ADNAN MENDERES UNIVERSITY
CSE 203 Object-Oriented Programming

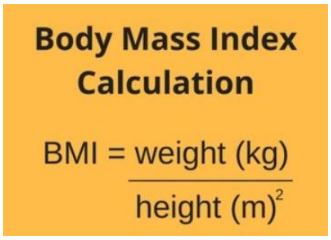
Lab 03

- You should submit **only 1 compressed folder:** ***NameSurname_studentNo_Lab3.zip***
- Do your homework in **ECLIPSE IDE**.
- Do not use Turkish Characters(ç,ğ,ı,ö,ş,ü) for naming project, methods, classes.
- Late submissions are not allowed.
- You should do homework **YOURSELF**. Group working is not allowed.
- Copy homework will be evaluated as 0.
- **DO NOT upload a screenshot or something else.**
- Use Google Classroom for your questions.

HOMEWORK

Computing Body Mass Index

Body Mass Index (BMI) is a measure of health based on height and weight. It can be calculated by taking your weight in kilograms and dividing it by the square of your height in meters. The interpretation of BMI for people 20 years or older is as follows:



BMI	Interpretation
BMI < 18.5	Underweight
18.5 ≤ BMI < 25.0	Normal
25.0 ≤ BMI < 30.0	Overweight
30.0 ≤ BMI	Obese

Computing Body Mass Index - Using Bmi Object Array (2 Classes: Bmi and BmiTest)

- Write your all code into 2 class: **Bmi.java** (without main method) and **BmiTest.java** (with main method)
- Use an **Array of Bmi** (array of reference type) instead of primitive type arrays.
  Array of Bmi contains references of Bmi objects.

Design a class named Bmi that contains:

- A private String data field named name that stores name of person
- A private int data field named age that stores age of person.
- A private double data field named weight in pounds for person.
- A private double data field named height in inches for person.
- A public static final double data field named KILOGRAMS_PER_POUND and its value: 0.45359237 (constant value).
- A public static final double data field named METERS_PER_INCH and its value: 0.0254 (constant value).
- A **no argument constructor** (constructor with no parameters) that creates a BMI. Assign default values to data fields in the constructor. The default values

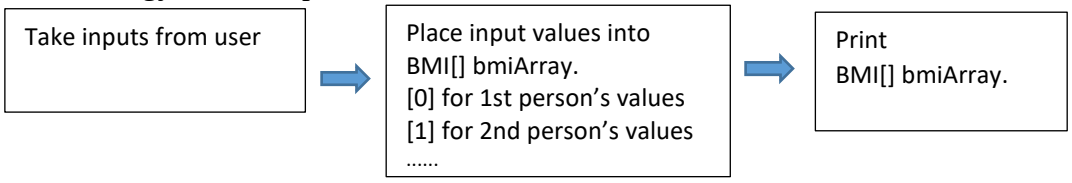
for name, age, weight and height fields are respectively: “John Black”, 25, 100 and 50.

- A second constructor that creates a BMI with the specified **name**, **weight** and **height (default age is 20)**.
- A third constructor that creates a BMI with the specified **name, age**, **weight** and **height**.
- The accessor (getter) and mutator (setter) methods for name, age, weight and height.
- A method named getBMI() that returns the BMI.(You may use Math.round() for returned value.)
- A method named getStatus() that returns the BMI status (e.g. normal, overweight, etc.)
- Note that one pound is 0.45359237 kilograms and one inch is 0.0254 meters.
- Do not use “**this**” keyword in the implementation of homework.

Design a class named BmiTest that contains main method. In the Class:

- Define Size (number of users whose bmi will be calculated) as 3
- **Create and initialize An Array of Bmi**
- Take and separate input values. Write **a function** for separating input values. Call that function in main function. You may use split() method of String class or indexOf() and substring() methods of String class.
- Place separated input values into Array of Bmi.
- Print Array of Bmi

Output is given below. In this case your code will take the benefit of the object-oriented methodology; less complex, extendible, re-usable, more robust and secure.



Sample run:

```
---ENTER PERSON 1'S VALUES---
Enter name, age, weight, height: (as space separated)
john black 25 100 50
---ENTER PERSON 2'S VALUES---
Enter name, age, weight, height: (as space separated)
sara king 20 215 70
---ENTER PERSON 3'S VALUES---
Enter name, age, weight, height: (as space separated)
kim young 21 145 70
** The BMI result for john black ( Age: 25 Weight: 100.0 Height: 50.0) is
Overweight
** The BMI result for sara king ( Age: 20 Weight: 215.0 Height: 70.0) is
Obese
** The BMI result for kim young ( Age: 21 Weight: 145.0 Height: 70.0) is
Normal
```