**What is TestNG?**

**TestNG** is an automation testing framework in which NG stands for “Next Generation”. TestNG is inspired by JUnit which uses the annotations (@). TestNG overcomes the disadvantages of JUnit and is designed to make end-to-end testing easy.

Using TestNG, you can generate a proper report, and you can easily come to know how many test cases are passed, failed, and skipped. You can execute the failed test cases separately.

For example:

• Suppose, you have five test cases, one method is written for each test case (Assume that the program is written using the main method without using testNG). When you run this program first, three methods are executed successfully, and the fourth method is failed. Then correct the errors present in the fourth method, now you want to run only fourth method because first three methods are anyway executed successfully. This is not possible without using TestNG.

• The TestNG in Selenium provides an option, i.e., testng-failed.xml file in test-output folder. If you want to run only failed test cases means you run this XML file. It will execute only failed test cases.

Advantages:

• Generate the report in a proper format including a number of test cases runs, the number of test cases passed, the number of test cases failed, and the number of test cases skipped.

• Multiple test cases can be grouped more easily by converting them into testng.xml file. In which you can make priorities which test case should be executed first.

• The same test case can be executed multiple times without loops just by using keyword called ‘invocation count.’

• The TestNG framework can be easily integrated with tools like TestNG Maven, Jenkins, etc.

• Annotations used in the testing are very easy to understand ex: @BeforeMethod, @AfterMethod, @BeforeTest, @AfterTest

• WebDriver has no native mechanism for generating reports. TestNG can generate the report in a readable format like the one shown below.

• TestNG simplifies the way the tests are coded. There is no more need for a static main method in our tests. The sequence of actions is regulated by easy-to-understand annotations that do not require methods to be static.

**Advantages of TestNG over JUnit**

There are three major advantages of TestNG over JUnit:

• Annotations are easier to understand

• Test cases can be grouped more easily

• Parallel testing is possible

What are TestNG Annotations?

TestNG Annotations are used to control the next method to be executed in the test script. TestNG annotations are defined before every method in the test code. In case any method is not prefixed with annotations, it will be ignored and not be executed as part of the test code. To define them, methods need to be simply annotated with ‘@Test‘.

Types of TestNG Annotations

Below is the list of annotations that TestNG support in Selenium

• BeforeSuite

• BeforeTest

• BeforeClass

• BeforeMethod

• Test

• AfterMethod

• AfterClass

• AfterTest

• AfterSuite

Let’s explore how these methods work.

• @**BeforeMethod**: This will be executed before every @test annotated method.

• @**AfterMethod**: This will be executed after every @test annotated method.

• @**BeforeClass**: This will be executed before first @Test method execution. It will be executed one only time throughout the test case.

• @**AfterClass**: This will be executed after all test methods in the current class have been run

• @**BeforeTest**: This will be executed before the first @Test annotated method.

• @**AfterTest**: A method with this annotation will be executed when all @Test annotated methods complete the execution of those classes inside the <test> tag in the TestNG.xml file.

• @**BeforeSuite**: It will run only once, before all tests in the suite are executed.

• @**AfterSuite**: A method with this annotation will run once after the execution of all tests in the suite is complete.