

---

Guru99 Provides [FREE ONLINE TUTORIAL](#) on Various courses like

[Java](#) | [MIS](#) | [MongoDB](#) | [BigData](#) | [Cassandra](#) | [Web Services](#)

---

[SQLite](#) | [JSP](#) | [Informatica](#) | [Accounting](#) | [SAP Training](#) | [Python](#)

---

[Excel](#) | [ASP Net](#) | [HBase](#) | [Testing](#) | [Selenium](#) | [CCNA](#) | [NodeJS](#)

---

[TensorFlow](#) | [Data Warehouse](#) | [R Programming](#) | [Live Projects](#) | [DevOps](#)

---

## Top 19 TestNG Interview Questions & Answers

Following are frequently asked questions in interviews for freshers as well as experienced TestNG professionals.

### 1) What is TestNG?

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

### 2) 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 selenium 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 runtime

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

Advantages of TestNG over JUnit includes

- Compare to JUnit annotations, TestNG are easy to understand
- Unlike JUnit, TestNG does not require to declare @Before Class and @After Class
- 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 the following three additional setup: @Before/After Suite, @Before/After Test and @Before/After Group
- TestNG does not need to extend any class
- In TestNG, it is possible to run selenium web driver 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 another test case

#### 4) 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

The logo for TestNG, with 'Test' in a white, bold, italicized font with a black outline, and 'NG' in a large, bold, red font with a black outline.

## 6) Explain the use of testng.xml file

File testing.xml captures your entire testing in XML. This file makes it easy to define all your test suite and their parameters in one file, which you can verify in your code repository or e-mail to co-workers. It also makes it 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) What is Time-Out test in TestNG?

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

## 9) Explain exception test

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

## 10) What does the "suite test" does in TestNG?

"Suite Test" is done when you have to run a few unit tests together, " Suite Test" bundle this unit test together. An XML file is used to run the suite test.

## 11) 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) How you can run the JUnit tests using TestNG?

You can run the JUnit tests using TestNG by

- Placing JUnit library on the TestNG classpath so that 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) 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 a thread pool, the running time of the test method reduces greatly.
- **@Test(invocationCount=?):** The invocation count tells how many times TestNG should run this test method

### 14) What are the 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.ITestListener` Interface. These classes are informed at runtime by TestNG when the test begins, finishes, 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 of the information of the entire test run is delivered to this class.

### 15) 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 test groups that contain other groups. Groups are determined in your `testing.xml` file using the `group test`.

### 16) What are the ways to allow 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) What does it mean 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 instance parameters can be passed using Data providers. The annotation for data provider is `@DataProvider`. This annotation has only a 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) Name various methods to execute tests in TestNG?

The tests in TestNG are executed using TestNG class. For running tests in TestNG testing framework, the class is the main entry point. Users can make their 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

### **19) Give sample TestNG code**

[crayon-5f2e674461d13368829279/]