Lab 6. Testing HTML

# Setup

1. Go to <http://automationpractice.com/index.php> and create an account
2. Go to **File -> New -> Spring Starter Project** to create a new Spring Project
3. Type the following information in the wizard:

**Service URL:** <https://start.spring.io>

**Name:** testHtml

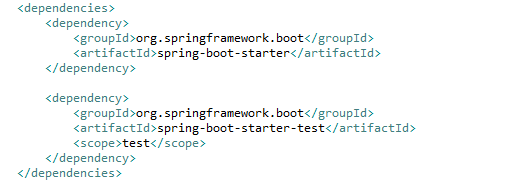
**Group:** mx.tec.lab

**Artifact**: testHtml

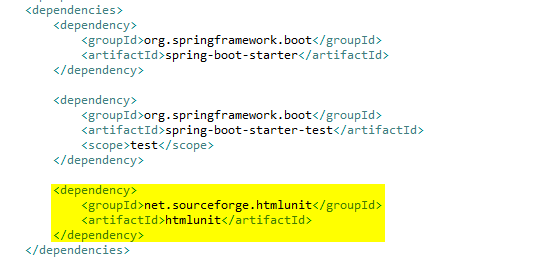
**Description:** Testing HTML Test Lab

**Package:** mx.tec.lab

1. Select Spring Boot Version 2.4.3 and click Finish
2. Open pom.xml file and verify you have the following dependencies



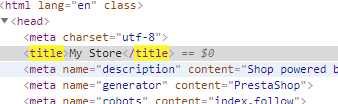
1. Verify that you have the **TestHtmlApplication** class in src/main/java (Package Explorer)
2. Verify that you have the **TestHtmlApplicationTests** class in src/test/java (Package Explorer)
3. Add a new dependency in your **pom.xml** file



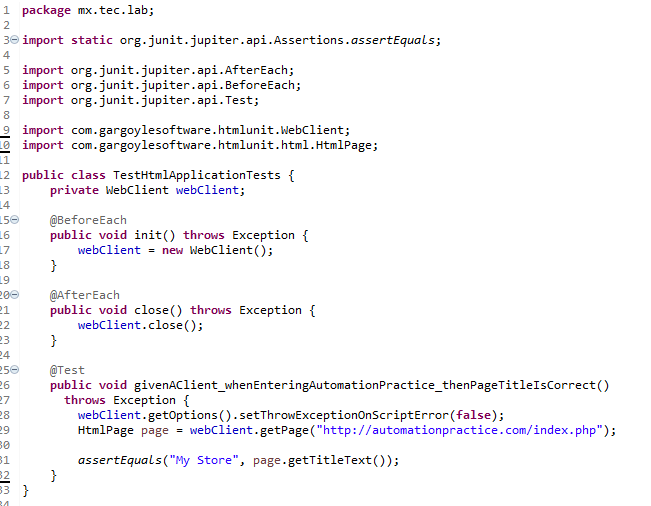
1. Click on **Run As -> Maven clean**
2. Verify that clean was successful in the Console
3. Click on **Run As -> Maven install**
4. Verify that install was successful in the Console

# Exercise

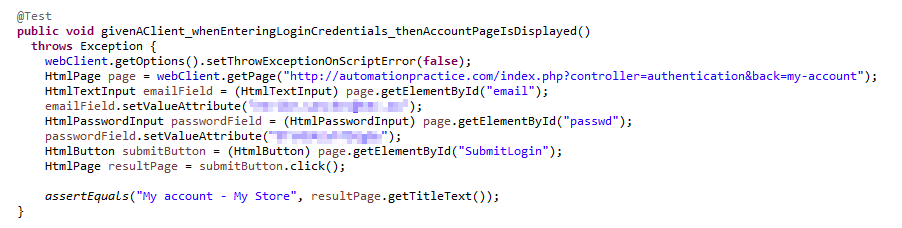
1. Go to <http://automationpractice.com/index.php>, inspect the code and check the title element.



1. Modify the TestHtmlApplicationTests class



1. Run the test and see the results
2. Go to the signin page in the browser
3. Check the elements with this ids in the inspector: login\_form, email, passwd and SubmitLogin
4. Process the login with the credentials you obtained
5. Inspect the title in the resulting page
6. Add a new test method



1. Add the following methods



1. Don’t forget the import

**import** **static** org.junit.jupiter.api.Assertions.*fail*;

# Challenge

1. Make the methods in step 9 pass successfully

* <http://htmlunit.sourceforge.net/apidocs/index.html>
* Tip: getFirstByXPath("//div[@class='alert alert-danger']/ol/li")