TestNG

1. Testing.xml structure

<suite name= “”>

<test name=””>

<classes>

<class name=””/>

<class name=””/>

</classes>

</test>

</suite>

1. If you want to run multiple modules

<suite name= “”>

**<test name=”CarLoan”>**

<classes>

<class name=””/>

<class name=””/>

</classes>

</test>

**<test name=”Personal Loan”>**

<classes>

<class name=””/>

<class name=””/>

</classes>

</test>

</suite>

1. Exclude/include particular Testcase

Include – if you want to exclude 98 out of 100 testcases you cant write 98 times exclude to just use include tag to execute the remaining 2 test cases

**Note**: Follow particular nomenclature while writing the testcases ex: Mobile\_T1 etc so that if you want to exclude all mobile cases you can use

<exclude name=”mobile.\*”/>

<suite name= “”>

**<test name=”CarLoan”>**

<classes>

<class name=””/>

<class name=””/>

</classes>

</test>

**<test name=”Personal Loan”>**

<classes>

<class name=”test.day3”> //**packagename.classname**

<methods>

<exclude name=”mobilelogin”/> //method you want **to exclude** from running in **day3 class**

**<include name “”/>**

</methods>

</class>

<class name=””/>

</classes>

</test>

</suite>

1. How to run all testcases in package

<suite name=”1”>

<test name=”Loans”>

<packages>

<package name = “test”/>

</packages>

</test>

</suite>

Uses:

=====

To run multiple testcases at one go by just running testng.xml

You need to have @Test Annotation followed by method

You can define multiple tests from single class

You can moduralize the testcases based up on functionality and trigger them accordingly

Annotations

1.BeforeTest : Executes before @Test

i.e first @beforeTest method executes and next all @Test methods under <test> tag in testing.xml

executes

2. @AfterTest – Executes after all @Test methods under <test> tag in testing.xml

3. @Before Suite: Executes as first method and then all the methods(@Test) under multiple <test> tags executes

Ex: @before suite

<test>

@test

@test

@test

</test>

<test>

@test

@test

</test>

@aftersuite

[4.@Aftersuite-](mailto:4.@Aftersuite-) Executes after all the methods under multiple </test> tags

[5.@BeforeMethod-](mailto:5.@BeforeMethod-) Executes before every @Test in the class. If there are 3 tescases in a class then,before method executes before each and every method

[6.@AfterMethod-](mailto:6.@AfterMethod-) Executes after every @Test present in the class

[7.@BeforeClass-](mailto:7.@BeforeClass-) Executes once before all the cases in a class

8. @afterclass- Executes before all the cases in a class

@beforesuite

@beforeclass

@beforeTest

@beforeMethod

@test

@afterMethod

@beforeMethod

@test

@afterMethod

@beforeMethod

@test

@afterMethod

@Aftertest

@Afterclass

@AfterSuite

Groups

9. 50 Cases in a project, customer wants to run only 4 cases for every regression which are in different classes. How to achieve this?

Note: If those cases are in 1 class you can use concept on <include>

Solution: we can group the regression cases as below

<suite name=””>

<test name=””>

<groups>

<run>

<include name=”smoke”/>

</run>

</groups>

</classes>

<class name=”1”/>

<class name=””/>

<class name=””/>

<class name=””/>

<class name=”100”/>

</classes>

</test>

</suite>

**@Test(groups={“smoke”})**

**10. Flow of annotations**

@BeforeSuit (launch Browser) .

@BeforeClass (by using I am creating one Insta account) 1, 2, . . . . . 10 times

@BeforeMethod ( In db I am adding badge for the profile) 1-1, 1-2, 2-1, 2-2 . . . . . . . . . . .

@Test 10 test cases - 10 accounts for test 1-1, 1-2, 2-1, 2-2 . . . . . . . . . . .

@AfterMethod (remove badge from database) 1-1,1-2, 2-1, 2-2 . . . . . . . . . . .

@AfterClass (i am delete the account) 1, . . . . 10 times

@AfterSuit (I am closing browser)

@BeforeSuite

@BeforeClass

@BeforeTest

@BeforeMethod

@Test

@AfterMethod

@BeforeMethod

@Test

@AfterMethod

@AfterTest

@AfterClass

@AfterSuite

11. If you wont set any priority what is the order of @Test execution

It follows the Testcase name in alphabetic order

12. how do you skip testcases of one particular class

Just remove the class name from <classes> tag in testng.xml

13. @Test(dependsOnMethods={“weblogin”,”mobile” })

14. You already know that there is a known bug ,its reported nd you don’t want to execute it. How to skip that particular testcase

@Test(enabled=”false”)

15. If you want to wait for certain amount of time during execution of a method before it throws exception.

@Test(timeOut=4000)

Note: if you use implicit wait for 4000 seconds it waits for each and every element and brings the performance down

16. How to define global variables/common variables in testng.xml instead of hardcoding the values

In Testng.xml: Suite level Parameteization

<suite name=””>

<parameter name= “URL” value= “qaclickacademy.com”/>

In ClassLevel:

@Parameters({“URL”})

@Test

Public void webloginpage(String urlname)

{

}

Test Level Parameterization:

It applicables to all tests under that classes

<test>

<parameter name=”” url=””>

<classes>

<class>

<methods>

<include>

</methods>

<class>

<class>

</classes>

17. Parameterizing with multiple data sets by running tests with multiple combinations

@Test(dataProvider="getData")

public void Userid(String username,String password,String id)

{

System.out.println("This block executes before each Test");

System.out.println(username);

System.out.println(password);

System.out.println(id);

}

@DataProvider

public Object[][] getData()

{

//i stands for number of times testcase should run

//j stands for no of parametrs it should send for one go

Object[][] data=new Object[3][3];

data[0][0]="1abcd";

data[0][1]="1xyz";

data[0][2]="1dsaf";

data[1][0]="2abcd";

data[1][1]="2xyz";

data[1][2]="2dsaf";

data[2][0]="3abcd";

data[2][1]="3xyz";

data[2][2]="3dsaf";

return data;

}

18. Listeners - If you want to take a screenshot/perform any action after testcase has pass/Failed you redirect them to Listeners methods.

Implement ITestListeners Interface and write code for the unimplemented methods like onTestFailure, Onsuccess

Give path in testng.xml that where your listeners code is

<suite name=””>

<Listeners>

<Listener class=”packagename.classname”/>

</Listeners>

<test name=””>

Listeners.Java:

@override

Public void onTestSuccess(ITestResult result)

{

S.o.p(“I successfully passed”);

}

19. How to capture the name of testcase which failed

@override

Public void onTestFailure(ITestResult result)

{

S.o.p(“I have failed” +**result.getName()**);

}

20. Parallel execution using Testng

<suite name= “” **parallel= “tests” thread-count= “2”**>

<test name=””>

We can write at test level as well

<suite name= “” >

<test name=”” **parallel= “classes” thread-count= “2”**>

21. Testng Reports

Refresh the project.

Test-output folder>index.html -> right click->properties->copy url and pate in the browser->you will see passed/failed cases indetail

22. Assertion

Assert.assertEquals(actual,expected,”custom message”)

Custom message helps in identifying the test case which failed with help of its content

Ex: Assert.assertEquals(actual,expected,”**Tilte is mismatched**”)

AssertTrue()

Assertion verifies the boolean value returned by a condition. If the boolean value is true, then assertion passes the test case, and if the boolean value is false, then assertion aborts the test case by an exception.

 Assert.assertTrue(driver.findElement(By.cssSelector("input[id\*='SeniorCitizenDiscount']")).isSelected()

Hard Assertions: Assert are hard assertions. If it fails error will be thrown and further lines of code will not be executed.

Soft Assetions: further lines will be executed and exception can be thrown at specified location. Here control will be in our hands.

SoftAssert softass = new SoftAssert();

softass.assertEquals(actual,expected,”**Tilte is mismatched**”)

if you want to throw the error if any testcase failed at end of all your cases execution you can use below

softass.assertAll();

23. How to run only failed cases In Testng

Go to test-output folder>>testng-failed.xnl>>right click run as testng suite

24. Cross browser Testing



