Assignment #3:

Objects, Instance Methods & Encapsulation

Requirements:

You are required, but not limited, to turn in the following source files:

1. Assignment3.java (Download this file and use it as your driver program for this assignment. Modifications on Assignment3.java file to fit your requirements are needed)
2. Car.java
3. Makes.java

Makes class

The Makes class describes information of the makes of a car. It has following attributes:

Attribute name	Attribute type	Description
country	String	The country of the car where it's made
manufacturer	String	The car manufacturer's name
brandName	String	The car brand's name, for example Explorer

The default value of country, manufacturer, brandName is "?".
Provide a constructor to set these default values.
public Makes( )

The following accessor methods should be provided to get the attributes:
public String getCountry()
public String getManufac()
public String getBrand()

The following modifier(mutator) methods should be provided to set the attributes:
public void setCountry(String nCountry)
public void setManufac(String nManufac)
public void setBrand(String nBrand)

The following method must be defined:
public String toString()

The toString method should return a string of a "Makes" object in the following format:
\nCountry:\tUSA\n
Manufacturer:\tFord\n
Brand:\t\tExplorer\n

Car class

The Car class describes a car that an owner can have. It has the following attributes:

Attribute name	Attribute type	Description
year	String	Year of the car is produced
color	String	The color of the car
price	double	The purchasing price
makes	Makes	The makes information of a car (see class **Makes**)

The following accessor methods should be provided to get the attributes:
public String getYear( )
public String getColor( )
public double getPrice( )

public Makes getMakes( )

The following modifier(mutator) methods should be provided to change the attributes:
public void setYear(String nYear)
public void setColor(String nColor)
public void setPrice(double nPrice)
public void setMakes(String nCountry, String nManufac, String nBrand)

The following method must be defined:
public String toString()

The toString() method constructs a string of the following format:
\nCountry:\tUSA\n
Manufacturer:\tFord\n
Brand:\t\tExplorer\n
Year:\t\t2015\n
Color:\t\tSilver\n
Price:\t\t30700.95\n\n

Assignment3 class
In this assignment, download Assignment3.java file, and modify it for your assignment. Assignment3.java will be the construction of 2 new classes and a driver program/class (the class containing a main method).

The following is the description of Assignment3 class.

The driver program will allow the user to interact with your other class modules. The purpose of this module is to handle all user input and screen output. The main method should start by displaying the following menu in this exact format:

Choice\t\tAction\n
------\t\t------\n
A\t\tAdd Car\n
D\t\tDisplay Car\n
Q\t\tQuit\n
?\t\tDisplay Help\n\n

Next, the following prompt should be displayed:
What action would you like to perform?\n

Read in the user input and execute the appropriate command. After the execution of each command, re-display the prompt. Commands should be accepted in both lowercase and uppercase.

Sample display
(A) Add Car
Your program should display the following prompt:

Please enter the car information:\n
Who is the car's manufacturer?\n

Read in the user input. Then the following prompt:

What is the car's brand?\n

Read in the user input. Then the following prompt:

Which country is the car made?\n

Read in the user input and set country, manufacturer and brandName on the makes object. Then the following prompt:

what year was the car made?\n

Read in the user input and set year attribute of the car object. Then the following prompt:

What color is the car?\n

Read in the user input and set color attribute of the car object. Then the following prompt:

How much was the car's price?\n

Read in the user input and set the price attribute of the car object.

Note that there is only one Car object in this assignment. Thus when "Add Car" option is selected more than once, the new one overwrites the old Car object information.

(D) Display Car
Your program should display the car information in the following format:

\nCountry:\tUSA\n
Manufacturer:\tFord\n
Brand:\t\tExplorer\n
Year:\t\t2015\n
Color:\t\tSilver\n
Price:\t\t30700.95\n\n

Make use of the toString( ) method of the Makes class to display this information. The toString method is used together with System.out.print method.
(System.out is NOT to be used within the toString method.)

(Q) Quit
Your program should stop executing and output nothing.

(?) Display Help
Your program should redisplay the "choice action" menu.

Invalid Command
If an invalid command is entered, display the following line:
Unknown action\n

Sample Input / Output
You can also refer to the Input files and Output files for the test cases that will be used as inputs for your program and examine what are the expected outputs of the corresponding input files.

Error Handling
Your program is expected to be robust to handle four test cases.

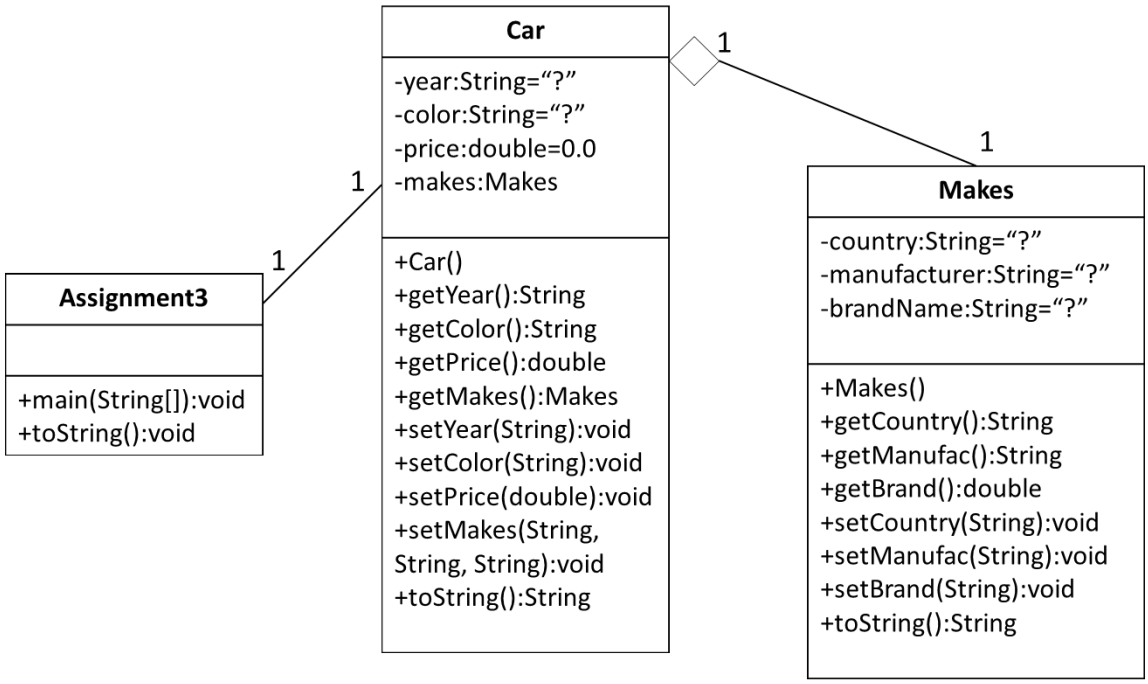
Requirements to get full credits in Documentation
The assignment number, your name, StudentID, Lecture number, and a class description need to be included at the top of each file/class.

A description of each method is also needed. Some additional comments for each method to explain codes that are hard to follow should be written.

Skills to be applied
In addition to what has been covered in previous assignments, the use of the following items, discussed in class, will probably be needed:

Classes
Instance Data
Accessors/Modifiers(Mutators)
Visibility Modifiers (Access specifier)
Encapsulation concept
Aggregation relationship between classes
Wrapper classes
NumberFormat class in java.text package

Program Description
Class Diagram:



Grading policy: There is a minimum deduction of 10 pts for any program that violates the public interface of the `Car` class and the `Makes` class. There is a 5 pt deduction for missing/incomplete javadocs, and a minimum deduction of 20 pts for an incorrectly submitted project.

Submission: Your Eclipse project is named `yourStudentID_3`.The project is submitted as `yourStudentID_3.zip`. Submit via eCourse. No other submissions will be graded.

Deadline: May 11, 2017