**Listeners - Context, iTest, iTestResults**

**ItestListenerClass.java**

package ListenerDemo;

import org.testng.ITestContext;

import org.testng.ITestListener;

import org.testng.ITestResult;

public class ItestListenerClass implements ITestListener{

public void onTestStart(ITestResult result) {

// if a @test method is started, it will record the log

System.out.println("Test method has been started");

}

public void onTestSuccess(ITestResult result) {

// if testmethod is success

System.out.println("Test method is success");

}

public void onTestFailure(ITestResult result) {

// if testmethod is failure

System.out.println("Test method is failure");

System.out.println("screenshot code");

}

public void onTestSkipped(ITestResult result) {

System.out.println("Test method is skipped");

}

public void onTestFailedWithTimeout(ITestResult result) {

System.out.println("Test method is failure due to timeout");

}

public void onStart(ITestContext context) {

// this method will record events in the very begining of testing

System.out.println("Testing has started");

}

public void onFinish(ITestContext context) {

// this method will record events at end of testing

System.out.println("Testing has ended");

}

}

**TestDemoListener.java**

package ListenerDemo;

import org.testng.Assert;

import org.testng.annotations.Listeners;

import org.testng.annotations.Test;

@Listeners(ItestListenerClass.class)

public class TestDemoListener {

@Test(priority='1')

public void method1()

{

System.*out*.println("Executing method1");

}

@Test(priority='2')

public void method2()

{

System.*out*.println("Executing method2");

Assert.*assertTrue*(false);

}

}