

## Importing Python modules

1-in the script editor, use an import statement to import statsmodels and add an as statement to alias statsmodels to sm.

2-add an as statement to alias seaborn to sns.

In [1]:

```
# Import statsmodels under the alias sm
import statsmodels as sm
# -----
# Use an import statement to import seaborn with alias sns
import seaborn as sns
```

## Correcting a broken import

Fix the import of numpy to run without errors.

In [ ]:

```
# Fix the import of numpy to run without errors
import numpy as np    #=>>>After changed
```

## Creating a float

1-Define a variable called bayes\_age and set it equal to 4.0.

2-Display the variable bayes\_age.

In [4]:

```
# Fill in Bayes' age (4.0)
bayes_age = 4.0
# -----
# Display the variable bayes_age
print(bayes_age)
```

4.0

## Creating strings

1-Define a variable called favorite\_toy whose value is "Mr. Squeaky".

2-Define a variable called owner whose value is 'DataCamp'.

3-Show the values assigned to these variables.

In [5]:

```
# Bayes' favorite toy
favorite_toy = "Mr. Squeaky"
# Bayes' owner
owner = 'DataCamp'

# Display variables
print(favorite_toy)
print(owner)
```

Mr. Squeaky  
DataCamp

## Correcting string errors

Correct the mistakes in the code so that it runs without producing syntax errors.

In [8]:

```
# One or more of the following lines contains an error
# Correct it so that it runs without producing syntax errors
#>> birthday = "2017-07-14"
#>> case_id = 'DATACAMP!123-456?'
# _____#
#Soulitons
birthday = "2017-07-14"
case_id = 'DATACAMP!123-456?'
```

## Load a DataFrame

Use `pd.read_csv` to load data from a CSV file called `ransom.csv`. This file represents the frequency of each letter in the ransom note for Bayes.

In [10]:

```
# Import pandas
#import pandas as pd
# Load the 'ransom.csv' into a DataFrame
#>> r = ____.( '____' )
# Display DataFrame
#>> print(r)

# _____#

#Soulition
import pandas as pd
r = pd.read_csv('ransom.csv')
print(r)
```

# Correcting a function error

## Exercise

The code in the script editor should plot information from the DataFrame that we loaded in the previous exercise. However, there is an error in function syntax. Remember that common function errors include: Forgetting closing parenthesis Forgetting commas between each argument Note that all arguments to the functions are correct. The problem is in the function syntax.

## Instructions

Correct the code so that it runs without syntax errors

In [12]:

```
# One or more of the following lines contains an error
# Correct it so that it runs without producing syntax errors
# Plot a graph
plt.plot(x_values y_values)
# Display the graph
plt.show()
#_____#

#Soulition
plt.plot(x_values, y_values)
plt.show()
```

# Snooping for suspects

## Exercise

We need to narrow down the list of suspects for the kidnapping of Bayes. Once we have a list of suspects, we'll ask them for writing samples and compare them to the ransom note. A witness to the crime noticed a green truck leaving the scene of the crime whose license plate began with 'FRQ'. We'll use this information to search for some suspects. As a detective, you have access to a special function called `lookup_plate`.

## Instructions

1-Create a variable called `plate` that represents the observed license plate: the first three letters were FRQ, but the witness couldn't see the final 4 letters. Use asterisks (\*) to represent missing letters.

2-Call `lookup_plate()` using the variable `plate`.

3-Calling `lookup_plate()` with the license plate `FRQ**` produced too many results. Luckily, `lookup_plate()` also accepts a keyword argument: `color`. Use the color of the car ('Green') to get a smaller list.

In [14]:

```
# Define plate to represent a plate beginning with FRQ
# Use * to represent the missing four letters
_____ = _____
# Call the function lookup_plate()
_____(____)
# Call lookup_plate() with the keyword argument for color
lookup_plate(plate, ____=____)
#_____#

#Soulition
plate = 'FRQ****'
lookup_plate(plate)
lookup_plate(plate, color='Green')
```