Lab Exercise 5 (Part 2)

Focus

1. Library functions
2. User-defined value returning functions
3. Modules

This lab maps to learning objectives 1 through 4 in Competency Five Part 2 – Write a working program that uses Value Returning Function and Modules that store functions.

Part 2A: Building upon an Existing Solution

For this portion of the lab, ***you will reuse the program you wrote in Lab5 Part1 A***.

That means you will open the lab you wrote in the previous assignment and change it. You should NOT start from scratch. Also, all the requirements/prohibitions from the previous lab MUST also be included /omitted from this lab.

Redesign the solution in the following manner:

1. All the functions used are value-returning functions. **(LO 1)**
2. Put the functions in an **external module** and let the main program import the module. **(LO 4)**
3. Save the program as firstname_lastname_Lab5Ba.py where you will replace firstname and lastname with your actual first and last name.

Part 2B: Write Something New!

Write a complete and syntactically correct Python program to solve the following problem: Write a simple guessing game using the following specs:

1. Generate a random number between 1 and 1000. **(LO 1, 2, 3)**
2. Ask the player to guess the number. You will keep track of the number of guesses.

3. If the player’s guess is higher than the number generated then display the message “Too High!” If the player’s guess is within a 10-point difference of the number generated but higher than the number generated, then give the message “Getting warm but still high!”
4. If the player’s guess is lower than the number generated then display the message “Too Low!” If the player’s guess is within a 10 point difference of the number generated but lower than the number generated, then give the message “Getting warm but still Low!”
5. The program will keep taking guesses until the player guesses the number. **(LO 1)**
6. Once the player guesses the number, give them a congratulatory message like “You rock! You guessed the number in x tries!!” where x is the actual number of tries it took the player to guess the number. You can write any message as long as you include the number of tries in the message.
7. Once the player has guessed the number, ask them if they wish to play again. If they do then generate another random number and start the game over again. **(LO 3)**
8. All input to the program will be interactive from the keyboard.

Use the IDLE programming environment if you are using Python with IDLE. Some of you may be using Komodo or some other Python IDE. Please save your file as firstname_lastname_Lab5Bb.py where you will replace firstname and lastname with your actual first name and last name. Remember to use the extension .py.

Run and test your program for all conditions. Once you are sure it works you will turn in the items listed in the next section.

Turn In

All labs will be graded in Blackboard. Once you are done with the lab turn it in to the Lab 5 B link. Please read the How To Submit instructions if you have any questions or contact the instructor / academic coach. For this lab you will turn into Blackboard:

1. The Python *code files* you saved in part A (the main program and the imported module)
2. The Python code file you saved in part B