CS 170 ch.7 Lab 2

# Task 1

Question: **Car.java/CarDemo.java**

In this project, you will be including two classes:

A class called**CarDemo.java** whose purpose is to house the main method.

A functional/Concrete class called**Car.java** with the following components:

* 6 instance variables/ fields : (remember to give your variables private accessibility)
* String make;   String   model;      int   year;       int   speed;    String   color;  double  price;
* 1  default constructor (no parameters)
* an overloaded second constructor that takes in initial values to set your fields (6 parameters)
* **instance methods:**

      -several mutator and accessor methods (ex.   setModel(), setMake(), setColor(), getMake(), getYear()., getColor()............etc.)

     -   accel(): this method increases the speed by 10

     -   an overloaded accel() method that takes one parameter (amount) and increases the speed by a specific amount

         Note: if the speed exceeds 100; alert the driver with a message " Caution!! You may get a ticket!" and do not increase the speed.

       - brake(): this method decreases the speed by 10

        - an overloaded brake() method that takes one parameter (amount) and decreases the speed by a specific amount

       Note: your speed should not be negative. Your code should check the speed before decreasing and prints a message, "Speed cannot be negative".

        -displayFeatures(): this method displays all the instance variables

|  |
| --- |
|  |

**Testing Your Class**

**part A**

To test your class you will need to instantiate at least two different Car objects in the main method in the CarDemo class.

Instantiate the first object "adamCar" using the default constructor.

Instantiate the second object "sarahCar" using the second constructor.

**part B**

For the first object that uses the default constructor, you will need to invoke the mutator methods to assign values to your fields.

**Part C**

Demonstrate the use of the accessor method getColor() by accessing the color of each car and printing it to the console.

**part D**

Invoke the displayFeatures() method for both objects and display the variables.

**part E**

-accel Adam's car twice, use the accel with no parameters, then accel a second time passing in an argument of 100.

-brake adam's car using the overloaded brake and pass in 20 as an argument.

-add print statements after every accel and brake.

|  |
| --- |
|  |

|  |
| --- |
|  |

# Task 2

Question: **MyRectangleProject**

***part 1:***

Students will write two classes: **Rectangle.java** and **RectangeDemo.java**

**Rectangle Class:**

The Rectangle should have two private instance variables length, and width.

We will be using the default constructor so you will not be including a constructor in your code.

We will be writing 6 methods in the Rectangle class,

**setLength(), setWidth(), getLength(), getWidth()**, .(mutator methods and accessor methods.)

      getArea(), and getPerimeter(): calculate and return the values

**Note:**

You will need to decide whether these methods are void or return a value. You will also need to make a decision on the parameters needed.

**RectangleDemo Class:**

In the RectangleDemo class first, instantiate one object called**box1**. Populate the length and width variables using the appropriate methods.

Calculate **Area1** in the main method using the getLength() and getWidth() methods and print it out.

Calculate **Area2** in the main method using the getArea() method and print it out.

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |
|  |

***Part 2:***

-add an overloaded constructor that initializes the length and width to any given values. You will now need to add a default constructor that initializes the length and width variables to 0.

-add a printDimensions() method that prints the dimensions, length, and width of the rectangle.

-Test your code with various objects of different lengths and widths. You may call them **box2**, and**box3**.

**Test case1:**

box1: length = 10; width = 20; calculate Area1, Area2, Perimeter. Print the dimensions then, Area1, Area2, and the Perimeter:

**sample run:**

                     The length is: 10 ft.

                      The width is: 20 ft.

                      Area1 is: 200 sq. ft.

                      Area2 is: 200 sq. ft.

                     Perimeter is: 60 ft.

**Test case2:**

box2: length = 4; width = 5; calculate Area1, Area2, Perimeter. Print the dimensions then, Area1, Area2, and the Perimeter:

**Test case3:**

box3: length = 50; width = 40; calculate Area1, Area2, Perimeter. Print the dimensions then, Area1, Area2, and the Perimeter:

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

# Task 3

Question: **The use of reference variables**

1- Alter the Car Project from task #1. In the main method, add a third reference variable that refers to one of the existing objects.

Invoke the getColor() method for both reference variables that refer to the same object. They should both have the same color.

Print out the reference variables (this should demonstrate that reference variables contain addresses).

2- Alter Car. java and add a static variable called dealer and initialize with the value "Car Max".

     Add a static method called displayDealer() that displays the name of the dealership.

|  |
| --- |
|  |

|  |
| --- |
|  |