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

spam = -6

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).**

eggs = 'goodbye'

bacon = 'GOODbye'

assert eggs.lower()==bacon.lower()

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

assert False

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

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?**

import logging

logging.basicConfig(filename = 'debug.log',level = logging.DEBUG, format = '%(asctime)s %(levelname)s %(message)s')

**6. What are the five levels of logging?**

DEBUG, INFO, WARNING, ERROR,CRITICAL

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

logging.disable(logging.CRITICAL)

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

We can disable logging messages without removing the logging function calls by logging.disable(logging.CRITICAL).

We can create our own custom logging messages with times stamp which helps us better in debugging and disable or enable lower-level logging message according to requirement.

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

Step In: When the next statement to execute reaches a method call, don’t execute the method as a whole, but rather, execute the first line of that method and stop

Step Over: When the next statement to execute reaches a method call, execute the method as a whole and stop

Step Out: quickly execute the rest of the code until it steps out of the function it currently is in.

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

It execute until the last break point.

**11. What is the concept of a breakpoint?**

It is a line of code that cause debugger to pause when the execution reaches the line.