

Object Oriented Programming (JAVA)

24F CST 8284

Assignment 1

Instructions

This assignment is designed to be completed and submitted by you **individually**. It is your responsibility to ensure that your work is NOT **replicated** by any other student, and that you do not copy or submit someone else's work. Avoid plagiarism.

The **due date** for submission of this assignment is **as stated in the Assignment 1 page on Brightspace**. The final date to demo this assignment is in **Week #7 (whichever day your lab session falls into)**. If you finish on time and need to demo earlier, please ask your lab professor.

Carefully read every instruction and the tasks to ensure that you deliver all aspects of your solution as required. This assignment when completed fully and correctly **weighs 10%** of the entire course marks.

Problem Description

Digitization of health records of patients can be shared easily using computer systems among care providers to facilitate better patient care.

In every patient's electronic record. In this assignment, your task will be to facilitate the creation of hospital patients' records during their surgery procedures in a hospital.

Ensure that you completely review the **code file** and to **understand** what they contain and the **modification** you are required to make.

In this assignment:

1. **Download, review** and **modify** the java class file provided for you named **MyHealthData.java**.
2. **Create** the commensurate **UML class diagram** for this assignment to show all the **attributes, methods, constructors, etc.**, which are required to help explain your solution.
3. The **attributes** of **MyHealthData.java** should include the following patient's data:
 - a. First name
 - b. Last name
 - c. Gender
 - d. Home address
 - e. Level of education
 - f. Height (in inches)
 - g. Weight (in pounds)
 - h. **Hint 1:** Check YOUR **UML class diagram** and ensure that you **include** all other **attributes** that have been listed in the class.
4. The class (**MyHealthData**) should have a constructor that receives these data as parameters.
5. For each attribute mentioned in above, you should provide the appropriate **set** and **get** methods.
6. The class must provide **methods** to calculate and return the patient's:
 - a. Age (in days)
 - b. *MaximumHeartRate* (**Hint 2:** *MaximumHeartRate* is calculated as:
 $220 - \text{Age in days}$)
 - c. Target heart rate range (The Hints to calculate the two components in this range are provided below):

Hint 3: *MinimumTargetHeartRate* is calculated as: 50% of *MaximumHeartRate*.

Hint 4: *MaximumTargetHeartRate* is calculated as: 85% of *MaximumHeartRate*.

- d. Body Mass Index (BMI) (**Hint 5:** This calculation has already been provided for you)
7. Write a Test class for your code named **MyHealthDataTest.java**. This code must **prompt the user for input** of the patient's data (described above), **instantiate an object** of the class **MyHealthData** for that patient and then prints the data from that object.
The Health data to be **displayed** must include the patient's:
 - a. First name
 - b. Last name
 - c. Gender
 - d. Home address
 - e. Level of education
 - f. Date of birth
 - g. Height
 - h. Weight
 - i. Calculate and print patient's age in days, BMI, maximum heart rate, Target heart rate range (*MinimumHeartRate* and *MaximumHeartRate*).
 - j. It should also display the **BMI values** (check your code to see this value).
8. Execute your code and include your **output file to your submission**. Your output file must be **eclipse screen shot only**. Any other form of representation of your output will NOT be accepted. Be sure to capture the portion of the eclipse screen that includes your output the title portion above the output that shows the time of the execution.
9. Use the **Javadoc style comments** to explain your work in detail. You are required to generate and submit the Javadoc files.

10. **Zip all your files** and submit your work your lab section (designated portal by your lab professor). While naming your file, use the naming format (use to avoid losing marks): **Lab_Section_XXX_YourFirstName_YourLastName**.

COMPONENT 2

1. Create a **JUnit test** for **MyHealthData** and name it **MyHealthDataTest2**. Write your test case using suitable **assertions**.
2. In your first **JUnit test output file (1)**, show a successful execution of your test with no errors (showing your output and the green bar).
3. In your second **JUnit test output file (2)**, show a failed execution of your test (as a result of introducing an error into your code) – by showing your output and red bar).
4. Using Javadoc style comments state any modification or assumption you have made and where you made it. These assumptions and/or modifications must be stated at the beginning of your **Test** file.