PROGRAMMING ASSIGNMENT 1: BMI CALCULATOR (20 POINTS)

The BMI (Body Mass Index) formula is the result of investigations performed by Adolphe Quetelet, a Belgian scientist. Although the BMI formula was developed in the 19th century, it became more popular in the second part of the 20th century.

The BMI formula describes body conditions taking into account the weight and height of a person. The formulas for calculating BMI using metric units is:

\[
BMI = \frac{\text{kilograms}}{\text{height\_meters}^2}
\]

You will be writing a program that is based on weight entered in pounds and height entered as feet and inches. Therefore, you will need the following conversion facts:

1 pound = 0.45359237 kilograms.
1 inch = 0.0254 meters

Compose a program that reads in from the user

- a person’s weight in pounds, as well as height as a combination of two separate values of feet and inches

and outputs:

- The BMI for the entered weight and height data
- The height, as a number of total inches
- Assuming a healthy range for BMI is between 18.5 and 24.9, the healthy weight range for a given height expressed in inches, first in pounds and then in kilograms.

SAMPLE OUTPUT

- This section shows a sample run of the program. Values entered by the user are shown in boldfaced, green, italicized font, as in: Tom.

```
======== Body Mass Index Calculator ========
Please enter your name: Tom
Welcome, Tom!

Enter weight in pounds: 185
Enter height [feet]: 5
Enter height [inches]: 11

=============== Calculations ===============
The BMI is 25.8
For height of 71.0 inches a healthy weight range is between 133 and 179 lbs, which is between 60 and 81 kg.
```
OTHER REQUIREMENTS

1. Use symbolic constants, by convention written in all uppercase letters, to represent all the fixed values in this problem. **Do not hard-code any values in your code** where a variable or constant is appropriate.

2. Numbers are to be formatted appropriately, **using two decimal places as shown.**

3. **It is important that the order of inputs and outputs match the sample output as closely as possible with the shown formatting of numbers.**

SUBMISSION

Always test your program to make sure it works using a variety of cases. Once you have completed your homework submit your Python file on Blackboard.

GRADING

Your program should compile without syntax errors to receive any credit. If a part of your program is working, you will receive partial credit, but only if the program compiles without syntax errors.

<table>
<thead>
<tr>
<th>#</th>
<th>Criteria</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Correct number and order of inputs and outputs.</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Compute the calculations from the values provided in the Introduction</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>Formatting output with correct number of decimals, separator line of equal signs, new lines.</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>Comments and good programming style</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>Use of symbolic constants and commented out code as described in Requirement #1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>20</strong></td>
</tr>
</tbody>
</table>

REMINDERS

You may use the integrated development environment of your choice (PyCharm, Eclipse, Spider, VS Code, etc.) to complete this and any projects during the course, unless specifically instructed otherwise.

This project requires knowledge of only very basic mathematical operations on numeric values. When working on a project always make sure you understand the program requirements first, and then think about the algorithm that you will use and write out the steps. Only after you have
thought through the details of the algorithm and verified it on a few test cases, should you start working on its implementation in Python. Be sure to test your program to ensure the output matches the assignment. When working on the program it is important to learn to develop it gradually by implementing one logical step of the algorithm at a time and testing the program after implementing each step.