Programing Languages 2 - Test 1(T12)

2015. autumn

Name: _____

As I already did it by taking this course, I accept all rules and regulations written on the homepage of the course. Without signature this Exam cannot be graded, and counted in to the final grade.

_____

signature

1. Create a new NetBeans Java project. The name of the project has to be your family name (If you have more than one family names you can choose). The name of the package has to be testOneT12. Name the Main class "Main"
2. Create a new class named `Vehicle` under the project. This class has the following attributes: `plateNumber` (like "ASD-123"), `power` (like 120), `automata` (a boolean variable holding true if the car is automata and false if not). All this properties have to be **hidden** from the outside.
3. Create getter and setter methods for the above attributes. For the `plateNumber` and the `automata` add getter and setter methods, but for the third one provide **only a getter**.
4. Overload the `setAutomata(Boolean automata)` method of the class by adding a new `setAutomata(String automata)` method to the class. If the String parameter is "auto" then the Vehicle will be automata in all other cases it will not be.
5. **Add a constructor** to the class with which the user can set all three properties of the newly instantiated `Vehicle` object. The class must not have any other constructors.
6. Override the `toString` method of the class. If it is called, it returns a String like the following (for a vehicle with plate number "FTG-175", that is automata and it has 120hp: "`FTG-175: 120hp (a)`", or for one with plate number "ABC-777", is manual and has 100hp: "`ABC-777: 100hp (m)`"). Take care to return a String **exactly** in this format. *(Without the "" characters)*
7. Create a child class of the Vehicle class named Van. It has an **extra attribute** named carryingCapacity (int). Write getter and setter methods for this private attribute as well. This class has **only one constructor** which sets all the four fields of the class. **Override the toString** method of this class as well so that it prints out the same String as its parent plus the string " max. " plus the `carryingCapacity` plus the string "kg". (e.g.: "`GTF-123: 130hp (a) max. 2500kg`").
8. In the Main class, in the main method instantiate a new Vehicle object with the following properties: plate number: "`GTF-264`", power: `110`, automata: `false`. Print out the object.
9. In the Main class, in the main method instantiate a new Van object with the following properties: plate number: "`HJZ-375`", power: `97`, automata: `true`, carryingCapacity `2200`. Print out the object.
10. In the Main class create a **static method**, that gets a Van object as a parameter and returns a boolean value. True if the carrying capacity of the van is less than 50 and false otherwise. The name of the method has to be `isVanStrong(Van v);`
11. Call the above created method and print out the result of it to the screen.