Rex Jones II

Chapter 9:
Cross Browser Testing

Introduction

In this chapter, we will discuss the Parameter tag for the TestNG xml file, the Parameters Annotation, and the different ways to supply Test Data. Cross Browser Testing is a form of Data Driven Testing because we can drive different data sets using the parameter tag and Parameters Annotation. I'm going to send different browser names from the xml file to the Parameters Annotation.

Parameter Tag via xml File

The parameter tag specifies the name and value of a parameter. In this xml file, we have the same class name "Test Automation U" located in 3 different Test: One test is for Internet Explorer, one test is for Firefox, and the other test is for Chrome.

We have the option of placing the parameter tag within the suite level or test level. The parameter tag will get overridden at the test level if we add the tag in both places with the same parameter name. This xml file has different parameter names. We see URL and BrowserType. Why will it get overridden at the test level if the parameter name was the same? It's overridden at the test level because the test level is the closest to the class level which uses the parameter name. I decide the level to place the parameter name and value by determining if all classes need the same value. If all classes need the same value then I add the parameter tag at the suite level. If the classes require a unique value then I add the parameter tag at the test level.

All 3 classes need to access the same website. Therefore, at the suite level, we have parameter name = "URL" in parenthesis then the value of the URL is https://testautomationu.applitools.com.

Next, is the test level. The parameter name will be the same for each test but the values are going to be different. Let's start with Test On IE. The parameter name is BrowserType and the value is InternetExplorer. This test will use Internet Explorer as a browser.

Test on Firefox is next. It uses Firefox as the browser. The last test is Chrome which has a value of Chrome.

Parameters Annotation

The purpose of a Parameters Annotation is to point out how to pass parameters and which parameters to pass to the Test Method.

We are going to pass the URL and BrowserType from xml to the Parameters Annotation. Import the annotation and description states "Describes how to pass parameters to a Test Method". The value we

write in this annotation must match the parameter name from the xml file. The parameter names are URL and BrowserType.

Parameters in the annotation are different from parameters in the Test Method. Parameters in the annotation are a list of parameter names to be looked up in the xml file. Parameters in the Test Method receives the values from the xml file. Therefore, one gets the parameter name and the other one gets the parameter value.

Let's walkthrough this code before we execute. If the Browser Type is Internet Explorer then IE gets setup and open. After it is open then the window maximizes and load the url. The url https://testautomationu.applitools.com is sent from the xml file. The rest are print statements to show which browser is open, print the title, get the motto, and show which browser is closed. The same process repeats for each test in the xml file.

We must run the xml file. A failure will happen if we execute from the class. We saw all 3 browsers load with no Failures. First, we see Open Internet Explorer

Test Automation University | Applitools

Your path to become a QA Automation superstar!?

Close Internet Explorer

Next, we see Open Firefox and the print statements for Firefox. Next, we see Chrome and the print statements with Chrome. The Results tab shows all 3 tests passed with their data set.

In this chapter, we hard-coded our Test Data sets.

Different Ways To Supply Test Data

However, we can use different ways to supply our Test Data. Those ways are CSV File, Database, Properties File, Microsoft Excel, and Hard Code our Data. All of the ways have their Pros and Cons. Next in the final chapter, I will provide a brief overview of additional TestNG Concepts that were not covered in our course.

Additional Concepts

Here's some additional TestNG concepts that were not covered in this course:

- The ability to disable a Test Method by setting enable to false
- Execute a package at runtime
- Executing TestNG from the Command Prompt
- Provide optional values using the Optional Annotation
- Add Listeners which is a registration for Test Results
- Add Logs
- And the ability to view default reports

- Create custom reports
- Last but not least, multithreading which executes multiple components of a program at the same time and there's more TestNG concepts available to us

Thanks

I want to end by saying Thanks to Angie Jones. She's good, I like her and Thanks to Applitools. They gave me an opportunity to be your instructor for Test Next Generation - TestNG (A Powerful Test Framework).

You can reach me at Rex.Jones@Test4Success.org. I have a social network that provide videos on Selenium, Java, and TestNG. The videos are available on YouTube, LinkedIn, and Facebook.

I am also the author of 6 books that covers programming and automation. Some of the books are getting updated to include videos.

Thank You for watching this course and I Wish You Much Success.