Programming Lab Exercise 2

Before you start:
Create a folder called l**ab2** inside your personal **java** folder you created at the start. Save all your work for lab 2 in this folder. As usual name your files according to the question e.g. **lab2aq2.java** unless otherwise requested in the question.

Using Eclipse:

Your understanding of the concept of classes and methods are examined here, in particular:

- Writing class definitions
- Creating instances of classes
- Calling object methods
- Accessing data belonging to an object instance

Complete each question (successfully!) before you move on to the next one.

Exercises:

Q1.
Develop a java class called HotelRoom which will be used to manage the rooms in a Hotel. The class should contain a private integer value called roomNumber, and a private string called roomType (which can be either “Single” or “Double”). Write the getter and setter methods for each of these variables. You will also need to write the necessary constructor method which takes no arguments.
Write a driver program which instantiates 2 room objects from this class as follows:
roomA (room number is 200, type is “Single”)
roomB (room number is 201, type is “Double”).
The program should then print out the details for these rooms by calling the appropriate getter methods.

Q2.
To enhance our room management system we want to know if the room is occupied or vacant. Add a new private integer which can be either 0 or 1 (0 means vacant, 1 means occupied). In addition, we need to know what the nightly rate for each room is. Add a double variable called rate. Write the necessary methods to set and get this new variable. In the driver program set roomA to be occupied with a rate of 100, and roomB to be unoccupied with a rate of 80 by calling the appropriate method.
Add these values to the output statements.

Q3.
Add a second constructor method which takes the 4 values as arguments (roomNumber, roomType, occupied, rate) and instantiates the instance variables with these values. Demonstrate this by creating a roomC object with roomNumber=202, roomType=”Single”, occupied=0, rate=90).

Q4.

We want to be sure that we don't allow double booking of a room. Write a new method called **`isOccupied`**() which replaces the vacancy parameter and returns a Boolean of True if the room is already occupied and False otherwise. Demonstrate this in the driver program by trying to set roomB to occupied a second time. You should call the **`setOccupied`**() method only if the **`isOccupied`**() method returns **`False`**.