Guided LAB - 303.9.1 Encapsulation in Java

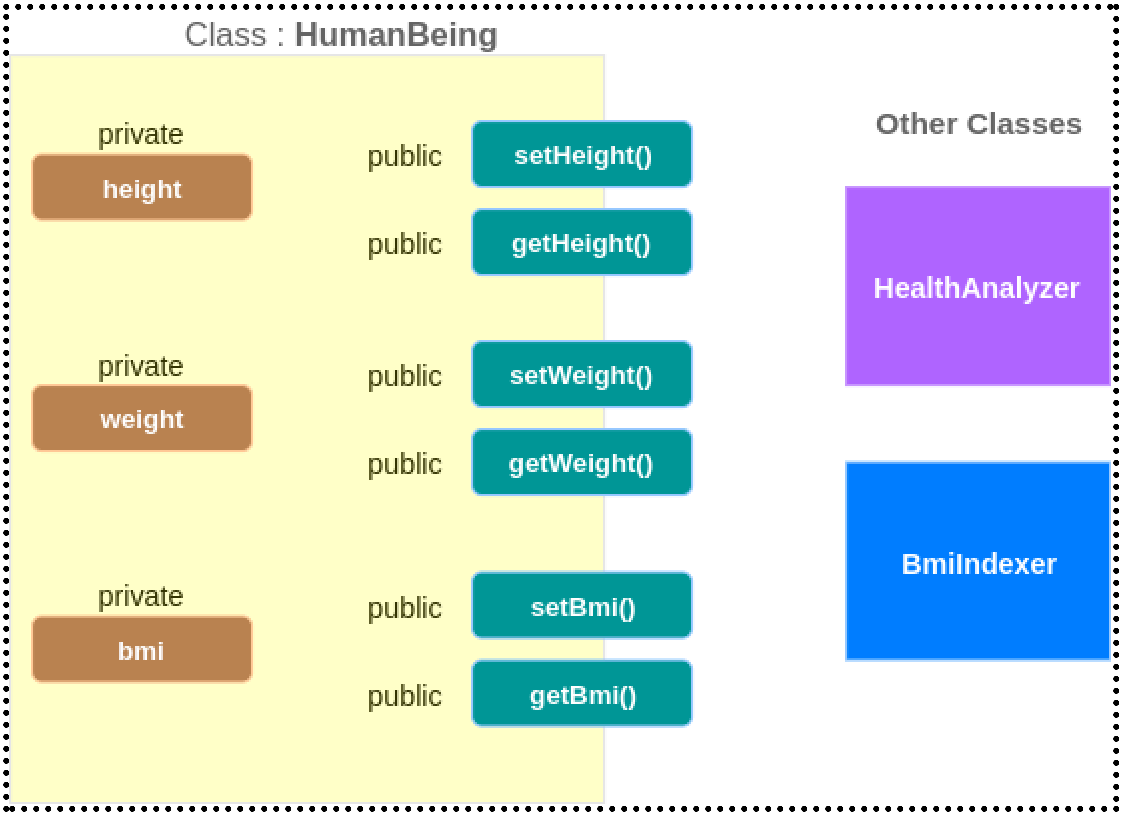
Objective:

In this lab, we will demonstrate Encapsulation in the Java language. By the end of this lab, learners will be able to use Encapsulation.

Introduction to Encapsulation in Java:

Encapsulation refers to hiding the class variable from other classes and giving access to them only through methods (setters and getters).

Encapsulation in Java means binding the data (variables) with the code (methods – setters and getters). The below diagram depicts Encapsulation in Java.



💡 Points of the above diagram:

- Class Variables (in the example: *height, weight, and bmi*) are declared ***private;*** hence, they are not visible to other classes.
- For each variable, there is a setter and getter method, which **sets** a value to the variable and **gets** the variable's value, respectively.
    - Example: For variable height, ***setter method is setHeight()*** and ***getter method is getHeight().***
- Setter and Getter methods are ***public;*** hence, they are visible to other classes.

A Program Example:

Create a class named ***HumanBeing*** with two constructors: class variables (**weight, height, and bmi)** and setter and getter methods.

```
public class HumanBeing {

    private float weight;
    private float height;
    private float bmi;

    public HumanBeing(float weight, float height, float bmi )
    {
        this.weight = weight;
        this.height = height;
        this.bmi = bmi;
    }
    public HumanBeing()
    {

    }
    public float getWeight() {
        return weight;
    }
    public void setWeight(float weight) {
        this.weight = weight;
    }
    public float getHeight() {
        return height;
    }
```

```java
    public void setHeight(float height) {
        this.height = height;
    }
    public float getBmi() {
        return bmi;
    }
    public void setBmi(float bmi) {
        this.bmi = bmi;
    }
}
```

Create a class named **EncapsulationExample**, with a ***main()*** method.

```java
public class EncapsulationExample {
   public static void main(String[] args) {
       HumanBeing h1 = new HumanBeing();
       // using setters of HumanBeing
       h1.setHeight(1.65f);
       h1.setWeight(68);
       h1.setBmi(calculateBmi(h1));

       // using getters of HumanBeing
       System.out.println("Person has "+h1.getWeight()+" kgs and is "+h1.getHeight()+"
meters in height, which results in BMI of "+h1.getBmi());
   }

   public static float calculateBmi(HumanBeing h1){
       return h1.getWeight()/(h1.getHeight()*h1.getHeight());
   }
}
```

Run/compile your Code,

Output:

```
Person has 68.0 kgs and is 1.65 meters in height, which results in BMI of 24.977045
```

Submission Instructions:

Include the following deliverables in your submission -

- Submit your source code using the Start Assignment button in the top-right corner of the assignment page in Canvas.

CANVAS STAFF USE ONLY: Canvas Submission Guideline:

Instructions for Canvas Assignment Creation
Assignment Name: GLAB - 303.9.1 - Encapsulation in Java **Points:** 100 **Assignment Group**: Module 303: Java SE Review (Not Graded) **Display Grade As:** Complete/Incomplete **Do not count this assignment towards the final grade:** Checked **Submission Types**: File Upload **Everything else is the default.**