Assignment 5

Due date: as indicated on Moodle

You may work alone or in a group of 2.

Objectives:

- Handling exceptions
- Create a custom exception class and use it

Instructions:

Remember Assignment 1 where you developed a BMI calculator? In this assignment, you will modify the code of assignment 1 to add exception handling.

1. You read the weight and height from the user using a Scanner object. Use exception handling to handle the case where the user types a string instead of giving you a number. Make sure your program loops until the user enters a valid numerical value. That numerical vale may or may not be in the valid range of height or weight.

2. **Remember**, the application should not accept weights more than 300 kg or less than 10 kg and heights more than 2.2 meters or less than 0.2 meters.

3. You need to create three exception classes: WeightOutOfRangeException, HeightOutOfRangeException, and NoBMIException.

4. Redo the BMI Calculator code by having a class BMICalculator that has two attributes: height and weight. This class must have the following methods:

    - Default constructor that initializes the values height and weight to 0
    - No other constructors
    - setHeight(height): this method throws HeightOutOfRangeException if the given height is out of range
    - setWeight(weight): this method throws WeightOutOfRangeException if the given weight is out of range
    - The two methods setHeight and setWeight do not catch or handle the exceptions. The driver class is meant to handle the thrown exceptions.
    - ReturnSingaporeanBMIMeaning. This method calculates BMI value ONLY if we have valid height and weight and returns a string that has the Singaporean interpretation of the BMI. Otherwise it throws a NoBMIException. The exception is meant to be handled in the driver class.

5. You must create a UML class diagram in a pdf format. You may use draw.io or the free version of LucidChart

6. Once you are done, you need to zip the whole project files, the UML diagram in pdf format and upload the zip file to Moodle.

7. The whole team will give a demo to the instructor in the lab.

Grading Rubric

Criteria	**Marks**
Internal documentation	1
Code quality (Meaningful names, indentation, spacing, etc..)	1
Handling the exceptions thrown by the Scanner object	2
Creating the exception classes and using them	5
Class design (UML class diagram)	1
Total	**10**

Have fun ☺