

# Lab 1

November 20<sup>th</sup>, 2017

## Schedule

Date	Time	Room
November 20 <sup>th</sup> , 2017	11:30 - 13:00	Q117 and Q120
November 20 <sup>th</sup> , 2017	13:15 - 14:45	Q117 and Q120
November 24 <sup>th</sup> , 2017	9:45 - 11:15	Q117 and Q120

**Students assigned to Room Q120 - please, don't forget your notebooks!**

The tasks on this sheet are intended to be done by students having no or very little experience in (procedural) programming. Other students may skip this task and proceed with lab 2 next week.

## Task 1.1 : BMI calculation

Write a program that calculates the body mass index (BMI) of a human. The BMI is a measure for a human's body shape (i.e. underweight, normal weight, overweight) based on his or her weight and height. It is calculated with the formula:

$$\text{BMI} = \frac{w}{h^2}, \quad w := \text{weight in kg}, \quad h := \text{height in m} \quad (1)$$

When the program is started, it queries weight and height from the user. Then it calculates the BMI and displays it to the user. Complete the following steps:

1. Create a new Eclipse project and add a class named `BmiCalculation` with a method `main()`, as in the *HelloWorld* example from the Eclipse tutorial.
2. Declare the needed variables given in Eq. (1) and choose appropriate data types.
3. Write statements that query weight and height from the user and assign them to the corresponding variables (mind the comments on the class `SimpleIO` below).
4. Add a statement to calculate the BMI.
5. Add a statement to display the calculated BMI to the user.
6. Test your program for several inputs.

Use the class (= module) `SimpleIO` for the interaction with the user. You need to download it from StudIP and add it to your project. You download it as file *SimpleIO.java*. Create a new package (*File*→*New*→*Package*) with the name `de.tuhh.diss.io`. You will see it in the package explorer on the left. Use the mouse to drag the file *SimpleIO.java* into the package in the project explorer.

At the beginning of each of your labs, import `SimpleIO` by typing the declaration:

```
import de.tuhh.diss.io.SimpleIO;
```

Then you can use its methods `SimpleIO.print()` and `SimpleIO.println()` to display data to the user. Methods such as `SimpleIO.readInteger()` and `SimpleIO.readDouble()` can be used to query data from the user, e.g.

```
myNewInteger = SimpleIO.readInteger();
```

queries an integer value from the user and assigns it to the variable `myNewInteger`. Note that you have to declare `myNewInteger` first.

### Task 1.2 : Check for correct input values

Since your BMI calculation might produce illogical output data if unexpected values of weight or height are entered, the program should be modified. Add if-statements to your program to check for illegal input data of the user, e.g. negative values for weight. Prompt the user again to enter a legal value and proceed with calculation. Print the correct result of your calculation.

### Task 1.3 : Advanced check for correct input values

What happens with your input check if a user enters an illegal value twice? Modify your input check by a suitable repetition-loop. The user should be prompted again and again until a legal value is entered. Proceed with the calculation and print the correct result to the console.

### Task 1.4 : Classification of results

Based on the BMI, you can rate if the entered person has underweight, normal weight or overweight:

Condition	Result
$\text{BMI} < 18.5$	underweight
$\text{BMI} \geq 25.0$	overweight
$18.5 \leq \text{BMI} < 25.0$	normal weight

Add two double constants to your program and protect them from modification within the program:

```
■ BMI_UPPERBOUND = 25.0
```

```
■ BMI_LOWERBOUND = 18.5
```

Use if-statements, the given table and the constants to perform classification. Display the results to the user. How many if-statements do you need?

### Task 1.5 : Read multiple values

Your program should now be modified to build an average over multiple BMI calculations. Your program should be able to handle several subsequent user inputs for height and weight, calculate the BMI per user without output and display the average BMI of all entered persons at the end.

Create a new integer constant `NUM_PERSONS` to determine the number of persons the user has to enter. Create a for-loop, which queries weight and height of multiple people. Within the loop, calculate the

BMI of the person and use it for average calculation at the end of your program. Think about reasonable values for initialization, condition and stepping of your for-loop. Do you have to store all entered values? Once the user entered the values of all different persons, calculate their average BMI and print it to the console. Test your program with at least `NUM_PERSONS = 2`.