

```
# -*- coding: utf-8 -*-  
"""
```

Spyder Editor

Plots three different line plots.

```
"""  
  
# Our standard math library  
# The choice of name is yours, but np and plt are used by almost everyone  
import numpy as np  
  
# our standard graphics library  
import matplotlib.pyplot as plt  
  
def sin2(x):  
    # Every function should have a docstring describing its purpose and  
    # arguments (if more complex)  
    """ Calculates sin^2 of x """  
    f = np.sin(x) ** 2  
    return f  
  
def circle():  
    """ Returns x and y values of the unit circle """  
    phi = np.linspace(0.0, 2.0 * np.pi, 1000)  
  
    x = np.cos(phi)  
    y = np.sin(phi)  
  
    return x, y  
  
# calculate the sine of ONE value  
x = 2.0  
y = np.sin(x)  
print("y=", y)  
  
# create an array with 1000 x values  
x = np.linspace(-10.0, 10.0, 1000)  
print("x=", x)  
  
# calculate the sine for all values at once  
y = np.sin(x)  
print("y=", y)  
#  
  
# plot the sine and cosine functions  
plt.figure()  
  
# labels are used to produce the legend  
plt.plot(x, y, label="sin(x)")  
plt.plot(x, np.cos(x), label="cos(x)")  
  
# set the upper and lower limits of x  
plt.xlim(-10.0, 10.0)  
  
# addd axis labels  
plt.xlabel("x")  
plt.ylabel("f(x)")  
# add the legend  
plt.legend(loc="upper left")  
  
plt.savefig("sin_cos.png")  
plt.show()
```

```
# -----  
# calls the sin2 function and plots it  
plt.figure()  
  
plt.plot(x, sin2(x))  
  
plt.xlim(-10.0, 10.0)  
  
plt.xlabel("x")  
plt.ylabel("sin^2(x)")  
  
plt.show()  
  
# -----  
# plot the unit circle  
# optional argument figsize is used to make this a square plot  
plt.figure(figsize=(6.0, 6.0))  
  
x, y = circle()  
plt.plot(x, y)  
  
plt.xlabel("x")  
plt.ylabel("y")  
  
plt.show()
```