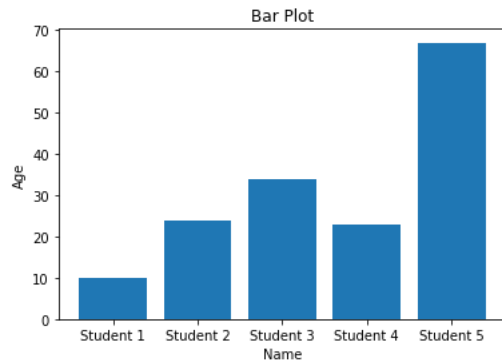
9. Create two lists: one with names of five students and other with their marks.

```
In [36]: names = ["Student 1", "Student 2", "Student 3", "Student 4", "Student 5"]
         ages = [10, 24, 34, 23, 67]
```

10. Display bar plot between above two variables.

Hint: Use bar(x, y) function from pyplot module.

```
In [37]: plt.bar(names, ages)
         plt.title("Bar Plot")
         plt.xlabel("Name")
         plt.ylabel("Age")
         plt.show()
```



```
In [ ]:
```