2CSDE80 Software Testing and Quality Assurance
Practical 4
18BCE106 Divya Mahur

Aim: To study and perform sample tests using the J-Unit Testing tool

JUnit Testing

JUnit is a Java unit testing framework that's one of the best test methods for regression testing. An open-source framework is used to write and run repeatable automated tests. As with anything else, the JUnit testing framework has evolved over time. The major change to make note of is the introduction of annotations that came along with the release of JUnit 4, which provided an increase in the organization and readability of JUnits.

Annotations:

@Test annotation specifies that method is the test method.

@Test(timeout=1000) annotation specifies that method will be failed if it takes longer than 1000 milliseconds (1 second).

@BeforeClass annotation specifies that method will be invoked only once, before starting all the tests.

@Before annotation specifies that method will be invoked before each test.

@After annotation specifies that method will be invoked after each test.

@AfterClass annotation specifies that method will be invoked only once, after finishing all the tests.

JUnit Framework can be easily integrated with either of the following −

- Eclipse
- Ant
- Maven

JUnit test framework provides the following important features −
Fixtures

- Test suites
- Test runners
- JUnit classes

JUnit Classes

JUnit classes are important classes, used in writing and testing units. Some of the important classes are −

- Assert − Contains a set of assert methods.
- TestCase − Contains a test case that defines the fixture to run multiple tests.
- TestResult − Contains methods to collect the results of executing a test case.

Code:

JunitTest.java (Program Logic)

```
package UnitTesting;

public class JunitTest {
	public int square(int n) {
		return n*n;
	}
	public int findMin(int[] arr) {
		int min=Integer.MAX_VALUE;
		for(int i=0;i<arr.length;i++)
		{
			if(min>arr[i])
				min=arr[i];
		}
		return min;
	}

	public int findMax(int[] arr) {
		int max=Integer.MIN_VALUE;
		for(int i=0;i<arr.length;i++)
		{
			if(max<arr[i])
				max=arr[i];
```

```
                }
                return max;
        }
}
```

test.java (Test-case)

```
package UnitTesting;

import static org.junit.Assert.*;

import org.junit.Test;

public class squareTest {

        @Test
        public void test() {
                JunitTest ob1 = new JunitTest();
                int output=ob1.square(6);
                // Testing output
                assertEquals(36,output);
        }

        @Test
        public void testMin() {
                JunitTest obj1 = new JunitTest();
                int output_f=obj1.findMin(new int[] {20,30,10,1});

                //test
                assertEquals(1,output_f);
        }

        @Test
        public void testMax() {
                JunitTest obj1 = new JunitTest();
                int output_f=obj1.findMax(new int[] {20,30,10,1});

                //test
                assertEquals(30,output_f);
        }

}
```