**Yashwant Desai – Assignment 11**

1. Create an assert statement that throws an AssertionError if the variable spam is a negative integer.

Answer: assert spam >= 0

2. Write an assert statement that triggers an AssertionError if the variables eggs and bacon contain strings that are the same as each other, even if their cases are different (that is, 'hello' and 'hello' are considered the same, and 'goodbye' and 'GOODbye' are also considered the same).

Answer: Either assert eggs.lower() != bacon.lower() 'The eggs and bacon variables are the same!' or assert eggs.upper() != bacon.upper(), 'The eggs and bacon variables are the same!'

3. Create an assert statement that throws an AssertionError every time.

Answer: assert False, 'This assertion always triggers.'

4. What are the two lines that must be present in your software in order to call logging.debug()?

Answer:

import logging

logging.basicConfig(level=logging.DEBUG, format=' %(asctime)s -

%(levelname)s -  %(message)s')

5. What are the two lines that your program must have in order to have logging.debug() send a logging message to a file named programLog.txt?

Answer:

logging.basicConfig(filename='programLog.txt', level=logging.DEBUG,

format=' %(asctime)s -  %(levelname)s -  %(message)s')

6. What are the five levels of logging?

Answer: the five levels of logging are debug, info, warning, error and critical

7. What line of code would you add to your software to disable all logging messages?

Answer: logging.disable(logging.CRITICAL)

8.Why is using logging messages better than using print() to display the same message?

Answer: Once we're done debugging, we'll end up spending a lot of time removing print() calls from our code for each log message. We might even accidentally remove some print() calls that were being used for nonlog messages.

9. What are the differences between the Step Over, Step In, and Step Out buttons in the debugger?

Answer: When using the debugger, the STEP command will execute the next line of code and then pause again. If the next line of code happens to be a function call, the debugger will "step into" that function and move to the first line of code inside that function. On the other hand, the OVER command executes the next line of code, similar to the STEP button. However, if the next line of code is a function call, the OVER button will "step over" the code within the function without going into its details. Lastly, the OUT command will allow the debugger to execute lines of code at full speed until it returns from the current function.

10.After you click Continue, when will the debugger stop ?

Answer: Clicking the Continue button will cause the program to execute normally until it terminates or reaches a breakpoint.

11. What is the concept of a breakpoint?

Answer: A breakpoint can be set on a specific line of code and forces the debugger to pause whenever the program execution reaches that line.

**Regards,**

**Yashwant**