CSC 1051 – Writing classes Examples
1) Let’s implement a Person class.

```java
public class Person
{
    // instance variables: name, age, and happiness (true or false)

    // constructor: everyone starts out happy and at age 0

    // birthday(): increases age by one

    // makeHappy(): causes this person to become happy

    // makeSad(): causes this person to become not happy

    // getAge(): returns the age of this person

    // toString()

}
```

2) Implement the driver class:

```
// Instantiate one object of the Person class, assign it to a variable named Jackie

// print out info about Jackie

// use makeSad() method to make Jackie unhappy.

// use the birthday() method to increase jackie’s age

// use makeHappy() method to make Jackie happy.

// print out info about Jackie
```
3) In the driver class, add a few more Person objects and increase their ages a few times. Use getAge() to access their ages to compute and then print out the sum of all their ages.

4) Add another method to the Person class, complain(). If the person is happy, the method should return the String “I have nothing to complain about”, otherwise, the String “I am so sad”. Add appropriate code to the driver class to test it.

4) Add another version method to the Person class, complain(), that has one parameter, an integer n. If the person is happy, the method should print “I have nothing to complain about”, otherwise, “I am so sad”. The method should not return anything. Add appropriate code to the driver class to test it.

5) Add another version of the method complain() to the Person class, with one parameter, an integer n, that causes it to return a String that contains the complaint n times if the person is sad (it should behave exactly the same as previous version for happy persons). Add appropriate code to the driver class to test it.

6) Sketch a UML class diagram for your classes. Include all the instance variables and methods from previous questions.

7) OPTIONAL: To provide more flexibility, you can add more methods (“getters” and “setters”) for all the instance variables. For example, you might also have a method setAge() that sets the age to the value given by its int parameter (so you don’t need to use the birthday() method to increase the Person’s age one year at a time).

More Practice: TRY THIS AGAIN WITH DOGS!

Create a class for a Dog, similar to the Person class, above. Instead of having a boolean variable to encode happiness, have an integer that encodes the dog’s state (eg: 0=sleeping, 1=eating, 2=running, etc – come up with a short list of states and stick to it). Include a constructor, toString(), and accessor and mutator methods for all its variables. Come up with questions similar to 2, above, for the Dog class. For example, write some code that would set a Dog object’s state to sleeping or running, depending on the Dog’s age. Draw the UML class diagram for the Dog class.