

Chapter 6.2: Hard Asserts

We ran our Test Script and it showed Passed without a verification step. The Test Script showed Pass although there is no code in this userSearch Test Method. I'm going to add the code back.

Let's pretend our Test Method requires us to verify the Home Page for OrangeHRM. Go to OrangeHRM and walkthrough the steps. Sign In – Admin / admin123 then click the Login Button. Here's the Home Page. We are going to verify the Welcome hyperlink, Admin tab, and Dashboard. Go back to Eclipse and change the step numbers for userSearch, userSignOut, and teardown. Search User will be 6, User Sign Out will be 7, and Number 8 will be teardown Close Chrome and Application.

Let's add a new Test Method between signIn and userSearch then add some assertions. @Test public void testHomePageVerification. Assert is located in the TestNG class so we write Assert dot and we see the same assertions from last chapter. Our assertion will be assertEquals.

There're so many methods but they are overloaded. Select an assertion with a String message. We see actual, expected, and message. Actual is true, Expected is true. This assertion will pass because both parameters are equal. The message only shows up if there is a failure so let's write the message like there is an assertion failure. The Welcome Link Is Not Correct On The Home Page. Print statement 3 Verify Welcome Link.

Verify Admin tab. Assert.assertFalse. The description shows Asserts that a condition is false. If it is not an AssertionError, with the given message, is thrown. Pass this assertion by writing false with a message that states The Admin Tab Is Not Displayed On The Home Page. In order for this assertion to pass the condition must return false. sysout Verify Admin Tab

Verify Dashboard. Assert.assertTrue. Pass this assertion by writing true with a message that states The Dashboard Is Not Correct On The Home Page. sysout 5 Verify Dashboard. This test will Pass. Let's Run. We see all 4 Test Methods Passed. All of the Steps 1 – 8 are printed on the Console including Number 3 Verify Welcome Link, Number 4 Verify Admin Tab, and Number 5 Verify Dashboard. The Results tab also shows each Test Method Passed.

Now, let's see what happens when this Test Method Fails. I'm going to Fail assertEquals for the Welcome Link and assertTrue for the Dashboard. Make the expected result false for assertEquals and the condition false for assertTrue then Run. As expected, testHomePageVerification FAILED.

Let's look at the steps. Steps 3, 4, and 5 are missing. The AssertionError shows our message because our Test failed. The Welcome Link Is Not Correct On The Home Page expected false but found true. Scroll to the bottom and we see no more Assertions. Why is there only 1 AssertionError when 2 assertions failed? The Results tab also shows a Failure with 1 AssertionError.

There is only 1 AssertionError because we used a Hard Assert. A Hard Assert stops immediately after a Failure then move on to the next annotation. In this case, assertEquals statement failed and skipped the subsequent assertions: assertFalse and assertTrue then moved on to the Test Method userSearch which has an @Test annotation. That's not good and that's why we did not see a Print Statement for Steps 3, 4, and 5. For this reason, TestNG introduced Soft Asserts.

Soft Alerts are located in the asserts package and there's only 2 methods: assertAll and doAssert. However, the Soft Assert class can extend the Assertion class so it can use all of these methods.

Next, we will cover Soft Asserts.