CS 1510: Intro to Computing - Fall 2017
Assignment 2: Working with selection statements

Due: Friday, September 14, 2017, by 11:59 p.m.

Introduction

This week we have been working with selection statements. Let's see if you understand this idea. Just like last week, in this assignment you will :

1. Look at the two customer requests below (step #1 in the software development process).
2. After you have studied the problems and prepared your problem solving solution you should write the code that solves each of the problems. Please make sure that you write the code in two separate files saved with the names given below.
3. Once you think you have the code working, **_TEST it thoroughly_**.

Notes and requirements:

1. Make sure that you save your programs in the correctly named files
2. Use meaningful variable names with the proper style (use_snake_casing)
3. Use meaningful constants and names where appropriate and use proper style (ALL_CAPS_AND_UNDERSCORES)
4. Every file containing python code that you submit should contain a header commen block containing three pieces of information as shown below:

```
"""
File: filename
Author: your-name
Description: one-line description of the file
"""
```

Customer Request #1 (weight.py)

You are contacted by Dr. Zoltan Pepper who is analyzing the overall fitness of olympic athletes. To help understand how they compare he has decided to use the concept of Body Mass Index (BMI).

BMI is a number calculated from a person's weight and height. It does not measure body fat directly, but research has shown that it correlates well with body fat calculations from more dependable measures such as underwater weighing. The formula for BMI is:

$$BMI = weight / height2$$

where weight is in kilograms and height is in meters.

For example, Peter Kirkbride (a weightlifter from the UK) is 74.5 inches tall and weighs 207 lbs.
This means his BMI is 26.2213925..

(Note, 1 inch is 2.54 centimeters. 1 pound is 0.453592 kg)

Dr. Pepper also points out the following table from the CDC:

BMI	Weight Status
Below 18.5	Underweight
Below 25 (but 18.5 or higher)	Normal
Below 30 (but 25 or higher	Overweight
30.0 and above	Obese

Dr. Pepper asks you to **write a program (called weight.py) that:**

- asks for the users weight in pounds and height in inches
- converts both of these to kilograms and meters
- calculates the BMI
- prints a message to the user containing BOTH their BMI number and their weight status based on the table above

Customer Request #2 (calculator.py)

Esther Mate's favorite subject at school has always been mathematics.
She asks you to **write a program (called calculator.py) that:**

- takes in two integers that are both between 0 and 100.
- takes in an operator to operate on the integers.
- prints the first error encountered then does not continue execution.
- *Errors include:*
    - either of the integer values is not between 0 and 100.
    - the operator is not of the type: +, -, *, /, //, %, **
- prints the correct calculation if the two integers and the operator are all valid.

For example:

```
What is the first integer value? 0
What is the second integer value? 100
What operation are you performing? +
The first integer value is invalid.
>>> ================================ RESTART ================================
>>>
What is the first integer value? 99
What is the second integer value? 100
What operation are you performing? *
The second integer value is invalid.
>>> ================================ RESTART ================================
>>>
What is the first integer value? 45
What is the second integer value? 34
What operation are you performing? ?
You entered an invalid operator.
>>> ================================ RESTART ================================
>>>
What is the first integer value? 12
What is the second integer value? 102
What operation are you performing? >
The second integer value is invalid.
>>> ================================ RESTART ================================
>>>
What is the first integer value? 3
What is the second integer value? 45
What operation are you performing? **
Answer: 2954312706550833698643
>>>
```

Hint: The operator will be coming in as a string, and you cannot use it directly in computations. You will need to determine if the string operator is one of the valid operators using conditional statements.

Be sure your program's output matches the above screenshot (the errors triggered and the order of questions asked) to receive the most points.

Submit your work

Submit both files **weight.py** and **calculator.py** to eLearning.

Remember, **no late work will be accepted.** Start early!