

## QL-5) – Untestable Code: Key Observations

### Activity 1

After **refactoring** the code and beginning to design the test, you may realise there is no state or return value available for an assert statement.

This is a crucial learning point. Most testing relies on calling methods that return a value, which can then be validated using assertions. However, in this case, the method `process_data()` is a void - it doesn't return anything. So how can we verify its behaviour?

The answer lies in the type of testing we apply. We need to move away from **black box testing**, where we treat the method as a sealed unit - passing inputs and expecting outputs - without considering internal workings. Instead, we must apply **white box testing**, where we treat the method as transparent. This allows us to identify and insert test hooks or features within the method, enabling it to provide observable feedback to the unit test.