**Project Documentation**

**Project name:** BMI calculator

**Group members:** Hamza Hadžiabdić, Nejra Mujezinović, Benjamin Šukalo

**Table of Contents**

[Project Description 1](#_Toc181827816)

[Table 1: BMI classification table 1](#_Toc181827817)

[Functions/Methods 1](#_Toc181827818)

[Python 2](#_Toc181827819)

[C# 2](#_Toc181827820)

[JavaScript 2](#_Toc181827821)

# **Project Description**

A BMI (body mass index) calculator is a tool used as an indication of whether or not a person is of a healthy weight. This is achieved by using the following formula: **bodyweight (in kilograms)/height (in meters squared)**, or if one is using the Imperial system: **(bodyweight (in lbs) \* 703)/(height (in inches squared))**. The result of this formula is then compared against a table to determine which BMI classification a person falls under. The table that will be used in this project is shown below:

|  |  |
| --- | --- |
| **Classification** | **BMI range** |
| Underweight | < 18.5 |
| Normal | 18.5 – 24.9 |
| Overweight | 25.0 – 39.9 |
| Obese | > 39.9 |

### Table 1: BMI classification table

This particular project will begin by asking the user if they want to use the Imperial system or metric system to input their data. Once the user makes their choice, the program will perform the necessary conversions and calculations and compare that number to the aforementioned BMI table. Depending on the language, this comparison will be done using either switch statements or if/else statements in the case of Python where built-in switch statements do not exist. Finally, it will display the user’s BMI as well as inform them whether they are underweight, healthy, overweight, or obese. As this is a console application, this information will be displayed in the terminal, or in the case of JavaScript, the developer console.

# **Functions/Methods**

Seeing as how a BMI calculator is a very simple application, it will only use a few methods for each language.

## **Python**

input() built-in function for displaying a prompt and taking an input from the user

pow() built-in function for raising a number to the nth power

print() built-in function for displaying data on the terminal

## **C#**

Console.ReadLine() method that reads data from the standard input stream

Convert.ToSingle() method within the C# Convert class which converts the a specified value to a float

Math.Pow() method for raising a number to the nth power from the C# Math class

Console.WriteLine() method for displaying data on the standard output stream, and ends with a current line terminator

## **JavaScript**

prompt() method for displaying a prompt and taking an input from the user

Math.pow() method for raising a number to the nth power from the JavaScript Math namespace object

console.log() built-in method for displaying data on the developer console