

0.1 JUnit

0.1.1 Description of JUnit

JUnit is a unit testing framework for the Java programming language. JUnit has been important in the development of test-driven development, and is one of a family of unit testing frameworks which is collectively known as xUnit that originated with SUnit.

JUnit is linked as a JAR at compile-time; the framework resides under package *junit.framework* for JUnit 3.8 and earlier, and under package *org.junit* for JUnit 4 and later.

0.1.2 Test fixture of JUnit

A JUnit test fixture is a Java object. With older versions of JUnit, fixtures had to inherit from *junit.framework.TestCase*, but the new tests using JUnit 4 should not do this. Test methods must be annotated by the *@Test* annotation. If the situation requires it, it is also possible to define a method to execute before (or after) each (or all) of the test methods with the *@Before* (or *@After*) and *@BeforeClass* (or *@AfterClass*) annotations.

We have adopted *JUnit version 4.12* and the library is from *Maven* remote repository. In our implementation of JUnit test fixture, we have well utilized the *@Before* and *@After* method to initialize and discard objects that will be used in all other *@Test* methods.

0.2 JMock

JMock is a library that supports test-driven development of Java code with mock objects.

Mock objects help you design and test the interactions between the objects in your programs.

The jMock library:

1. makes it quick and easy to define mock objects, so you don't break the rhythm of programming.
2. lets you precisely specify the interactions between your objects, reducing the brittleness of your tests.
3. works well with the autocompletion and refactoring features of your IDE
4. plugs into your favourite test framework is easy to extend.