Assignment Content

Top of Form

**ASSIGNMENT NAME:          Car**

**GRADE VALUE:                     5%**

**UML DUE DATE: Jan 30**

**INSTRUCTIONS**

Write a program to model certain aspects of a car. Your car should know the following things about itself:

* + **make**: the make of the car
  + **distance driven**: the odometer reading
  + **fuel efficiency**: the amount of fuel required to travel a given distance
  + **fuel capacity**: the size of the fuel/gas tank
  + **fuel amount**: The quantity of fuel/gas currently in the tank

In the code comments, indicate the units in which each of the above quantities is measured.

When a new car object is created, it should have its make, fuel efficiency and fuel capacity set. All newly created cars have no fuel in the tank.

Your program must be able to add fuel/gas in the car tank. The amount of fuel to be added is specified as an argument to this method. The fuel amount can never be greater than the capacity. If someone tries to add more fuel than the tank can hold, this is not an error, but the excess fuel is spilled on the ground—it does not appear in the fuel amount.

Your car should be able to move. The distance to be moved is specified when the car is asked to move. Moving causes the odometer reading to increase, and the fuel amount to decrease, based on the fuel efficiency of the car. The car can’t move once it runs out of gas. Note, the fuel amount can never be negative. For example, if you move 100 kilometres, and the car only has enough fuel to go 80 km, then the distance driven should be increased by 80 kilometres, the fuel amount is reduced to zero. No error message is provided when the car is out of gas.

Your car should be able to describe all its fields/attributes using a method called toString that returns a string (including linefeeds where appropriate).

Be sure to include getter and setter methods for each of the fields in your class **where appropriate**.

Write another class, CarTest, which holds a main method that tests your Car class. It should create at least two instances of Car, and show that all of the methods described above have been implemented correctly, by calling these methods and printing out the results. Show that the fuel tank cannot be overfilled and that the car does not run without fuel. Moreover, working with two instances also shows that changes to the properties of one car do not affect the other car.

**Note:**Provide a UML diagram for your class including all fields and methods for approval before coding this program.

This assignment covers Week 1 to 4. After UML is approved, you can get started with coding class and methods. Later write required if-else statements, which you are learning in week 4.

**Sample Output run:**

Honda has been driven 0.0 kilometers

It has a tank capacity of 50 litres

It get 10.0 kilometers to the litre

It has 0.0 litres in Honda's tank right now

Nissan has been driven 0.0 kilometers

It has a tank capacity of 75 litres

It get 10.0 kilometers to the litre

It has 0.0 litres in Nissan's tank right now

Honda has been driven 0.0 kilometers

It has a tank capacity of 50 litres

It get 10.0 kilometers to the litre

It has 40.0 litres in Honda's tank right now //filled 40L

Nissan has been driven 0.0 kilometers

It has a tank capacity of 75 litres

It get 10.0 kilometers to the litre

It has 75.0 litres in Nissan's tank right now //filled 100L

Honda has been driven 40.0 kilometers

It has a tank capacity of 50 litres

It get 10.0 kilometers to the litre

It has 36.0 litres in Honda's tank right now //drive 40Km

Nissan has been driven 750.0 kilometers

It has a tank capacity of 75 litres

It get 10.0 kilometers to the litre

It has 0.0 litres in Nissan's tank right now //drive 800Km

**RESOURCES:**

Please email me via Blackboard if you need help with this assignment.

* + Explain the problem you are having
  + Include a zip file of your project

**SUBMISSION REQUIREMENTS**

Zip your entire project folder and submit in Blackboard.

**REMINDER**

This is an individual assignment. You are reminded that it is an academic offence to submit other people’s work on your own, or to give others your work so that they can submit it as their own.

Bottom of Form