Listeners in TestNG

Interface ITestListeners

```
package listernersDemo;
import org.testng.ITestContext;
import org.testng.ITestListener;
import org.testng.ITestResult;
public class ItestListernerClass  implements
ITestListener{
    @Override

    //Invoked each time before a test will be
invoked.
    public void onTestStart(ITestResult result) {

        System.out.println("Test has been Invoked");

    }
    //Invoked each time a test succeeds.
    @Override
    public void onTestSuccess(ITestResult result) {

        System.out.println("Log : Test has been
successfull");
    }

   // Invoked each time a test fails.
    @Override
    public void onTestFailure(ITestResult result) {

        System.out.println("Log: Test has failed");
    }
    @Override
    public void onTestSkipped(ITestResult result) {
        System.out.println(" Log: Test has been
skipped");
    }
    @Override
    public void onTestFailedWithTimeout(ITestResult
result) {
        System.out.println("Log: Test has been
failed due to timeout");
    }

//  Invoked before running all the test methods
belonging to the classes inside the <test> tag
    // and calling all their Configuration methods.

    @Override
    public void onStart(ITestContext context) {

        System.out.println("The Main test has
started");
    }
    //Invoked after all the test methods belonging
to the classes inside the <test> tag have run
    // and all their Configuration methods have been
called.
    @Override
    public void onFinish(ITestContext context) {
        System.out.println("The Main test has
Completed");
    }


}
package listernersDemo;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Listeners;
import org.testng.annotations.Test;
// @Listeners(ItestListernerClass.class)

public class TestMethodsdemo {

    public static WebDriver driver;

    @BeforeTest
    public void OpenBrowser()
    {
        driver = new ChromeDriver();
    }

    @AfterTest
    public void Closebrowser()
    {
        driver.close();
    }


    @Test(priority='1')

    public void method1()
    {
        driver.get("https://www.amazon.in/");
        System.out.println(driver.getTitle());

    }
    // this will also pass
    @Test(priority='2')
    public void method2()
    {
        WebDriver driver = new ChromeDriver();

driver.get("https://www.selenium.dev/downloads/");
        System.out.println(driver.getTitle());
    } // This will pass

    @Test(priority='3')
    public void method3()
    {


driver.get("https://www.selenium.dev/documentation/"
);
        String expectedtitle = "Selenium";
        String actualTitle = driver.getTitle();

Assert.assertEquals(actualTitle,expectedtitle); //
test will failed as titles will not match.
    }
    //this test will fail due to assertion


    @Test(priority='4')
    public void method4()
    {


driver.get("https://www.selenium.dev/projects/");
        System.out.println(driver.getTitle());
        Assert.assertTrue(false);

    }  // this will fail due to assertion

    //The maximum number of milliseconds this test
should take to complete.
    //If it hasn't returned after thistime, it will
be marked as a FAIL

    @Test(priority='5',timeOut = 1000)
    public void method5() throws
InterruptedException
    {
        Thread.sleep(3000);   // wait for 3 sec, test will not complete in given time

driver.get("https://www.selenium.dev/projects/");


    }   // test will fail with timeout error
}
```

===================

```
package listernersDemo;


import org.testng.annotations.AfterClass;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Listeners;

import org.testng.annotations.Test;



public class Demo1 {


    // that means the connectDB() method will get executed as the first method of this class

    // this method will be executed only once in the beginning
    // this method will get executed before execution of any test
method starts



    @BeforeClass

    public void connectDB()

    {

        //code

    }


    // that means the closeDB() method will get executed as the last
method of this class

  // this method will be executed only once in the end

    // this method will get executed after execution of all test method
completes


    @AfterClass

    public void clossDB()

    {

        //code

    }



    @BeforeMethod
    public void openbrowser()
    {
        // code
    }

    @AfterMethod
    public void closeBrowser()
    {
        // code
    }


    @Test(priority='1')
    public void method1()
    {
        System.out.println("Selenium Code");
    }


    // this method will not be executed
    @Test(priority='2', dependsOnMethods = {"method1"},
enabled=false )
    public void method2()
    {

        System.out.println("Selenium Code number 2");

    }




}
```

==========================

```
<!DOCTYPE suite SYSTEM
"https://testng.org/testng-1.0.dtd">
<suite name="TestNGRunner1" verbose="1">
    <listeners>
        <listener
class-name="listernersDemo.ItestListernerClass"></listener>
    </listeners>

 <test name="TestCycle-1">
   <classes >
      <class name="listernersDemo.TestMethodsdemo"/>
      <class name="listernersDemo.Demo1"/>
   </classes>
 </test>
  </suite>
```