**ASSIGNMENT -11**

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

**ANS. the assert** statement is used to continue the execute if the given

condition evaluates to True. If the assert condition evaluates to False,

then it raises the AssertionError exception with the specified error

message.

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

**ANS.**

In order to catch the assertion error, we need to **declare the assertion statement in** the try block with the second expression being the message to be displayed and catch the assertion error in the catch block.

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

**ANS**. Reproduce the problem.

Describe the bug. Try to get as much input from the user to get the exact reason.

Capture the program snapshot when the bug appears. ...

Analyse the snapshot based on the state and action. ...

Fix the existing bug, but also check that any new bug does not occur.

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

**ANS.** Logging levels explained. The most common logging levels include FATAL, ERROR,

WARN, INFO, DEBUG, TRACE, ALL, and OFF.

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

**ANS. logger = logging.** **getLogger('my-logger') logger.** **propagate = False** # now if you use logger it will not log to console. This will prevent logging from being send to the upper logger that includes the console logging.

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

**ANS.** Also, **logs are configurable**. You can easily filter them, send them to files, format them, add timestamps, and any other things you might need on a global basis. Print statements are not easily managed. Definitely +1 for sending output to files.