## Let’s Go For A Drive

Write a class called Car that keeps track of its mileage (number of miles it’s driven), the number of gallons of gas left in its tank, and whether or not it is broken down. You should be able to create a Car by specifying the initial amount of gas in the tank (decimal), and the initial mileage (decimal), like so:

Car myCar = new Car(10.0, 1000.0); // 10.0 gallons, 1000.0 miles driven

The main functionality of a Car is the turnOnAndDrive method that takes as a parameter the number of miles the car should drive. This method should do the following:

* Every time a car turns on to drive, there is a chance it breaks down. Specifically, the chance of a breakdown is 0.1 (10%) if you’ve driven 0 to 10,000 miles, 0.2 (20%) if you’ve driven 10,001 to 20,000 miles, etc. If the car breaks down, it can no longer drive.
* If the car turns on successfully, it should drive the number of miles passed in as a parameter. Its mileage and gas left should be updated accordingly to indicate the car has driven additional miles, and has used up some gas (there is a provided MILES\_PER\_GALLON constant that indicates how many miles the car can go on one gallon of gas. You do not have to define this constant in your answer).
* If the car can drive, but does not have enough gas to drive the full specified distance, it should drive until it runs out of gas.
* The method should return true if the car was able to drive the specified distance, and false otherwise (if it breaks down or does not have enough gas).

For example, using our myCar from earlier, assuming our car does not break down, and assuming MILES\_PER\_GALLON = 25, myCar.turnOnAndDrive(25) would return true and our car would now have 9 gallons of gas and 1025 miles driven.

Then, if we call myCar.turnOnAndDrive(250) it would return false because our car does not have enough gas to drive this distance, and our car would now have 0 gallons of gas and 1,250 miles driven (because 9 gallons of gas gets us 225 more miles).

In total, your class must implement the following public methods:

**public interface** Car {  
 */\*\*  
 \* Attempts to drive # miles, returns true for success or false for fail  
 \*/* **public boolean** turnOnAndDrive(**int** milestToDrive);  
  
 */\*\*  
 \* Returns the number of miles this car has driven.  
 \*/* **public int** getMileage();  
  
 */\*\*  
 \* Returns true if the car is broken down, and false otherwise.  
 \*/* **public boolean** isBrokenDown();  
  
 */\*\*  
 \* Sets the car as no longer broken down  
 \*/* **public void** repair();  
  
 */\*\*  
 \* Adds the given number of gallons of gas to the car’s gas tank  
 \*/* **public void** fillGas(**double** numberOfGallons);  
}

Do:

1. Create a maven project for your spaceship
2. Create a git repository on github for the spaceship. Add the maven project.
3. Create an OOP model for your spaceship. Chose the best data types and structures for the problem.
4. Implement the turnOnAndDrive(…) method. Use the RandomGenerator class for modelling the breakdown chance
5. Add unit tests for your implementation. The coverage should be higher than 80%

Part 2:

You are employed by a car manufacturer who wants you to write software to help test how long their cars will go before they break down. Write a method called testCar that makes a new Car and returns the mileage of that car once it breaks down for the first time. The car should start out with 10 gallons of gas, and 0 miles driven, and you should drive the car in increments of 10 miles. If the car runs out of gas, you should refill it to 10 gallons of gas. Once the car breaks down, you should return how many miles the car had driven.