Testing using Selenium and JUnit4 in Java

Overview

In this tutorial, we will create a JUnit4 Test Case in Java using Selenium for browser automation.

> **Please note**
>
> There is a more up-to-date tutorial using JUnit5 providing additional capabilities. Please check it instead.

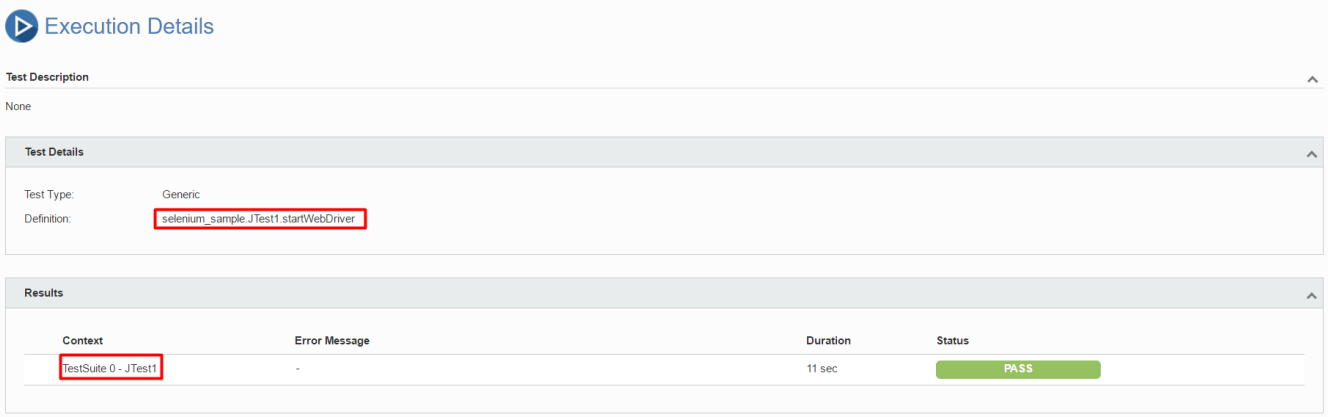
Description

```
package selenium_sample;

import org.junit.Assert;
import org.junit.Test;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
public class JTest1 {
    @Test
    public void startWebDriver(){

        //WebDriver driver = new FirefoxDriver();
        WebDriver driver = new ChromeDriver();
        driver.navigate().to("http://seleniumsimplified.com");
        Assert.assertTrue("title should start differently",
                          driver.getTitle().startsWith("Selenium Simplified"));
        driver.close();
        driver.quit();
    }
}
```

After successfully running the Test Case and generating the JUnit XML report (e.g., TESTS-TestSuites.xml), it can be imported to Xray (either by the REST API or through **Import Execution Results** action within the Test Execution).

JUnit's Test Case is mapped to a Generic Test in Jira, and the **Generic Test Definition** field contains the name of the class, and the method name that implements the Test Case.

The Execution Details of the Generic Test contains information about the Test Suite, which in this case corresponds to the Test Case class.



References

- http://www.seleniumhq.org/docs/05_selenium_rc.jsp#java
- https://github.com/eviltester/startUsingSeleniumWebDriver/blob/master/src/test/java/com/eviltester/webdriver/MyFirstChromeTest.java