**Introduction to TestNG :**

In selenium using java there are two framework available:

1)JUnit

2)TestNG

TestNG is a testing framework design to simplify a broad range of testing

needs from unit testing to system testing

TestNG is an open source framework where NG stand for Next Genaration

TestNG is inspired from JUnit

Main method is not used for TestNG programs.

TestNG programs contains only methods that contain @Test Annotation .

if we don't write @Test annotation then this method will not execute.

Advantages of TestNG

1)TestNG annotation are easy to create Test cases.

2)Test cases can be grouped and prioterized more easily.

3)Exicute multiple programs / classes using xml.

4)Generate HTML reports.

5)Parallel test execution is possible.

---------------------------------------------------

Simple Program

package TestNG;

import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {

@Test

public void verifyTitle() {

Assert.assertEquals("Yahoo", "Yahoo");

}

@Test

public void abcd() {

Assert.assertEquals("Gmail", "Gmail1");

}

@Test

public void abc() {

Assert.assertEquals("Gmail", "Gmail");

}

}

TestNG Priority

package TestNG;

import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {

@Test(priority=2)

public void verifyTitle() {

System.out.println("Verify Title method");

}

@Test(priority=1)

public void abcd() {

System.out.println("abcd method"); }

@Test(priority=3)

public void abc() {

System.out.println("abc method"); }

}

--------------------------------------------------------------------------------

dependsOnMethods :

1)If first method is depends on second method ,if second method failed then first method will get skipped.

2)Only that two(which having reference of "dependsOnMethods") method will be exicuted

Program:

package TestNG;

import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {

@Test

public void login() {

System.out.println("Login successfully");

}

@Test

public void logout() {

System.out.println("Logout successfully");

}

@Test(dependsOnMethods= {"advancedSearch"})

public void search() {

System.out.println("Search successfully");

}

@Test

public void advancedSearch() {

Assert.assertEquals("Gmail", "Gmail1");

}

}

alwaysRun=true :

for ignoring the dependency of methods

package TestNG;

import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {

@Test

public void login() {

System.out.println("Login successfully");

}

@Test

public void logout() {

System.out.println("Logout successfully");

}

@Test(dependsOnMethods= {"advancedSearch"},alwaysRun=true)

public void search() {

System.out.println("Search successfully");

}

@Test

public void advancedSearch() {

Assert.assertEquals("Gmail", "Gmail1");

}

}

1)@BeforeMethod

2)@AfterMethod

1)@BeforeMethod

This method exicute before each methods

2)@AfterMethod

This method exicute after each methods

Ex:

package TestNG;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {

@BeforeMethod

public void login() {

System.out.println("Login successfully");

}

@AfterMethod

public void logout() {

System.out.println("Logout successfully");

}

@Test(priority=2)

public void addProduct() {

System.out.println("Add product successfully");

}

@Test(priority=1)

public void addVendor() {

System.out.println("Add vendor successfully");

}

@Test(priority=3)

public void addCurrency() {

System.out.println("Add currency successfully");

}

}

------

Output:

[RemoteTestNG] detected TestNG version 7.4.0

Login successfully

Add vendor successfully

Logout successfully

Login successfully

Add product successfully

Logout successfully

Login successfully

Add currency successfully

Logout successfully

PASSED: addProduct

PASSED: addVendor

PASSED: addCurrency

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

Default test

Tests run: 3, Failures: 0, Skips: 0

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

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

Default suite

Total tests run: 3, Passes: 3, Failures: 0, Skips: 0

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

---------------------------------------------------------------------------------------------------------------

3)@BeforeClass

4)@AfterClass

3)@BeforeClass

This method is exicute one time before the class.

4)@AfterClass

This method is exicute one time after the class.

Ex:

package TestNG;

import org.testng.annotations.AfterClass;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Test;

import junit.framework.Assert;

public class ClassName1 {

@BeforeClass

public void login() {

System.out.println("Login successfully");

}

@AfterClass

public void logout() {

System.out.println("Logout successfully");

}

@Test(priority=2)

public void addProduct() {

System.out.println("Add product successfully");

}

@Test(priority=1)

public void addVendor() {

System.out.println("Add vendor successfully");

}

@Test(priority=3)

public void addCurrency() {

System.out.println("Add currency successfully");

}

}

-----------

Output:

[RemoteTestNG] detected TestNG version 7.4.0

Login successfully

Add vendor successfully

Add product successfully

Add currency successfully

Logout successfully

PASSED: addVendor

PASSED: addCurrency

PASSED: addProduct

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

Default test

Tests run: 3, Failures: 0, Skips: 0

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

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

Default suite

Total tests run: 3, Passes: 3, Failures: 0, Skips: 0

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

5)@BeforeTest

6)@AfterTest

5)@BeforeTest

This method exicute once before all classes.

6)@AfterTest

This method exicute once after all classes.

Ex:

Program: 1:

package TestNG;

import org.testng.Assert;

import org.testng.annotations.AfterClass;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Test;

public class DependsOnMethods {

@BeforeClass

public void login() {

System.out.println("Login successfully");

}

@AfterClass

public void logout() {

System.out.println("Logout successfully");

}

@Test(priority=2)

public void addProduct() {

System.out.println("Add product successfully");

}

@Test(priority=1)

public void addVendor() {

System.out.println("Add vendor successfully");

}

@Test(priority=3)

public void addCurrency() {

System.out.println("Add currency successfully");

}

}

-------------------

Program 2:

package TestNG;

import org.testng.Assert;

import org.testng.annotations.Test;

public class Abc {

@Test

public void verifyTitle() {

Assert.assertEquals("Yahoo", "Yahoo");

}

@Test

public void abcd() {

Assert.assertEquals("Gmail", "Gmail1");

}

@Test

public void abc() {

Assert.assertEquals("Gmail", "Gmail");

}

}

----------

Program 3:

package TestNG;

import org.testng.annotations.AfterClass;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.BeforeTest;

import org.testng.annotations.Test;

public class Priority {

@BeforeTest

public void login() {

System.out.println("Test Login successfully");

}

@AfterTest

public void logout() {

System.out.println("Test Logout successfully");

}

@Test(priority=2)

public void verifyTitle() {

System.out.println("Verify Title method");

}

@Test(priority=1)

public void abcd() {

System.out.println("abcd method"); }

@Test(priority=3)

public void abc() {

System.out.println("abc method");

}

}

------------

Suite file is:

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">

<suite name="Suite">

<test thread-count="5" name="Test">

<classes>

<class name="TestNG.DependsOnMethods"/>

<class name="TestNG.Abc"/>

<class name="TestNG.Priority"/>

</classes>

</test> <!-- Test -->

</suite> <!-- Suite -->

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

Output:

[RemoteTestNG] detected TestNG version 7.4.0

Test Login successfully

Login successfully

Add vendor successfully

Add product successfully

Add currency successfully

Logout successfully

abcd method

Verify Title method

abc method

Test Logout successfully

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

Suite

Total tests run: 9, Passes: 8, Failures: 1, Skips: 0

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

----------------------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------------------------------------------

7)@BeforeSuite

8)@AfterSuite

7)@BeforeSuite

This method exicute once before @BeforeTest method

8)@AfterSuite

This method exicute once after @AfterTest method

Program 1:

package TestNG;

import org.testng.Assert;

import org.testng.annotations.AfterClass;

import org.testng.annotations.AfterMethod;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Test;

public class DependsOnMethods {

@BeforeClass

public void login() {

System.out.println("Login successfully");

}

@AfterClass

public void logout() {

System.out.println("Logout successfully");

}

@Test(priority=2)

public void addProduct() {

System.out.println("Add product successfully");

}

@Test(priority=1)

public void addVendor() {

System.out.println("Add vendor successfully");

}

@Test(priority=3)

public void addCurrency() {

System.out.println("Add currency successfully");

}

}

----------

Program 2:

package TestNG;

import org.testng.Assert;

import org.testng.annotations.Test;

public class Abc {

@Test

public void verifyTitle() {

Assert.assertEquals("Yahoo", "Yahoo");

}

@Test

public void abcd() {

Assert.assertEquals("Gmail", "Gmail1");

}

@Test

public void abc() {

Assert.assertEquals("Gmail", "Gmail");

}

}

--------------

Program 3:

package TestNG;

import org.testng.annotations.AfterClass;

import org.testng.annotations.AfterSuite;

import org.testng.annotations.AfterTest;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.BeforeSuite;

import org.testng.annotations.BeforeTest;

import org.testng.annotations.Test;

public class Priority {

@BeforeSuite

public void b\_suite() {

System.out.println("This is @BeforeSuite method");

}

@AfterSuite

public void a\_suite() {

System.out.println("This is @AfterSuite method");

}

@BeforeTest

public void login() {

System.out.println("Test Login successfully");

}

@AfterTest

public void logout() {

System.out.println("Test Logout successfully");

}

@Test(priority=2)

public void verifyTitle() {

System.out.println("Verify Title method");

}

@Test(priority=1)

public void abcd() {

System.out.println("abcd method"); }

@Test(priority=3)

public void abc() {

System.out.println("abc method");

}

}

========

Suite file is:

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">

<suite name="Suite">

<test thread-count="5" name="Test">

<classes>

<class name="TestNG.DependsOnMethods"/>

<class name="TestNG.Abc"/>

<class name="TestNG.Priority"/>

</classes>

</test> <!-- Test -->

</suite> <!-- Suite -->

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

Output:

[RemoteTestNG] detected TestNG version 7.4.0

This is @BeforeSuite method

Test Login successfully

Login successfully

Add vendor successfully

Add product successfully

Add currency successfully

Logout successfully

abcd method

Verify Title method

abc method

Test Logout successfully

This is @AfterSuite method

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

Suite

Total tests run: 9, Passes: 8, Failures: 1, Skips: 0

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

-----------------------------------------------------------------------------------------------------------------

Sequence of annotations:

1.@BeforeSuite

2.@BeforeTest

3.@BeforeClass

4.@BeforeMethod

5.@Test(as per priority)

6.@AfterMethod

7.@AfterClass

8.@AfterTest

9.@AfterSuite

Grouping test cases:

In grouping we make the group of test cases,and access those test cases from xml file by

mentioning the group name which test case we required.

XML file syntax for grouping is:

Program 1:

package TestNG\_Grouping;

import org.testng.annotations.Test;

public class ClassName1 {

@Test(groups= {"sanity","regression"},priority=1)

public void login() {

System.out.println("Login successfully");

}

@Test(groups= {"sanity","regression"},priority=10)

public void *logout*() {

System.out.println("Logout successfully");

}

@Test(groups= {"sanity"},priority=4)

public void search() {

System.out.println("Search successfully");

}

@Test(priority=2)

public void addVendor() {

System.out.println("Add vendor successfully");

}

@Test(groups= {"regression"},priority=3)

public void advancedSearch() {

System.out.println("Advanced search successfully");

}

@Test(groups= {"sanity","regression"},priority=5)

public void prepaidRecharge() {

System.out.println("Prepaid recharge successfully");

}

@Test(groups= {"regression"},priority=6)

public void billPayments() {

System.out.println("Bill payment successfully");

}

}

=====

xml file:

<?xml version="1.0" encoding="UTF-8"?>

<suite name="Suite" parallel="false">

<test name="Test">

<groups>

<run>

<include name ="sanity"/>

</run>

</groups>

<classes>

<class name="TestNG\_Grouping.ClassName1"/>

</classes>

</test> <!-- Test -->

</suite> <!-- Suite -->

=======

Output:

[RemoteTestNG] detected TestNG version 7.4.0

[TestNGContentHandler] [WARN] It is strongly recommended to add "<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd" >" at the top of the suite file [C:\Users\Admin\eclipse-workspace\April Batch\src\TestNG\_Grouping\ClassName1.xml] otherwise TestNG may fail or not work as expected.

Login successfully

Search successfully

Prepaid recharge successfully

Logout successfully

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

Suite

Total tests run: 4, Passes: 4, Failures: 0, Skips: 0

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

\* Parallel Test Exicution :

Thread:-

A Thread is concurrent unit of execution.

There are two types of Parallel Test Exicution:

1. Parallel Test Exicution Methods

2. Parallel Test Exicution Class

---------

1. Parallel Test Exicution Methods:

Program:

package TestNG\_Parallel\_Test\_Exicution;

import org.testng.annotations.Test;

public class ParallelTestExicutionMethods {

@Test

public void testCase1() {

long id=Thread.currentThread().getId();

System.out.println("Test case 1 is successful"+" Thread id :"+id);

}

@Test

public void testCase2() {

long id=Thread.currentThread().getId();

System.out.println("Test case 2 is successful"+" Thread id :"+id);

}

@Test

public void testCase3() {

long id=Thread.currentThread().getId();

System.out.println("Test case 3 is successful"+" Thread id :"+id);

}

@Test

public void testCase4() {

long id=Thread.currentThread().getId();

System.out.println("Test case 4 is successful"+" Thread id :"+id);

}

}

---------

XML file:

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">

<suite name="Suite" parallel="methods" thread-count="2">

<test name="Test">

<classes>

<class name="TestNG\_Parallel\_Test\_Exicution.ParallelTestExicutionMethods"/>

</classes>

</test> <!-- Test -->

</suite> <!-- Suite -->

-----------

Output:

[RemoteTestNG] detected TestNG version 7.4.0

Test case 2 is successful Thread id :15

Test case 1 is successful Thread id :14

Test case 4 is successful Thread id :14

Test case 3 is successful Thread id :15

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

Suite

Total tests run: 4, Passes: 4, Failures: 0, Skips: 0

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

2. Parallel Test Exicution Classes:

Here as example we are taking two classes

Class 1 :

package TestNG\_Parallel\_Test\_Exicution\_Classes;

import org.testng.annotations.Test;

public class ParalellTestExicutionClass1 {

@Test

public void testCase1() {

long id=Thread.currentThread().getId();

System.out.println("Test case 1 is successful"+" Thread id :"+id);

}

@Test

public void testCase2() {

long id=Thread.currentThread().getId();

System.out.println("Test case 2 is successful"+" Thread id :"+id);

}

@Test

public void testCase3() {

long id=Thread.currentThread().getId();

System.out.println("Test case 3 is successful"+" Thread id :"+id);

}

@Test

public void testCase4() {

long id=Thread.currentThread().getId();

System.out.println("Test case 4 is successful"+" Thread id :"+id);

}

}

-------

Class 2:

package TestNG\_Parallel\_Test\_Exicution\_Classes;

import org.testng.annotations.Test;

public class ParalellTestExicutionClass2 {

@Test

public void testCase5() {

long id=Thread.currentThread().getId();

System.out.println("Test case 5 is successful"+" Thread id :"+id);

}

@Test

public void testCase6() {

long id=Thread.currentThread().getId();

System.out.println("Test case 6 is successful"+" Thread id :"+id);

}

}

------------

XML file:

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<suite name=*"Suite"* parallel=*"classes"* thread-count=*"2"* >

<test name=*"Test"* >

<classes>

<class name=*"ParallelTestExicution.ParallelTestExicutionClass1"*/>

<class name=*"ParallelTestExicution.ParallelTestExicutionClass2"*/>

</classes>

</test> <!-- Test -->

</suite> <!-- Suite -->

------------

Output:

[RemoteTestNG] detected TestNG version 7.4.0

Test case 1 is successful Thread id :14

Test case 5 is successful Thread id :15

Test case 2 is successful Thread id :14

Test case 6 is successful Thread id :15

Test case 3 is successful Thread id :14

Test case 4 is successful Thread id :14

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

Suite

Total tests run: 6, Passes: 6, Failures: 0, Skips: 0

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

-------------------------------------------------------------------------------------------------------------------------

invocationCount in TestNG

Invocation count is used when you want to run the same tests multiple times. Below example illustrates how to use invocation count in TestNG. In below example, test1 will be executed 5 times.

**package** TestNG;

**import** org.testng.annotations.Test;

**public** **class** InvocationCountDemo {

@Test(invocationCount = 5)

**public** **void** test1(){

System.***out***.println("Invocation count demo");

}

}

enabled = false :

Sometimes, it happens that our code is not ready and the test case written to test that method/code fails. In such cases, annotation **@Test(enabled = false)** helps to disable this test case.

If a test method is annotated with *@Test(enabled = false)*, then the test case that is not ready to test is bypassed.

**package** TestNG;

**import** org.testng.annotations.Test;

**public** **class** EnabledequaltoFalseDemo {

@Test(enabled = **false**)

**public** **void** btest1() {

System.***out***.println("B.btest1");

}

}

timeOut=time in millisecond:

If a test class contains multiple test methods, if one of the test method is time consuming to execute

then TestNG by default fail that test method and execute other test methods which can be possible using timeOut.

Example:

**package** TestNG;

**import** org.testng.annotations.Test;

**public** **class** TimeOutDemo {

@Test

**public** **void** ContactVerify(){

System.***out***.println("Contact validation is successful");

}

@Test(timeOut = 1000)

**public** **void** LandingPage(){

System.***out***.println("Landing page verification is successful");

}

@Test

**public** **void** LoanContact(){

System.***out***.println("Loan contact details verification is successful");

}

}

Hard and Soft Assertions in Selenium

1)Hard Assert

2)Soft Assert

1)Hard Assert:

When assertion get fail then selenium stops the remaining execution of that same method.

Program:

**package** HardAssertSoftAssert;

**import** org.testng.Assert;

**import** org.testng.annotations.Test;

**public** **class** Test1 {

@Test

**public** **void** verifyPageTitle() {

String expected\_Title = "Google";

String actual\_Title = "Google1";

System.***out***.println("Test case execution started");

Assert.*assertEquals*(actual\_Title, expected\_Title);

System.***out***.println("Test case exicution finished");

}

}

Output:

[RemoteTestNG] detected TestNG version 7.4.0

Test case exicution started

FAILED: verifyPageTitle

java.lang.AssertionError: expected [Google] but found [Google1]

at org.testng.Assert.fail(Assert.java:99)

at org.testng.Assert.failNotEquals(Assert.java:1037)

at org.testng.Assert.assertEqualsImpl(Assert.java:140)

at org.testng.Assert.assertEquals(Assert.java:122)

at org.testng.Assert.assertEquals(Assert.java:629)

at org.testng.Assert.assertEquals(Assert.java:639)

at HardAssertSoftAssert.HardAssert.verifyPageTitle(HardAssert.java:13)

at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)

at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)

at java.lang.reflect.Method.invoke(Unknown Source)

at org.testng.internal.MethodInvocationHelper.invokeMethod(MethodInvocationHelper.java:133)

at org.testng.internal.TestInvoker.invokeMethod(TestInvoker.java:598)

at org.testng.internal.TestInvoker.invokeTestMethod(TestInvoker.java:173)

at org.testng.internal.MethodRunner.runInSequence(MethodRunner.java:46)

at org.testng.internal.TestInvoker$MethodInvocationAgent.invoke(TestInvoker.java:824)

at org.testng.internal.TestInvoker.invokeTestMethods(TestInvoker.java:146)

at org.testng.internal.TestMethodWorker.invokeTestMethods(TestMethodWorker.java:146)

at org.testng.internal.TestMethodWorker.run(TestMethodWorker.java:128)

at java.util.ArrayList.forEach(Unknown Source)

at org.testng.TestRunner.privateRun(TestRunner.java:794)

at org.testng.TestRunner.run(TestRunner.java:596)

at org.testng.SuiteRunner.runTest(SuiteRunner.java:377)

at org.testng.SuiteRunner.runSequentially(SuiteRunner.java:371)

at org.testng.SuiteRunner.privateRun(SuiteRunner.java:332)

at org.testng.SuiteRunner.run(SuiteRunner.java:276)

at org.testng.SuiteRunnerWorker.runSuite(SuiteRunnerWorker.java:53)

at org.testng.SuiteRunnerWorker.run(SuiteRunnerWorker.java:96)

at org.testng.TestNG.runSuitesSequentially(TestNG.java:1212)

at org.testng.TestNG.runSuitesLocally(TestNG.java:1134)

at org.testng.TestNG.runSuites(TestNG.java:1063)

at org.testng.TestNG.run(TestNG.java:1031)

at org.testng.remote.AbstractRemoteTestNG.run(AbstractRemoteTestNG.java:115)

at org.testng.remote.RemoteTestNG.initAndRun(RemoteTestNG.java:251)

at org.testng.remote.RemoteTestNG.main(RemoteTestNG.java:77)

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

Default test

Tests run: 1, Failures: 1, Skips: 0

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

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

Default suite

Total tests run: 1, Passes: 0, Failures: 1, Skips: 0

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

2.Soft Assert:

When assertion get fail then selenium not stop the remaining execution, remaining code line are executed.

Program:

**package** HardAssertSoftAssert;

**import** org.testng.Assert;

**import** org.testng.annotations.Test;

**import** org.testng.asserts.SoftAssert;

**public** **class** SoftAssert1 {

@Test

**public** **void** verifyPageTitle() {

String expected\_Title = "Google";

String actual\_Title = "Google1";

System.***out***.println("Test case exicution started");

SoftAssert softassert = **new** SoftAssert();

softassert.assertEquals(actual\_Title, expected\_Title);

String expected\_PageUrl = "google.com";

String actual\_PageUrl = "google.com1";

softassert.assertEquals(expected\_PageUrl, actual\_PageUrl);

System.***out***.println("Test case exicution finished");

}

}

-----------

OutPut:

[RemoteTestNG] detected TestNG version 7.4.0

Test case exicution started

Test case exicution finished

PASSED: verifyPageTitle

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

Default test

Tests run: 1, Failures: 0, Skips: 0

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

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

Default suite

Total tests run: 1, Passes: 1, Failures: 0, Skips: 0

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

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