

Test Case Priority in TestNG

TestNG is a [Testing](#) framework, that covers different types of test designs like a unit test, functional test, end to end test, UI test and integration test.

You can run a single or multiple test cases in your TestNG code.

If test priority is not defined while, running multiple test cases, TestNG assigns all @Test a priority as zero(0).

Now, while running; lower priorities will be scheduled first.

Demo of TestNG code without Priority

Let's take a scenario where sequencing will be required in order to pass all test cases:



Scenario: Generate a code where you are required to perform a Google search with a specific keyword say "Facebook". Now, verify that Browser title is changed to "Facebook - Google Search".

Note: Each step which you code should be in separate methods

Method 1: Open Browser say Firefox (openBrowser())

Method 2: Launch Google.com (launchGoogle())

Method 3: Perform a search using "Facebook" (performSearchAndClick1stLink())

Method 4: Verify Google search page title (FaceBookPageTitleVerification())

Code for our scenario:

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.testng.Assert;
import org.testng.annotations.Test;

public class Priority_In_testNG {
    WebDriver driver;

    // Method 1: Open Brower say Firefox
    @Test
    public void openBrowser() {
        driver = new FirefoxDriver();
    }

    // Method 2: Launch Google.com
    @Test
    public void launchGoogle() {
        driver.get("http://www.google.co.in");
    }

    // Method 3: Perform a search using "Facebook"
    @Test
    public void peformSeachAndClick1stLink() {
        driver.findElement(By.xpath(".*[@title='Search']")).sendKeys("Facebook");
    }

    // Method 4: Verify Google search page title.
    @Test
    public void FaceBookPageTitleVerification() throws Exception {
        driver.findElement(By.xpath(".*[@value='Search']")).click();

        Thread.sleep(3000);
        Assert.assertEquals(driver.getTitle().contains("Facebook - Google Search"), true);
    }
}
```

```
}  
}
```

Explanation of Code

As mentioned above we have created 4 test cases for performing each action in an independent methods.

- The first method (**openBrowser**) states to initialize Firefox browser.
- The second method (**launchGoogle**) states that launch Google.com is in the initialized browser.
- The third method (**performSearchAndClick1stLink**) states that perform a search in the search box (with xpath ("**//*[@title='Search']**") with a search term as **Facebook** and
- The fourth and last method (**FacebookPageTitleVerification**) states that click on search icon of Google and verify that browser title has been changed to **Facebook - Google Search**.

Now run this code using testNG as shown in the video you will find all the test case are failing. The reason for failure: as there is a dependency of previous test case to pass, only than current running test case will be passed.

In this case,

- First method which is executed is **openBrowser()**. It got passed because it does not have any dependency.
- Second method executed is **FacebookPageTitleVerification()**; it is failing because we are trying to click search button and verifying browser title.
- You can see that if search activity is not process then how any other step can get passed. Hence, this is the reason my test cases are failing.

PASSED: openBrowser

FAILED: FacebookPageTitleVerification

FAILED: launchGoogle

FAILED: performSearchAndClick1stLink

TestNG Without Priority



Importance of Priority in running testNG methods

As you have seen in the previous example that sequencing required in order to pass this scenario, so we will be modifying the previous piece of code with **Priority Parameter** so that each test should run against to the priority assigned to them.

Now as you can see we have assigned the Priority to each test case means test case will the lower priority value will be executed first.

Priority in testNG in action

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.testng.Assert;
import org.testng.annotations.Test;
```

```

public class Priority_In_testNG {
    WebDriver driver;

    // Method 1: Open Browser say Firefox
    @Test (priority=1)
    public void openBrowser() {
        driver = new FirefoxDriver();
    }

    // Method 2: Launch Google.com
    @Test (priority=2)
    public void launchGoogle() {
        driver.get("http://www.google.co.in");
    }

    // Method 3: Perform a search using "Facebook"
    @Test (priority=3)
    public void performSearchAndClick1stLink() {
        driver.findElement(By.xpath("//*[@title='Search']")).sendKeys("Facebook");
    }

    // Method 4: Verify Google search page title.
    @Test (priority=4)
    public void FacebookPageTitleVerification() throws Exception {
        driver.findElement(By.xpath("//*[@value='Search']")).click();
        Thread.sleep(3000);
        Assert.assertEquals(driver.getTitle().contains("Facebook - Google Search"), true);
    }
}

```

Explanation of Code

After assigning priority to each testcases, run the above code using testNG as shown in Video-2 mentioned below.

Here, you can see that test cases are prioritized. Test case having lower priority are executed first i.e. now there is a sequential execution according to priority in the test cases. Hence, all test cases are passing now.

Note the console of eclipse:

Output :

PASSED: openBrowser

PASSED: launchGoogle

PASSED: peformSearchAndClick1stLink

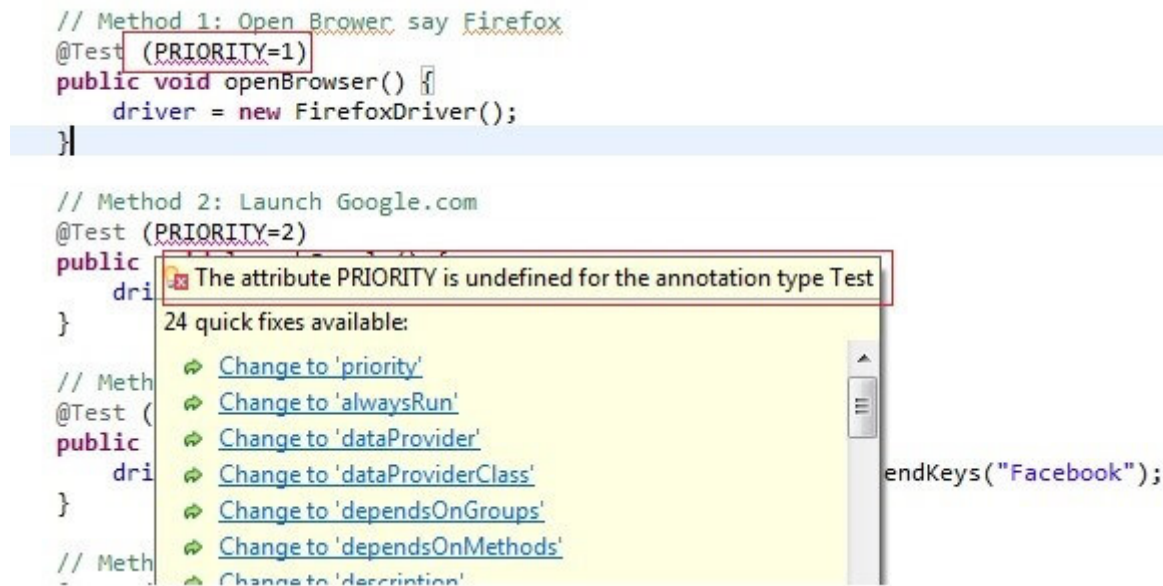
PASSED: FaceBookPageTitleVerification

TestNG With Priority



Case-sensitive in TestNG

Just for your information there is a standard syntax for defining priority in testNG i.e. **@Test (priority=4)**, suppose you are defining it in some other syntax say **@Test (PRIORITY=1)** then your IDE will show it as a compilation error. Refer image below:



Conclusion:

As you have seen that if there is a requirement to run a set of test-case in specific sequence then it can be easily done using **Priority** using testNG as a run tool.

This article is contributed by Ramandeep Singh, who is a test automation engineer at a leading MNC.