

# Testing Windows Applications using Appium and JUnit in Java

## Overview

In this tutorial, we will create a JUnit Test Case in Java, using the [Appium](#) library for automation of Windows applications.

## Description

The following automated test class validates the "Calculator" Windows native application using several methods (i.e., Tests) for validating each arithmetic operation.



### Please note

This example is taken from the public Github repository <https://github.com/Microsoft/WinAppDriver/tree/master/Samples/Java/CalculatorTest>. It also provides examples for other languages.

## Requirement

- WinAppDriver must be running in the destination machine (i.e., the one having Windows and the "Calculator" application), or the Appium Desktop.

```
WinAppDriver.exe 192.168.56.102 4723
```

We will make a simple update to the pom.xml file in order to generate a JUnit xml report.

## pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>CalculatorTest</groupId>
  <artifactId>CalculatorTest</artifactId>
  <version>1.0-SNAPSHOT</version>

  <dependencies>
    <dependency>
      <groupId>org.seleniumhq.selenium</groupId>
      <artifactId>selenium-java</artifactId>
      <version>3.3.1</version>
    </dependency>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.12</version>
    </dependency>
    <dependency>
      <groupId>io.appium</groupId>
      <artifactId>java-client</artifactId>
      <version>5.0.0-BETA6</version>
    </dependency>
  </dependencies>

  <reporting>
    <plugins>
      <plugin>
        <artifactId>maven-surefire-report-plugin</artifactId>
      </plugin>
    </plugins>
  </reporting>
</project>
```

The class implementing the automated tests needs to be updated in order to properly set up the IP of the Appium server.

```
//*****
//
// Copyright (c) 2016 Microsoft Corporation. All rights reserved.
//
// This code is licensed under the MIT License (MIT).
//
// THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
// IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
// FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
// AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
// LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
// OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
// THE SOFTWARE.
//
//*****

import org.junit.*;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.remote.DesiredCapabilities;
import java.util.concurrent.TimeUnit;
import java.net.URL;
import io.appium.java_client.windows.WindowsDriver;
```

```

public class CalculatorTest {

    private static WindowsDriver CalculatorSession = null;
    private static WebElement CalculatorResult = null;

    @BeforeClass
    public static void setup() {
        try {
            DesiredCapabilities capabilities = new DesiredCapabilities();
            capabilities.setCapability("app", "Microsoft.WindowsCalculator_8wekyb3d8bbwe!App");
            CalculatorSession = new WindowsDriver(new URL("http://192.168.56.102:4723"), capabilities);
            CalculatorSession.manage().timeouts().implicitlyWait(2, TimeUnit.SECONDS);

            CalculatorResult = CalculatorSession.findElementByAccessibilityId("CalculatorResults");
            Assert.assertNotNull(CalculatorResult);

        } catch (Exception e) {
            e.printStackTrace();
        } finally {
        }
    }

    @Before
    public void Clear()
    {
        CalculatorSession.findElementByName("Clear").click();
        Assert.assertEquals("0", _GetCalculatorResultText());
    }

    @AfterClass
    public static void TearDown()
    {
        CalculatorResult = null;
        if (CalculatorSession != null) {
            CalculatorSession.quit();
        }
        CalculatorSession = null;
    }

    @Test
    public void Addition()
    {
        CalculatorSession.findElementByName("One").click();
        CalculatorSession.findElementByName("Plus").click();
        CalculatorSession.findElementByName("Seven").click();
        CalculatorSession.findElementByName("Equals").click();
        Assert.assertEquals("8", _GetCalculatorResultText());
    }

    @Test
    public void Combination()
    {
        CalculatorSession.findElementByName("Seven").click();
        CalculatorSession.findElementByName("Multiply by").click();
        CalculatorSession.findElementByName("Nine").click();
        CalculatorSession.findElementByName("Plus").click();
        CalculatorSession.findElementByName("One").click();
        CalculatorSession.findElementByName("Equals").click();
        CalculatorSession.findElementByName("Divide by").click();
        CalculatorSession.findElementByName("Eight").click();
        CalculatorSession.findElementByName("Equals").click();
        Assert.assertEquals("8", _GetCalculatorResultText());
    }

    @Test
    public void Division()
    {
        CalculatorSession.findElementByName("Eight").click();
        CalculatorSession.findElementByName("Eight").click();
        CalculatorSession.findElementByName("Divide by").click();
        CalculatorSession.findElementByName("One").click();
    }
}

```

```

        CalculatorSession.findElementByName("One").click();
        CalculatorSession.findElementByName("Equals").click();
        Assert.assertEquals("8", _GetCalculatorResultText());
    }

    @Test
    public void Multiplication()
    {
        CalculatorSession.findElementByName("Nine").click();
        CalculatorSession.findElementByName("Multiply by").click();
        CalculatorSession.findElementByName("Nine").click();
        CalculatorSession.findElementByName("Equals").click();
        Assert.assertEquals("81", _GetCalculatorResultText());
    }

    @Test
    public void Subtraction()
    {
        CalculatorSession.findElementByName("Nine").click();
        CalculatorSession.findElementByName("Minus").click();
        CalculatorSession.findElementByName("One").click();
        CalculatorSession.findElementByName("Equals").click();
        Assert.assertEquals("8", _GetCalculatorResultText());
    }

    protected String _GetCalculatorResultText()
    {
        // trim extra text and whitespace off of the display value
        return CalculatorResult.getText().replace("Display is", "").trim();
    }
}

```

Tests can be run using Maven.

```
mvn clean test
```

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

Each 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.

Tests

+ Add

Overall Execution Status



TOTAL TESTS: 5

FILTERS

Test Set: All Assignee: All Status: Component: Search: Contains text X Clear

Show 100 entries Columns

	Key	Summary	Test Type	#Req	#Def	Test Sets	Assignee	Status	
1	CALC-1309	Division	Generic	0	0		Administrator	PASS	
2	CALC-1310	Subtraction	Generic	0	0		Administrator	PASS	
3	CALC-1311	Addition	Generic	0	0		Administrator	PASS	
4	CALC-1312	Multiplication	Generic	0	0		Administrator	PASS	
5	CALC-1313	Combination	Generic	0	0		Administrator	PASS	

Showing 1 to 5 of 5 entries First Previous 1 Next Last

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

Execution Status: PASS

Started On: Today 5:33 PM Finished On: Today 5:33 PM

Assignee: Administrator Executed By: Administrator

Execution Defects (0) Execution Evidences (0)

Execution Details

Test Description: None

Test Details: Test Type: Generic Definition: CalculatorTest.Multiplication

Results: Context: TestSuite CalculatorTest Error Message: - Duration: 1 sec Status: PASS

References

- https://github.com/Microsoft/WinAppDriver
- http://appium.io