

# By OnlineInterviewQuestions.com

## TestNG Framework Interview Questions

### Q1. What is TestNG?

**TestNG** is a testing framework **written in Java** to cover a wide array of test categories like **unit testing**, **functional testing**, **end-to-end test**, **integration test**, etc. It is inspired from JUnit and NUnit but has more powerful features such as annotations, flexible test configuration, support for data-driven testing, support for parameters, and more.

### Q2. List some features of TestNG Framework?

Some of the important features of TestNG framework are,

- Support for Annotation.
- Support for data-driven-testing and parameters.
- Support for multiple instances in the same test class.
- Has a good flexible execution model.
- Allows for distributed testing.
- Has default JDK functions for runtime and logging.
- It has support for concurrent testing.

### Q3. What is Testng full form?

The full form of TestNG is **Test Next Generation**.

### Q4. What is use of testng.xml file?

**TestNG.xml file** has configuration details that are used to define test suites and tests. After configuring test suites in this XML file, you can execute the test from the **command line** or **ant script**.

You can specify packages, XML listeners, XML parameters, include and exclude methods in this configuration file.

### Q5. How to create Group of Groups in TestNG?

**The Groups in TestNG** allows you to create sophisticated groupings of test methods. Here, you can declare methods other groups along with groups that contain other groups. It provides maximum flexibility in partitioning the test. These groups are created by specifying it in the **testng.xml file**. It can be found either in `<test>` or `<suite>` tag.

## Q6. [What is Soft and Hard Assert in TestNG?](#)

**The Soft assert** collects error during the testing process in TestNG. When an assert fails, the soft assert doesn't throw an exception and continues to do the next step with the assert statement.

**Hard assert**, by contrast, throws an **AssertExceptionimmediately** if an assert statement fails, then the test suite continues with the next **@Test**.

## Q7. [What are Parameterized tests in TestNG?](#)

**Parameterized testing** is an exciting feature with the **TestNG** which lets you run the same test over and over again with different values. The values to the tests are passed as parameters in two different ways such as with **testng.xml**, and **data providers**.

The first way uses the **testng.xml** configuration file to define simple parameters and reference them in the source files.

In a second way, you can use the **Data provider** annotation to pass complex parameters.

## Q8. [What is Time-Out test in TestNG?](#)

This is a feature that lets the test method to be terminated when it is taking a longer time than the specified time to execute. Some test gets stuck or may take a longer time than to complete than the execution and to handle these scenarios, we can specify timeout so it gets terminated as it reaches the specified time. This **Timeout** can be specified in two ways. They are at the **suite level** and at **each test method level**. Specifying at suite level applies for all the tests in the test suite. Specifying at the method level is applicable only for that method.

## Q9. [How to disable a test case in TestNG?](#)

**To disable a test in TestNG**, you can set the enabled attribute in the **@Test annotation to false**. It will disable the test method from execution from the test suite. To disable all the public methods inside the class, you can set this attribute at the class level.

```
//set the following attribute to false to disable test case in TestNG.  
@Test(enabled=false)
```

## Q10. Enlist different annotations available in TestNG?

Some of the different annotations available in TestNG are,

- **@BeforeSuite** - It will run before the execution of the test method in the suite.
- **@AfterSuite** - It will run after the execution of the test method in the suite.
- **@BeforeTest** - It will run before the execution of the test methods of available classes belonging to the folder.
- **@AfterTest** - It will run after the execution of the test methods of available classes belonging to the folder.
- **@BeforeClass** - it will run before the first method of the current class is invoked.
- **@AfterClass** - it will run after the first method of the current class is invoked.
- **@BeforeMethod** - it will be executed before the execution of each test method.
- **@AfterMethod** - it will be executed after the execution of each test method.
- **@BeforeGroups** - It will run once before the execution of all test cases belonging to the group.
- **@AfterGroups** - It will run once after the execution of all test cases belonging to the group.

## Q11. List attributes of parallel tests in TestNG?

**TestNG** allows for tests to run in parallel or in multi-threaded mode. Here, different threads are started simultaneously and test methods are executed in them. The parallel attribute on the '**test**' tag will run all the **@Test methods** in the tag in the same thread and each tag will run in separate thread parallelly. Not only in the '**test**' tag, but parallel attribute can also be set in **classes, methods, instances** to run it parallelly.

## Q12. What are Listeners in TestNG?

**Listeners** are like interfaces in **TestNG** that are used to modify the behavior of default TestNG. Listeners listen to the event defined in the script and behave accordingly to the event. There are many TestNG listeners available such as **IAnnotationTransformer, IAnnotationTransformer2, IConfigurable, IHookable, ITestListener** and more.

Please Visit [OnlineInterviewquestions.com](http://OnlineInterviewquestions.com) to download more pdfs