TestNG FAQ’s

**Explain what is TestNG?**

TestNG is an automated open source testing framework.  It is based on JUnit framework but is not a JUnit extension.

**2) Mention what are the TestNG features?**

TestNG features include

* TestNG uses more OO (object-oriented) and Java features
* It supports testing integrated classes
* Different Annotations are supported
* Separate compile time test code from data info /run time configuration
* Run-time configuration is flexible
* Flexible plug-in API
* For further flexibility embeds BeanShell
* Multi-threaded testing support
* Supports parallel testing, load testing, partial failure, dependent test methods
* After compilation of test, a request can be made to TestNG to run all the “front-end” tests or “slow”, “fast”, “database” ,
* For the same test class TestNG support for multiple instances
* For logging no dependencies, default JDK functions for logging and run-time

**3) List out the advantages of TestNG over Junit?**

Advantages of JUnit over TestNG includes

* Compare to JUnit annotations, TestNG are easy to understand
* Unlike JUnit, TestNG does not require to declare @BeforeClass and @AfterClass
* Method name constraint is not there in TestNG
* TestNG allows you the grouping of test cases easily which is not possible in JUnit
* TestNG supports following three additional setup: @Before/AfterSuite, @Before/AfterTest and @Before/AfterGroup
* TestNG does not need to extend any class
* In TestNG, it is possible to run selenium test cases in parallel
* Based on group TestNG allows you to execute the test cases
* TestNG allows you to determine the dependent test cases; each test case is autonomous to other test case

**4) Explain what are the basic steps required in writing TestNG tests?**

The basic steps required in writing TestNG includes

* Write down the business logic of your test and insert TestNG annotations in your code
* In a build.xml or testing.xml, add the information about your test
* Run TestNG

**5) List out various ways in which TestNG can be invoked?**

TestNG can be invoked in different ways like

* Using Eclipse
* With ant
* From the command line
* Using IntelliJ’s IDEA

**6) Explain what is testing.xml file used for?**

File testing.xml captures your entire testing in XML.  This file makes it easy to define all your test suites and their parameters in one file, which you can verify in your code repository or e-mail to coworkers. It also makes easy to pull out subsets of your tests or split several runtime configurations.

**7) In TestNG how can you disable a test?**

To disable the test case you don’t want, you can use annotations @Test(enabled = false).

**8) Explain what is Time-Out test in TestNG?**

The Time-Out test in TestNG is nothing but time allotted to perform unit testing. If the unit test fails to finish in that specific time limit, TestNG will abandon further testing and mark it as a failed.

**9) Explain what is exception test?**

TestNG gives option for tracing the Exception handling of code.  You can test whether a code throws the expected results or not.  The expectedExceptions parameter is availed along with @Test annotation.

**10) Mention what does the “suite test” does in TestNG?**

“Suite Test” is done when you have to run few unit test together, “ Suite Test” bundle this unit test together.  XML file is used to run the suite test.

**11) Explain what is parametric testing?**

Parameterized testing allows developers to execute the same test over and over again using different values.  In two different ways TestNG allows you to pass parameters directly to your test methods.

* With testing.xml
* With Data Providers

**12) Explain how can you run the JUnit tests using TestNG?**

You can run the JUnit tests using TestNG by

* Placing JUnit library on the TestNG classpath, so it can locate and use JUnit classes
* Change your test runner from JUnit to TestNG in Ant and then run TestNG in “mixed mode” . This will bring all your test in the same

This approach also enables you to convert your existing JUnit test to TestNG.

**13) Explain what does @Test(invocationCount=?) and (threadPoolSize=?) indicates?**

* **@Test (threadPoolSize=?):** The threadPoolSize attributes tell TestNG to form a thread pool to run the test method through multiple threads. With threadpool, the running time of the test method reduces greatly.
* **@Test(invocationCount=?):** The invocationcount tells how many times TestNG should run this test method

**14) Mention different ways in which you can produce reports for TestNG results?**

There are two ways to produce a report with Test NG, they are

* **Listeners:** For a listener class to implement, the class has to implement the org.testng./TestListener interface. These classes are informed at runtime by TestNG when the test begins, finsishes, skips, passes or fails.
* **Reporters:** For a reporting class to implement, the class has to implement an org.testng/Reporter interface. When the whole suite run ends, these classes are called. When called, the object consisting the information of the whole test run is delivered to this class.

**15) Explain what is Group Test in TestNG?**

It is a new feature included in TestNG, it allows you to dispatch methods into proper portions and perform grouping of test methods.  With group test, you can not only declare methods that belong to groups, but you can also specify groups that contain other groups.  Groups are determined in your testing.xml file using the group test.

**16) Explain in what ways does TestNG allows you to specify dependencies?**

TestNG allows you to specify dependencies in two ways

* Using attributes dependsOnMethods in @Test annotations
* Using attributes dependsOnGroups in @Test annotations

**17) Explain what it means when you have to pass parameters using data-providers in TestNG?**

When you have to pass complex parameter or parameters that are to be created from Java, in such instances parameters can be passed using Dataproviders.  The annotation for data provider is @DataProvider.  This annotation has only single string attribute, if the name is not declared; the Data provider’s name automatically defaults to the method’s name.  A data provider yields back an array of objects.

**18) Explain how you can execute tests in TestNG?**

The tests in TestNG  are executed using TestNG class. For running tests in TestNG framework, class is the main entry point.  Users can make their own TestNG object and invoke it in many different ways like

* On an already existing testing.xml
* On a synthetic testing.xml created entirely from Java
* By directly setting the test classes

**Introduction Of TestNG - Unit Testing Framework For Selenium WebDriver**

**TestNG** is unit testing framework and it has been most popular in very sort time among java software developers and **selenium webdriver software automation** test engineers. I can say, **TestNG** has not only most of all the features of JUnit but also many more advanced features in terms of **selenium webdriver testing framework**. Basically,

TestNG's software unit testing framework development is inspired from combination of NUnit and JUnit. You can [**VIEW JUNIT WITH WEBDRIVER TUTORIALS AND EXAMPLES**](http://www.software-testing-tutorials-automation.com/search/label/Junit%20with%20webdriver) before going to learn TestNG framework. There are many new features in TestNG which makes it more powerful, stronger and easier to use. [**VIEW SIMILARITIES AND DIFFERENCES**](http://www.software-testing-tutorials-automation.com/2014/03/what-are-similaritiesdifference-between.html) between TestNG and JUnit.

**TestNG Major Features**

* TestNG has a built in multiple Before and After and other annotations like @BeforeSuite, @AfterSuite, @BeforeTest, @BeforeGroups, @DataProvider, @Parameters, @Test etc.. All the TestNG annotations are very easy to understand and implement. We will look each and every annotation with detailed description in my latter posts.
* We can configure dependent test methods in TestNG means TestTwo() is dependent to TestOne(). We can also configure that if earlier test method (TestOne()) fails during execution then dependent software test method(TestTwo()) has to be executed or not.
* There is not required to extend any class in TestNG.
* We can configure our test suite using test methods, classes, packages, groups, etc.. in single testng.xml file.
* Support of configuring test groups like backendtest-group, frontendtest-group etc.. and we can tell TestNG to execute only specific group/groups.
* TestNG is supported by many tools and plug-ins like Eclipse, Maven, IDEA, etc..
* TestNG support parallel testing, Parameterization (Passing parameters values through XML configuration), Data Driven Testing(executing same test method multiple times using different data) .
* TestNG has built in HTML report and XML report generation facility. It has also built in  logging facility.

[**VIEW MORE TESTNG TUTORIALS**](http://www.software-testing-tutorials-automation.com/search/label/TestNG%20Tutorials)  
  
All these are major features of TestNG software unit testing framework and some of them are available in JUnit too. I recommend you to use  TestNG framework with WebDriver test because there are many useful features available in it compared to JUnit. Read my latter posts to learn TestNG framework with its detailed features.

**What Are The Similarities/Difference Between Junit and TestNG Framework For WebDriver**

As you know, **JUnit and TestNG** are very popular unit testing frameworks for java software developers and we can use them in our **webdriver** test execution for any software web application. Till now, I have described all [**WEBDRIVER TUTORIALS**](http://www.software-testing-tutorials-automation.com/2014/01/learn-selenium-webdriver-online-free.html) with junit framework. Many peoples are asking me to describe the **similarities and difference between JUnit and TestNG Framework**. First of all let me give you few similarities of [**JUnit**](http://www.software-testing-tutorials-automation.com/search/label/Junit%20with%20webdriver) and [**TestNG**](http://www.software-testing-tutorials-automation.com/search/label/TestNG%20Tutorials) frameworks and then I will tell you difference between both of them.

(Note : Bellow given similarities and differences are based on JUnit 4 version.)

**Similarities Between JUnit and TestNG**

1. We can create test suite in JUnit and TestNG both frameworks.
2. Timeout Test Is possible very easily in both the frameworks.
3. We can ignore specific test case execution of software web application from suite in both the frameworks.
4. It is possible to create expected exception test for software web application in both the frameworks.
5. Annotations - Few annotations are similar in both frameworks suite like @Test, @BeforeClass, @AfterClass. JUnit's Annotations @Before and @After are similar to TestNG's @BeforeMethod and @AfterMethod annotations.

**Difference Between JUnit and TestNG**

1. In TestNG, Parameterized test configuration is very easy while It is very hard to configure Parameterized test in JUnit.
2. TestNG support group test but it is not supported in JUnit.
3. TestNG has a feature to configure dependency test. Dependency test configuration for software web application is not possible in JUnit.
4. TestNG support @BeforeTest, @AfterTest, @BeforeSuite, @AfterSuite, @BeforeGroups, @AfterGroups which are not supported in JUnit.
5. Test prioritization, Parallel testing is possible in TestNG. It is not supported by JUnit.