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

Answer :

**Assert statement for negative integer:**

python

Copy code

assert spam >= 0, "spam should not be a negative integer"

1. 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 :

assert eggs.lower() != bacon.lower(), "eggs and bacon should not be the same, ignoring case"

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

Answer :

assert False, "This assertion always fails"

1. 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)

1. 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 :

import logging

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

1. What are the five levels of logging?

Answer :

**Five levels of logging:**

1. DEBUG
2. INFO
3. WARNING
4. ERROR
5. CRITICAL
6. 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 :

**Control**: Logging allows you to set different levels of importance (DEBUG, INFO, WARNING, ERROR, CRITICAL) and control what gets recorded.

**Flexibility**: You can easily redirect logs to different destinations (files, consoles, remote servers) and format them.

**Performance**: Logging can be configured to be turned off or filtered without changing the code.

**Consistency**: Provides a standardized approach to error reporting and debugging.9. What are the differences between the Step Over, Step In, and Step Out buttons in the debugger?

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

Answer :

**Step Over**: Executes the current line of code and moves to the next line, skipping over function calls without stepping into them.

**Step In**: Steps into the function call on the current line, allowing you to debug inside that function.

**Step Out**: Continues execution until the current function exits, returning to the caller.

11. What is the concept of a breakpoint?

Answer :

A breakpoint is a marker set in the code where the debugger will pause execution, allowing you to inspect the current state, variables, and control flow.