Chapter 4:
Priority Attribute

Introduction

Hello and Welcome To Chapter 4, Priority Attribute. In this chapter, we will discuss the Default Execution Order For Test Methods and the Priority Attribute For The Test Methods.

Default Execution Order For Test Methods

The Default Execution Order For Test Methods. A Test Method is marked by the @Test Annotation. We can set our methods as a Test Method and we can set our class as a test.

Class Marked By Test Annotation

Class Marked By Test Annotation. Let's go to Eclipse. In this example, I marked the class as part of the TestNG Test by adding a Test annotation at the class level. setup and teardown have Configuration Annotations that uses the BeforeClass and AfterClass annotations. signIn does not have an annotation. Also, searchTShirt and signOut do not have an annotation. Looking at this layout, it seems like the execution order should be setup, signIn, searchTShirt, signOut, then teardown. However, that's not the case, execution is going to run searchTShirt before signIn and signOut.

Why will it execute searchTShirt before signIn and signOut? It will execute searchTShirt first because the order depends on the names of each method. TestNG runs the program in ascending alphabetical order from A – Z. Therefore, in ascending order, se in searchTShirt comes before si in signIn and signOut. We can place the Test Methods anywhere on the editor and it will run in the same order every time. It will run in the same order because the Test annotations identify the Test Methods.

Before running this program, let's walkthrough the Test Application. The plan is to first Sign In, by entering an Email of TestNG@Framework.com and Password of TestNG1234 then click the Sign button. After clicking the Sign In button, we click T-Shirts, search for a Blue T-shirt, then click the Search button. Last, we Sign Out.

Now, let's run our Test Script. We see the order shows Number 2 Search For T-Shirt, Number 1 Sign In, and Number 3 Sign Out. The same order shows up in the Results tab. As a side note, only the methods with a public access modifier are marked as Test Methods when marking the class with a Test Annotation. Any other access modifier will not set the method as a Test Method. If I change one of these methods to private then that method will not show up in the Console or Results tab. Know what, I'm going to change 2 methods. Sign Out will have a default access modifier. Let's run. Only Step 2 Search For T-Shirt shows up. Change both methods back to public.

Methods Marked By Test Annotation

Methods Marked By Test Annotation. It’s the same when adding a Test annotation at the method level. In this example, the class is not marked as a test but all methods except for setUp and teardown are marked as Test Methods. Sign In has a Test annotation, Search T Shirt has a Test annotation, and Sign Out has a Test annotation. I’m going to run and we will see the same execution order. 2, 1, 3: Search For T-Shirt, Sign In, and Sign Out

Priority Attributes For Test Methods

Priority Attribute For Test Methods. The purpose of a priority attribute is to determine the execution order for our Test Method. The Test annotation has a lot of attributes. We can see those attributes by going to the TestNG Library, TestNG jar file, org.testng, annotations, scroll down to the Test.class then we see the attributes. Priority uses an integer Data Type.

We use a priority attribute by writing priority within a parenthesis after the Test annotation. The lowest number gets executed first. Some people start at zero but I prefer to use 1 since it’s the first Test Method. The next Test Method is Search T Shirt which will be priority equal 2. The last Test Method is Sign Out and that will have a priority of 3. Let’s run.

Now, we see the correct order. Step 1 Sign, Step 2 Search For T Shirt and Step 3 Sign Out. The Priority attribute helps us order our Test Methods. Next in Chapter 5, we will cover TestNG Assertions which verify if our test Pass or Fail.