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

```assert spam >= 0, 'spam should be a non-negative integer'```

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).

```assert eggs.lower() != bacon.lower(), 'eggs and bacon should not have the same value, even if their cases are different'```

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

```assert False, 'this assertion always triggers an AssertionError'```

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

```import logging``` and ```logging.basicConfig(level=logging.DEBUG)```

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

6. What are the five levels of logging?

The five levels of logging, in increasing order of severity, are DEBUG, INFO, WARNING, ERROR, and 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?

Using logging messages allows for more fine-grained control over which messages are displayed, and where they are displayed (console, file, etc.). It also makes it easier to turn off or modify messages without having to manually remove or change print statements throughout the code.

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

The Step Over button allows the code to execute one line at a time, without entering into function calls. The Step In button enters into a function call and executes it one line at a time. The Step Out button executes the lines of code in a function call until it returns to the line that called the function.

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

The debugger will stop again either when another breakpoint is encountered or when the program finishes executing.

11. What is the concept of a breakpoint?

A breakpoint is a point in the code where the debugger will stop executing and allow the programmer to examine the state of the program at that point. Breakpoints can be set at specific lines of code, and can be conditional (i.e. they will only trigger if a certain condition is met).