1. **TestNG:**

**U can control test cases , how they have to run, execution, group**

1. **Install and configure:**

>Testng.org: you can add .jar’s to build path or install testNG plugin for eclipse

> installing Plugin- goto testNG site> eclipse>how to install> copy URL for testNG plugin for eclipse

> Go to eclipse>help>install new soft> paste url in work with>wait>tick testNG>next>next>finish

1. **How to run test without Java compiler**

> Create new java Project

1. Add new package xyz- for storing all java class file

2. Create new class file in xyz package> don’t tick on public static ……..

3. create one public method eg.

**@Test**

**Public void FirstTest()**

**{**

**System.out.println("I am first test case");**

**}**

**@Test**

**Public void SecondTest()**

**{**

**System.out.println("I am second test case");**

**}**

> The @Test adds testNG annotations, any method after @Test will be treated as test case by TestNG

> For above Demo is name for test case

> run as TesNG test o/p test will run

> u can create multiple test cases from single class

1. **Creating TestNG XML file**

>Right click on project >> TestNG>> convert to TestNG

>XML hierarchy :::: test suite>>>test folder>>>test case

**<suite name=” name of suite”>**

**<test name="name of test">**

**<classes>**

**<class name="test.name of class file created in package of java project">**

**</class>**

**</classes>**

**</test>**

**</suite>**

1. **Editing TestNG XML:**

> you can changes or model the test cases based on functions (viz- mobile brands/tv brands etc) and trigger as we want

> **how to Exclude particular test cases from one class:** As we know method means test case , so we need to exclude that particular method from running syntax in XML as follows

**<suite name=” name of suite”>**

**<test name="name of test">**

**<classes>**

**<class name="test.name of class file created in package of java project">**

**<methods>**

**<exclude name="name of method"></exclude>**

**</methods>**

**</class>**

**</classes>**

**</test>**

**</suite>**

> **how to Include only particular test cases from one class:** as above XML syntax we need only to **add <include name=”name of test case or method”></include>**

**Scenario:** if we have many brands mobiles / washing machine/ tv in single class file and we want only exclude the washing machines of all brands then add **washingmachine.\*** in exclude syntax , Note: in framework naming convention is most important as for washing machines of all brands method name must start with washingmachineXYZ/ABC etc

**>How to add package in TestNG XML:** The Syntax for adding package in TestNG XML as follows which allow to execute all classes and all the test cases/methods from each class when we rung testNG XML as Suite

**<suite name="Name of suite">**

**<test name="Name of test">**

**<packages>**

**<package name="name of package in which all the class files are created"></package>**

**</packages>**

**</test>**

**</suite>**

**>TestNG Annotations:**

**a.** @BeforeTest : This annotation is used for executing the particular test case before running any testcase/method from test in testNG XML file which includes the class which have beforetest testcase

**Note:** This annotation is used when there is need of prerequisite to execute eg. Server to start etc

**b.** @AfterTest : This annotation is used for executing the particular test case after all testcase/method from test in testNG XML file which includes the class which have afterTest testcase

**c.** @BeforeSuite & @AfterSuite: it is used to set global environment variables to framework i.e before executing testcases in whole framework if there is need to set env’s that will be take care by this annotations

**we can set this annotation in any testcase/method of any class in the whole suite**

**d.** @BeforeMethod & @AfterMethod: these annotations are used when we need to execute particular test case before starting every testcase(methods in class) in class and after execution of each test case in single class file , this annotation is for class file only, it don’t have any relation with XML

**e.** @BeforeClass & @AfterClass: these annotations are used when we need to execute particular test case before start executing the testcases(methods in class) in that class and after execution of last test case of that class file , this annotation is for class file only , it don’t have any relation with XML

**> group in XML:** this is used for executing particular set of test cases , it is like running the smoke test .

@Test(groups= {"smoke"}):we need to add the tag like this here “smoke” is just a name for representation we can take any name we want. We have to add this syntax for the test cases which we need to execute across the classes

**<suite name="name of the suite">**

**<test name="name of the test">**

**<groups>**

**<run>**

**<include name="name of the group"></include>**

**</run>**

**</groups>**

**<classes>**

**<class name="test.name of the class"/>**

**<class name="test. name of the class "/>**

**<class name="test. name of the class "/>**

**<class name="test. name of the class "/>**

**</classes>**

**</test>**

**</suite>**