TEST AUTOMATION UNIVERSITY
Rex Jones II

Chapter 8:
Data Driven Testing

Introduction

In this chapter, we will discuss the DataProvider Annotation, the dataProvider Attribute, and the dataProviderClass Attribute. Let's take a look at the programming code before discussing the Annotation and both Attributes.

Line 11 shows a logIn method with 3 parameters: email, password, and success. They will receive their argument values from the logInData method. The values are placed in a 2-Dimensional Array. There are 3 rows of Test Data. Row 1, Row 2, and Row 3. The 1st column of data will pass email values to String email. The 2nd column of data will pass password values to String password. The 3rd column of data will pass success values to boolean success. Statement return data help us return values to the logIn method.

After the logIn method receives the argument values then the parameters use those values. Those values are printed in the following print statement. We see email, password, and success. That's a brief overview of this Java code for parameters, arguments, and 2-Dimensional Array.

DataProvider Annotation

Now, let's discuss the DataProvider Annotation. The DataProvider Annotation returns Java objects which are values to the Test Method. We implement a DataProvider Annotation by writing @DataProvider above the method that has the Test Data. In this case, logInData has the Test Data. Next, we import the annotation. The description states "Marks a method as supplying data for a test method". The logInData method will supply data to the logIn Test Method.

The DataProvider Annotation presents 2 purposes at the same time. The 1st purpose is to pass an unlimited number of values to a Test Method. There is no restriction. The values can be any Java Data Type. In our example, we use a String and boolean Data Type. The 2nd purpose is to allow the Test Method to be invoked with different data sets. Each set will run and have its own Test Results. In our example, we have 3 sets of data.

dataProvider Attribute

The dataProvider Attribute. The dataProvider Attribute connects the DataProvider Annotation to the Test Method. This how both methods converse with each other. Since it's an attribute of the Test Annotation, we write dataProvider with a lowercase d equals the DataProvider's method name logInData surrounded by double quotes. Let's run.

All 3 login Data sets Passed. Here's the print statements that show Email, Password, and Successful Log In as true or false. The Results tab shows all 3 data sets Passed.

We can also give the DataProvider Annotation a name. In this example, the name is login-provider. We connect the DataProvider Annotation and the Test Method by using the DataProvider name and not the method name. The attribute shows login-provider. We see the DataProvider's name is optional because it executed successfully the last time without a name. Either way is okay.

Let's quickly walk through this Selenium Code for entering each data set for email and password. Line 24 enters the email for all 3 data sets. First is TestNG@Framework.com then Joe@Doe.com, and last is Test@AutomationU.com. Joe@Doe.com is the only one that is not valid.

Line 25 enters the password. Just like with Email, a password is entered for all 3 data sets. Let's Run. Authentication Failed for Joe@Doe.com. The console shows 1 Failure. We see the print statements. The Results tab also shows 2 Passed and 1 Failure. AssertionError - The Actual & Expected Results Do Not Match expected Sign Out but found Contact Us.

dataProviderClass Attribute

The dataProviderClass Attribute allows us to separate the Test Method and DataProvider into different classes. Up to this point, our Test Method and DataProvider were located in the same class. The dataProviderClass is an attribute of the Test Annotation just like the dataProvider. Here's the DataProvider Annotation.

We see OrangeHRM and SignInDP are 2 different classes. SignInDP has the DataProvider Annotation and OrangeHRM has the Test Method. To connect these 2 classes, we make the DataProvider class static then write dataProviderClass = SignInDP.class in the Test annotation. There are 4 different data sets in the DataProvider Annotation. 2 valid data sets and 2 invalid data sets.

email and password. My bad, this should be username. ALT + SHIFT + R will change the name in all places. username and password receive their data sets at Lines 22 and 23.

Let's Run. We see 2 Failures. Not Valid / NotValid34, false and Invalid / Invalid123, false are the 2 failures. Here's the 2 Passed admin / admin123. The difference is one A is capitalized and the other a is lowercase. The Results tab shows the first and last data sets Failed. We can gather our Test Data in one place and reuse it with many Test Methods. Next is Cross Browser Testing which executes one Test Script on more than one browser at the same time.