Programing Languages 2 - Test 1(T8)

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 testOneT8. Name the Main class of our project "Main".
2. Create a new class named `Animal` in the project. This class has the following attributes: name, numberOfLegs, predator (a boolean variable holding true if the animal is a preadator else it holds false). All this properties have to be **hidden** from the outside.
3. Create getter and setter methods for the above attributes. For the name add both getter and setter methods, but for the other two provide **only getters**.
4. Overload the `setName(String name)` method by adding a new parameterless `setName()` method that sets the name of the Animal to "unknown".
5. Add a constructor **without parameters** to the class. This does not do anything else, but instantiates a new `Animal` object. **Add another constructor**. This one has 3 parameters with which the user can set all three properties of the newly instantiated `Animal` object.
6. Override the `toString` method of the class. If it is called, it returns a `String` like the following (for an animal named "`Wolf`", that has of course 4 legs: "`Wolf (4 legs) - predator!`", or for one named "`Chicken`", (that has 2 legs): "`Chicken (2 legs)`"). Take care to return a String **exactly** in this format. *(The "" characters are not printed out.)*
7. Create a child class of the `Animal` class named `Fish`. `Fish` objects are special, because thie value of numberOfLegs must be 0. Add only one constructor to this class that lets the user to set the name of the fish and if it is predator or not, but sets the number of legs to 0. There must not be any other constructors in the class. **Override the toString** method of this class as well so that it prints out only the name of the fish and that it is a predator or not. (Like "Tuna – predator"). Do not print out the number of legs.
8. In the Main class, in the main method instantiate a new `Animal` object with the following properties: name: "Chicken", 2 legs, not a predator. Print out the object.
9. In the Main class, in the main method instantiate a new `Fish` object with the following properties: name: "Tuna", predator. Print out the object.
10. In the Main class create a **static method** that gets two `Animal` objects as a parameters and returns true if their name are the same. Else it returns false.
11. Instantiate a new `Animal` object in the main method and **use the above method** on the Chicken and this new object from the main method. If their names match print out "Same names" else print out "Different names";