Class Lab

Create a new project named class_lab with a new class named ClassLab. When you create this class, do **not** use a main.

Make sure you use comments and correct java conventions.

- Copy the file ClassLabClient.java into your class_lab project folder that contains the ClassLab Class.
- The ClassLabClient file will have a lot of red errors and warnings. This lab will go over and fix all the warnings so the program runs correctly.

1. In ClassLab.java, create 3 private variables. A String that stores a name; an integer that stores an age; a String that stores a profession. When you initialize these variables, you don't set them to any values.
2. In ClassLab.java, create a constructor without any parameters. Set each of the private field variables to a "default" value (i.e.: name = "xxx";).
3. In ClassLab.java, create a constructor that has 3 parameters (each parameter representing a private field variable). Set each private field variable to the parameter being passed in (i.e.: this.name = name;).
4. Create an accessor method for each field variable. An accessor method is one that returns the value of the private field variable. The names of these accessor methods are: getName(), getAge(), getProfession(). They return the data type of the variable.
5. Write a mutator method that will set the name. The method will be called setName, returns a void and has a String parameter. Use "this" to set the private variable 'name'.
6. Write a toString() method. You return a String that is formatted in a way you want the name, profession and age printed to the screen.
7. Write a method ageDifference that will return an integer and has a ClassLab object as a parameter. Return the difference (subtract) between this age and the parameter age.
8. Create a static String field variable named 'relation'. Make this field variable public.
9. Write a static method named getRelation that returns the static relation variable. Notice how it is accessed in the client cod