Coding Assignment Instructions
Classes *Pedometer* and *PedometerClient*

Create Java classes *Pedometer* and *PedometerClient* in a new package named *net.ligent.students.pedometer* . Use the same approach that I demonstrated in the tutorial demo for *Counter* and *CounterClient*. Detailed requirements for each class are presented below.

Requirements for *Pedometer*

Since pedometers and counters have similar functions, you can consult the counter classes for hints on how to create the pedometer classes. Also, please follow the document that I have provided entitled "*Recipe for a New Object-Oriented Class (a new type)."* This document documents the steps that I followed in the tutorial demo and that you should follow while doing this assignment.

The Pedometer class should have **only one instance variable** named "*steps*". When you construct a new Pedometer, steps should be equal to zero. You should create standard getters and setters for the *steps* instance variable.

Remember to create a No-Argument constructor for *Pedometer*. While this is not technically required, it is a good practice. No other constructors will be required.

The *Pedometer* class should have a *toString()* method. The Pedometer class should have two other methods: *stepForward()* and s*tepBackward().* The first of these should add 1 to *steps*. The second should subtract 1 from *steps*. While you may begin by creating a naive version of these methods that access the instance variable *steps* directly, your final version should replace that code with more sophisticated code that uses *getSteps()* and *setSteps()* to read and modify the steps instance variable. The *setSteps()* method should have the modifier of *private* to prevent class users from setting the value for steps directly. The value may only be manipulated by calling *stepForward()* or *stepBackward().*

Use Eclipse to generate an *equals()* method for *Pedometer*. Two *Pedometer* instances should be considered equal when their number of steps are equal. Remember to generate a *hashCode()* method at the same time that you generate the *equals()* method. These two methods must be maintained together so that they always remain compatible.

Requirements for *PedometerClient*

As a testing platform for the *Pedometer* class, create the class *PedometerClient*. This should be a simple Java class with a *main* method that makes calls to the methods in *Pedometer* in order to fully test *Pedometer*. When coding the *main* method, please remember to include code for all necessary test cases. Typically, this means that you must create a minimum of one test case for each of the public methods in *Pedometer*. There should even be a test case for the *toString()* method.