**TestNG**

**+Summary**

TDD framework – Test Driven Development

OpenSource/JAR files

Used by Dev for Unit Testing of Java apps

Used to test New Gen framework

Powerful tool to design tcs and HTML report

Annotations/Priorties/Sequence/Dependency (btwn TCs)/Grouping/Data Provider

<https://testng.org/doc/documentation-main.html#testng-xml>

**+Install & Setup TestNG**

<https://www.guru99.com/install-testng-in-eclipse.html>

URL: <https://beust.com/eclipse/>

Once done, restart > Eclipse > Project > RC > Properties > JavaBuildPath > Add Library > TestNG > Next > Finish

Along with all jar files, TestNG jar files should also be available > select parent folder (TestNG) and Apply

Under project > Library > you should be able to see TestNG

**+ Start**

Src > Package > class > Don’t select public static void main >

Package <PackageName>;

Public class <ClassName> {

}

**+TestNG Basics - basics of Annotations (Pre/Test/Post)**

@Annotation should always be accompanied with a method

1TC: Suite – Test - Class – Method – **Test** - Method – Class - Test – Suite

2TCs: Suite – Test - Class – Method – **Test1** - Method - Method – **Test2** - Method – Class - Test - Suite

The order of running TC can be jumbled unless you have defined **‘priority’**

Group groups all TCs into single group

TCs are independent of each other

Package <PackageName>;

<<imports>>

Public class <ClassName> {

**@BeforeSuite** >> Starts with @; import form TestNG

Public void setUp(){ >> note we don’t write main for TestNG. If you write main it becomes Java

Sel code….

}

**@BeforeTest** >> Starts with @; import form TestNG

Public void launchBrowser (){

Sel code….

}

**@BeforeClass** >> Starts with @; import form TestNG

Public void login(){

Sel code….

}

**@BeforeMethod** >> Starts with @; import form TestNG

Public void enterURL(){

Sel code….

}

**@Test** >> Starts with @; import form TestNG; this is one tc

Public void titleTest(){

Sel code….

}

**@Test** >> Starts with @; import form TestNG; this is one tc

Public void search(){

Sel code….

}

**@AfterMethod** >> Starts with @; import form TestNG

Public void logout(){

Sel code….

}

**@AfterClass** >> Starts with @; import form TestNG

Public void closeBrowser(){

Sel code….

}

**@AfterTest**>> Starts with @; import form TestNG

Public void deleteCookies(){

Sel code….

}

**@AfterSuite**>> Starts with @; import form TestNG

Public void generateReport(){

Sel code….

}

}

To run: RC > RunAs > TestNG test

Results:

At the end, new tab will be created with Results.

Project > RC > Refresh > test-o/p folder > index.html file > RC > Properties > copy location > paste in browser > within this, you can see TCs by ‘Group’

**+Write Selenium webDriver code with TestNG integration**

Real example

**Priority and Groups**

Package <PackageName>;

<<imports>>

Public class GoogleTest {

Webdriver driver;

**@BeforeMethod**

Public void setUp(){

System.setProperty(“webdriver.chrome.driver”, “/Jo/Chome/chromedriver”);

driver = new ChromeDriver();

driver.manage().window().maximise();

driver.manage().window().deleteAllCookies();

driver.get(“http://www.google.com”);

}

**@Test (priority=1, groups = “Login’)**

Public void titleTest(){

String title = driver.getTitle();

System.out.println(title);

Assert.assertEquals(title, “Google”); >> Validation point

}

**@Test (priority=2)**

Public void logoTest(){

Boolean b = driver.findElement(By.xpath(“*Googleiconxpath*”)).isDisplayed(); >> check if logo is displayed and return a Y/N vaue

Assert.assertTrue(b); >> Validation point

}

**@AfterMethod**

Public void tearDown(){

Driver.quit();

}

}

**DependsOnMethods**

**@Test**

Public void titleTest(){

String title = driver.getTitle();

System.out.println(title);

}

**@Test (DependsOnMethods = titleTest )**

Public void logoTest(){

Boolean b = driver.findElement(By.xpath(“*Googleiconxpath*”)).isDisplayed(); >> check if logo is displayed and return a Y/N vaue

}

**@Test (DependsOnMethods = titleTest )**

Public void xyzTest(){

Boolean b = driver.findElement(By.xpath(“*Googleiconxpath*”)).isDisplayed(); >> check if logo is displayed and return a Y/N vaue

}

* **If titleTest fails all other TCs will not be executed as DependsOnMethods**

**Invocation Count**

**@Test (invocationCount =10 )**

Public void titleTest(){

String title = driver.getTitle();

System.out.println(title);

}

* The test case will be executed 10 times

**Exception Time Out (not usually used)**

**@Test (invocationTimeOut = 2000)**

Public void titleTest(){

String title = driver.getTitle();

System.out.println(title);

}

* If the test case is stuck and not getting competed, infinite loop or something, we are saying **invocationTimeOut = 2000** i.e test case should get executed in 2s, if not terminate it

If you have created, 10 classes within a package, you cant open them 1 by 1 and execute

Instead, create src > other > file > filename = testng.xml >

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

<suite name="Test Suite" >

<test name="Google Test" >

<classes>

<class name="Package Name.Class Name1" />

<class name="Package Name.Class Name2" />

</classes>

</test>

</suite>

* **LeftPane > src > testng.xml > RC > RunAs > TestNG > Review Results**