**Which one of the following refers to an end to end test?**

a) Unit test b) Aggregate test c) System test

**Which statement is true?**

a) JUnit is an open source framework to build web apps.

b) JUnit is a framework to help create unit mapping in a database.

c) JUnit is a framework to create unit tests.

**Is it true that the convention in Java is to create a unit test class associated with whatever class you are trying to test? Please elaborate on answer.**

That is correct, in Java it is recommended to store all the test classes in a package usually with the name Test, the test classes in this package should be named after the class that is being tested. For example, you have the class MyClass you should name it MyClassTest and should be stored in the Test package.

**Is it better to name the test methods according to the number order to keep methods name short and organized? i.e. testMethod1, testMethod2, etc. Please elaborate on answer.**

This is not recommended because the possible problems that it could cause in the future if too many tests are created so it will be difficult to keep track of every test. The tests should be given more representative names according to the method that is being tested.

**Talking about the structure of tests, what does the ‘given’ clause establish?**

a) The preconditions for the test.

b) The solution we are trying to get after the testing.

c) The steps needed to confirm the test was successful.

**The ‘then’ clause can be considered as the asserts section in our code? Please elaborate on answer.**

Yes, because this statement checks that the post condition holds by asserting values

**You should test the implementation rather than the behavior of your application? Please elaborate on answer.**

Its better to test behavior instead of implementation, because it’s the result of the implementation what matters the most, also tests that are independent of implementation are easier to maintain and understand.

**Accessing private members of your class while testing is considered sometimes as a good practice as it allows the developer to test deeper in the code? Please elaborate on answer**

Testing private members of a class is not recommended and thus not a good practice because this are not part of the public interface, so they don’t expose public behavior

**Which statement is true?**

a) TDD tries to use tests to drive the design of software.

b) TDD encourage good practices instead of focusing just on code reliability.

c) Force the refactoring of code only when there are tests failing rather than over-designing solutions.

d) All of above.

**One of the main characteristics of TDD is to write tests even before writing the actual code. Please elaborate on answer.**

Correct, In TDD each new functionality that is going to be added start with a test, so the developer can focus on the requirements before even writing the first line of code.