**Used Cars – Part 1**

Having already accomplished your goal of watching every Youtube video ever, you decide to embark on a new quest; an exciting journey into the world of used car sales. You decide to write software to track your inventory, such that you can better understand sales trends and build yourself a used car sales empire. 

Name your BlueJ project “usedCars”.

Begin by writing two classes that will represent your used car inventory:

1. The Car class will have the following:
   * String type – type of car, e.g. "Honda Civic"
   * int year – year the car was made
   * double price – price of the car, in dollars
   * double mpg – miles per gallon
   * A parameterized constructor to initialize all the instance variables.
   * boolean greatGasMileage() – returns true if the mpg of that Car is >= 36
   * String toString() – method that will return info for that Car, in the following fashion (don't worry about decimal formatting):

1998 Honda Civic, 32.0 mpg, $6499.0

1. The Truck class will have the following:
   * String type – type of truck, e.g. "Ford F-150"
   * int year – year the car was made
   * double price – price of the car, in dollars
   * int towing – the truck's towing capacity, in pounds
   * A parameterized constructor to initialize all instance variables
   * boolean canTowBoat() – returns true if the towing for that Truck is >= 2,000.
   * String toString() – method that will return info for that Truck, like the following:

2004 Ford F-150, 3200 lbs. towing, $8999.0

1. Next, write a class called Inventory that will store all the vehicles in your inventory. The Inventory class has the following:
   * ArrayList<Car> cars – PIV that is a list of all the cars in your inventory
   * ArrayList<Truck> trucks – PIV that is a list of all the trucks in your inventory
   * a default constructor and a parameterized constructor that initializes the ArrayList objects
   * void addCar(Car c) – add a Car object to the list (the lists should be private)
   * void addTruck(Truck t) – add a Truck object to the list
   * void listInventory() – iterates through all the cars and trucks in the inventory and prints their information (calls their toString() method). **Use for-each loops for this.**
2. In the main() method in a Runner class, create at least 2 Car objects and 2 Truck objects, and test that all your methods work (add to car and truck inventory inventory, and list the info for your inventory).