**Assignment 2 –** *Python Practice with User Interaction,* String Processing and While loop

**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Student number:** \_\_ \_\_ \_\_ \_\_ \_\_ \_\_

**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Student number:** \_\_ \_\_ \_\_ \_\_ \_\_ \_\_

***ASSIGNMENT MARK*: \_\_\_\_\_\_\_\_ / 50 Due Date: Friday, Feb. 16th**

As always, all your answers should use functions and every function should have a documentation string explaining the purpose of the function.

**Do not wait until the last minute to do this assignment in case you run into problems.**

Submit your assignment to the instructor by email at: drahmalki@gmail.com

**User Interaction**

You will write a few Python functions for practice with user interaction and string processing as well as while loop. In your program file, include a Python template comment at the top. Also include each of the following functions: getInteger(), getMathOp(), getYesNo(), doCalculation(), and calculator()

Function Name: **getInteger()**

**Parameters**:

string – A String which will be shown to the users to “prompt” them.

**Return Value**:

An integer.

**Description**:

Write a function that will accept one parameter, a string, and show it to the user. The function should then get input from the user and convert it into an integer. Return the integer value. If the user does not enter a proper integer, your function must tell them that they entered an invalid number using the warning: “Not a valid number, try again...”

Then, show the prompt again, and allow them to try and type another integer. Keep repeating until the user enters digits that can be successfully converted to an integer.

**HINT: <a string>.isdigit() will return True if the string is integer False otherwise**

**Test Cases**:

getInteger(“Please enter an integer:”)

Please enter an integer: 9

Function returns 9

getInteger(“Please enter an integer:”)

Please enter an integer: Hi there

Not a valid number, try again…

Please enter an integer: \*\*+-/\*

Not a valid number, try again…

Please enter an integer: 3

It returns 3

Function Name: **getMathOp()**

**Parameters**:

String - a String which will be shown to the users to “prompt” them.

**Return Value**:

A string with one of four values: “+”, “-”, “\*”, “/”, or “\*\*”

**Description**:

Write a function that takes in a string to use as a prompt and shows it to the user. Then get input from the user. If they entered a single sign out of the set \*\*,+,-,\*,/, return it. If they entered anything else, give them a warning: “You may only enter one of the characters: \*\* + - \* /” and then repeat until they enter a correct option.

**Test Cases**:

getMathOp(“Please enter one of the five permitted math operations:”)

Please enter one of the five permitted math operations: +

**If the user typed a valid math operation, it returns the string representing the math operation; in this case, getMathOp() returns +**

getMathOp(“Please enter one of the five permitted math operations:”)

Please enter one of the five permitted math operations: Hi there

You may only enter one of the characters: \*\*+-\*/

Please enter one of the five permitted math operations: \*\*+-/\*

You may only enter the one of the characters: \*\*+-\*/

Please enter one of the five permitted math operations: 17

You may only enter the one of the characters: \*\*+-\*/

Please enter one of the five permitted math operations: \*\*

The function returns \*\*

Function Name: **getYesNo**

**Parameters**:

string – a String which will be shown to the users to “prompt” them.

**Return Value**:

Boolean – True if the user entered yes

False if the user entered no

**Description**:

Write a function that takes in a string as a prompt to the user. It should display that string and get input from the user. If the user enters yes return True. If they enter no then, return False. If they enter anything else, give them the error “Not a valid answer, enter yes or no.” and then repeat the question until they give a valid answer.

**Test Cases**:

getYesNo(“Do you want to continue?”)

Do you want to continue? I don't know

Not a valid answer, enter yes or no.

Do you want to continue? Yes

The function returns True

Function Name: **doCalculation**

**Parameters**:

none

**Return Value**:

**Float** – Result of calculation.

**Description**:

Write a function that gets an integer, an operation, and a second integer from the user. After getting those three items (it may call other functions that you have written to do this work) it should perform the appropriate calculation and return the result. Note that your result must be converted to a float, even if it was an integer.

**Test Cases**:

doCalculation()

Enter your first number: 7

Enter your math operation: +

Enter your second number: 3

The function returns 10.0

Function Name: **calculator()**

**Parameters**:

none

**Return Value**:

none

**Description**:

Write a function that uses your other functions to allow the user to do multiple calculations. After each calculation, ask the user if they want to do another calculation.

Once the user says they do not want to do any more calculations, print a final message that states “Goodbye!”

**Test Cases**:

Perform the first calculation

Would you like to perform another calculation? (yes or no) **yes**

Perform the next calculation

Would you like to perform another calculation? (yes or no) no

Goodbye!

**Happy computing!**

**N.B:** Next, you are going to make sure you can use the submission properly. This will be how you turn in your assignments electronically. The submission procedure only accepts **zip files (NOT rar Files)**, so you are going to create a “Asg2\_YourFirstName\_YourStudentID.zip” file containing all the programs you just wrote as well as this document with your name and student number added to it. To create a “.zip” file, find the directory where the files are saved and select all of them. If you right-click on one of the selected files, you should have the option to create an archive by choosing “Send to” then “Compressed “zipped” Folder”. Rename the created “.zip” file and give it the name Asg2\_YourFirstName\_YourStudentID.zip.

***10 marks will be deducted if:***

1. ***the email did not have a full name and a student # in the subject field or***
2. ***the procedure for zipped file is not followed or***
3. ***Python template and documentation are missing or***
4. ***This document, with your names and student numbers on it, is not included with the zipped file***
5. ***Your classmate’s email is not included in CC or Bcc field of the email***

**A score of zero (0) will be given to everyone whose email is empty.**