1. Ans. assert(spam >= 0, 'The spam variable is an negative value.')
2. Ans.  assert(eggs.lower() != bacon.lower(), 'The eggs and bacon variables are the same!')
3. Ans. assert(False, 'This assertion always triggers.')
4. Ans. import logging logging.basicConfig(level=logging.DEBUG, format=' %(asctime)s - %(levelname)s - %(message)s')
5. Ans. import logging logging.basicConfig(filename='programLog.txt', level=logging.DEBUG, format=' %(asctime)s - %(levelname)s - %(message)s')
6. Ans. DEBUG, INFO, ERROR, WARNING and CRITICAL
7. Ans. logging.disable(logging.CRITICAL)
8. Ans . You can disable logging messages without removing the logging function calls. Logging messages provides a timestamp.
9. Ans. The Step button will move the debugger into a function call. The Over button will quickly execute the function call without stepping into it. The Out button will quickly execute the rest of the code until it steps out of the function it currently is in.
10. Ans. click Go, the debugger will stop when it has reached the end of the program or a line with a breakpoint.
11. Ans. breakpoint is a setting on a line of code that causes the debugger to pause when the program execution reaches the line.