**TestNG**

Selenium does not have any built-in hardware or framework for generating test reports, you require the help of an external framework like TestNG to fulfill the need for generating test reports and simplifying testing requirements such as functional testing, regression, end-to-end testing, and more

**TestNG in Selenium: An Overview**

In Selenium, there are mainly two testing frameworks available and they are:

* JUnit
* TestNG

**What is TestNG in Selenium?**

* TestNG is an open-source testing framework where NG stands for ‘Next Generation.’
* It is architected to simplify a broad range of testing needs starting from unit testing to integrated system testing. Initially, both JUnit and TestNG were designed solely for unit testing.
* TestNG is inspired by the JUnit Java platform and NUnit .NET platform, and some new functionalities were introduced in TestNG, making it more powerful and easy to use than the JUnit testing framework.
* NUnit testing framework for Selenium with , since NUnit supports the .NET platform.

**Advantages of TestNG over JUnit**

* TestNG Annotations are used to create test cases easily.
* 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.
* Test cases can be ‘grouped,’ ‘prioritized,’ and ‘executed’ more efficiently.
* It supports parameterization.
* It supports data-driven testing using Data Providers.
* It can generate HTML test reports of the results representing: the number of test cases runs, the number of test cases failed, the number of test cases skipped
* It effortlessly supports integration with various other tools and plugins like Eclipse IDE and built automation tools like Ant and Maven.
* It supports parallel execution.
* Logs can be generated.
* In TestNG, there is no need to state @AfterClass and @BeforeClass in a project, which is present in JUnit.
* You can specify any test method name in TestNG as the method’s name constraint is not present in TestNG like it is in JUnit.
* TestNG supports the following three additional setup and teardown levels: @Before/AfterSuite, @Before/AfterTest, and @Before/AfterGroup. TestNG goes beyond the idea of just writing @Test annotated methods and allows you to define these methods that will be run after or before your test suites, test groups, or test methods. This is very useful for your Selenium tests because you can create a Selenium server and browser instance before you start running your test suite.
* TestNG in Selenium does not require extending any class; hence, no need for the inheritance functionality.
* TestNG allows us to define the dependent test cases.
* TestNG in Selenium allows us to execute test cases based on the group. Let’s take a scenario where you have created two sets of groups ‘Regression’ and ‘Sanity.’ If you want to execute the test cases under the Sanity group, then you can do so easily with the TestNG framework.