

Capstone Project

@RestAssured

#GetAllShoes

```
package com.SportyShoe.restAssuredScripts;
```

```
import org.testng.annotations.Test;
```

```
import io.restassured.RestAssured;
```

```
public class GetAllshoes {
```

```
    @Test(priority=1)
    public void get_all_shoes()
    {
        RestAssured.given()
            .baseUrl("http://localhost:9010")
            .basePath("/get-shoes")
            .when()
            .get()
            .then()
            .statusCode(200)
            .and()
            .log()
            .all();
    }
```

```
    @Test(priority='2')
    public void get_all_users()
    {
        RestAssured.given()
            .baseUrl("http://localhost:9010")
            .basePath("/get-users")
            .when()
            .get()
            .then()
            .statusCode(401)
            .log()
            .all();
    }
}
```

#Post,Put and delete All Shoes

```
package com.SportyShoe.restAssuredScripts;
```

```
import org.testng.annotations.Test;
```

```
import io.restassured.RestAssured;
```

```
public class PostAndPutNewShoe {
```

```
@Test(priority='1')
public void add_new_product()
{

    RestAssured.given()
        .baseUrl("http://localhost:9010")
        .basePath("/add-shoe")
        .queryParams("id", "1020")
        .queryParams("image", "www.imge.com")
        .queryParams("name", "Nike")
        .queryParams("category", "Running")
        .queryParams("sizes", "5,6,7")
        .queryParams("price", "2000")
        .when()
        .post()
        .then()
        .log().all();

}
```

```
@Test(priority='2')
public void update_a_product()
{

    RestAssured.given()
        .baseUrl("http://localhost:9010")
        .basePath("/update-shoe")
        .queryParams("id", "1020")
        .queryParams("image", "www.imge123.com")
        .queryParams("name", "Reebok")
        .queryParams("category", "Running")
        .queryParams("sizes", "5,6,7")
        .queryParams("price", "2500")
        .when()
        .put()
        .then()
        .log().all();

}
```

```
@Test(priority='3')
public void delete_product()
{

    RestAssured.given()
        .baseUrl("http://localhost:9010")
        .basePath("/delete-shoe")
        .queryParams("id", "1020")
        .when().delete()
        .then().log().all();

}
```

```
}
```

@Selenium

#TestBase

```
package com.SportyShoe.SeleniumCucumberScripts;
```

```
import org.openqa.selenium.WebDriver;  
import org.openqa.selenium.chrome.ChromeDriver;  
import org.openqa.selenium.firefox.FirefoxDriver;
```

```
public class TestBase {  
  
    public static WebDriver driver;  
  
    public static void OpenBrowser(String browser)  
    {  
        if(browser == "Chrome")  
        {  
            driver = new ChromeDriver();  
        }  
  
        if(browser == "FireFox")  
        {  
            driver = new FirefoxDriver();  
        }  
  
        driver.manage().window().maximize();  
        driver.manage().deleteAllCookies();  
        driver.get("http://localhost:9010/");  
    }  
  
    public static void closebrowser()  
    {  
        driver.close();  
    }  
}
```

#HomePage

```
package com.SportyShoe.SeleniumCucumberScripts;
```

```
import org.openqa.selenium.WebDriver;  
import org.openqa.selenium.WebElement;  
import org.openqa.selenium.support.FindBy;  
import org.openqa.selenium.support.PageFactory;
```

```
public class HomePage extends TestBase {  
  
    @FindBy(xpath="//div[@class='container mt-3']/descendant::p[1]")  
    WebElement text;  
  
    @FindBy(linkText="New User? Register Here")
```

```

        WebElement registerLink;

        public HomePage(WebDriver driver) {
            PageFactory.initElements(driver, this);
        }

        public String getURL_page()
        {
            String URLnew = driver.getCurrentUrl();
            return URLnew;
        }

        public String Validate_Text_On_Page()
        {
            String pageText = text.getText();
            System.out.println(pageText);
            return pageText;
        }

        public void click_register_link()
        {
            registerLink.click();
        }
    }

```

#HomePageTest

```

package com.SportyShoe.Tests;

import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

import com.SportyShoe.SeleniumCucumberScripts.HomePage;
import com.SportyShoe.SeleniumCucumberScripts.TestBase;

import static org.testng.Assert.assertEquals;

import org.testng.Assert;
import org.testng.Assert.*;

public class HomePageTest extends TestBase {

    HomePage hp;

    @BeforeTest
    public void start_browser()
    {
        OpenBrowser("Chrome");
        hp = new HomePage(driver);
    }

    @Test(priority='1')

```

```

    public void test_getTitle_page()
    {
        String expected = "http://localhost:9010/";
        String Actual = hp.getURL_page();
        Assert.assertEquals(Actual, expected);
    }

    @Test(priority='2')

    public void Test_Validate_Text_On_Page()
    {
        String expected = "Powered By Simplilearn";
        String actualText = hp.Validate_Text_On_Page();
        Assert.assertEquals(actualText, expected);
    }

    @Test(priority='3')
    public void test_click_register_link() throws InterruptedException
    {
        Thread.sleep(1500);
        hp.click_register_link();
    }
}

```

#LoginPage

```
package com.SportyShoe.SeleniumCucumberScripts;
```

```

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;

```

```

public class LoginPage {

    @FindBy(xpath="//input[@id='email']")
    WebElement loginEmail;

    @FindBy(xpath="//input[@id='password']")
    WebElement loginpassword;

    @FindBy(xpath="//button[@type='submit']")
    WebElement loginbtn;

    @FindBy(linkText="Cart")
    WebElement clickCart;

    public LoginPage(WebDriver driver) {
        PageFactory.initElements(driver, this);
    }
}

```

#LoginPageTest

```
package com.SportyShoe.Tests;
```

```
import org.testng.Assert;  
import org.testng.annotations.BeforeTest;  
import org.testng.annotations.Test;
```

```
import com.SportyShoe.SeleniumCucumberScripts.HomePage;  
import com.SportyShoe.SeleniumCucumberScripts.LoginPage;  
import com.SportyShoe.SeleniumCucumberScripts.RegisterPage;  
import com.SportyShoe.SeleniumCucumberScripts.TestBase;
```

```
public class LoginPageTest extends TestBase {
```

```
    HomePage hp;  
    RegisterPage rp;  
    LoginPage lp;
```

```
    @BeforeTest  
    public void start_browser()  
    {  
        OpenBrowser("Chrome");  
        hp = new HomePage(driver);  
        rp = new RegisterPage(driver);  
        lp = new LoginPage(driver);  
    }
```

```
    @Test(priority='1')  
    public void test_login()  
    {  
        lp.user_login();  
    }
```

```
    @Test(priority='2')  
    public void test_getTitle_page()  
    {  
        String expected = "http://localhost:9010/login";  
        String Actual = hp.getURL_page();  
        Assert.assertEquals(Actual, expected);  
    }
```

```
    @Test(priority='3')  
  
    public void Test_validate_registration_Text()  
    {  
        String expected = "Hello Ashu !";  
        String actualText = rp.validate_registration_Text();  
        Assert.assertEquals(actualText, expected);  
    }
```

```
@Test(priority='4')
```

```
public void test_click_cart()
{
    lp.click_cart();
}
```

```
}
```

```
public void user_login()
{
    loginEmail.sendKeys("ashu@gmail.com");
    loginpassword.sendKeys("ashu@123");
    loginbtn.click();
}
```

```
public void click_cart()
{
    clickCart.click();
}
```

```
}
```

#RegisterPage

```
package com.SportyShoe.SeleniumCucumberScripts;
```

```
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
```

```
public class RegisterPage extends TestBase{
```

```
    @FindBy(xpath="//input[@id='name']")
    WebElement registername;
```

```
    @FindBy(xpath="//input[@id='email']")
    WebElement registeremail;
```

```
    @FindBy(xpath="//input[@id='password']")
    WebElement registerpassword;
```

```
    @FindBy(xpath="//button[@type='submit']")
    WebElement registerBtn;
```

```
    @FindBy(xpath="//div[@class='mt-4 p-5 bg-primary text-white
rounded']/descendant::p[3]")
    WebElement userText;
```

```
public RegisterPage(WebDriver driver) {
```

```

        PageFactory.initElements(driver, this);
    }

    public void register_user()
    {
        registername.sendKeys("ashu");
        registeremail.sendKeys("ashu@gmail.com");
        registerpassword.sendKeys("ashu@123");
        registerBtn.click();
    }

    public String validate_registration_URL()
    {
        String register_url = driver.getCurrentUrl();
        return register_url;
    }

    public String validate_registration_Text()
    {
        String user_name = userText.getText();
        return user_name;
    }
}

```

#RegisterPageTest

```

package com.SportyShoe.Tests;

import static org.testng.Assert.assertEquals;
import org.testng.Assert;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;
import com.SportyShoe.SeleniumCucumberScripts.HomePage;
import com.SportyShoe.SeleniumCucumberScripts.RegisterPage;
import com.SportyShoe.SeleniumCucumberScripts.TestBase;
public class RegisterPageTest extends TestBase {

    HomePage hp;
    RegisterPage rp;

    @BeforeTest
    public void start_browser()
    {
        OpenBrowser("Chrome");
        hp = new HomePage(driver);
        rp = new RegisterPage(driver);
    }

    @Test(priority='1')
    public void test_click_register_link() throws InterruptedException
    {
        Thread.sleep(1500);
        hp.click_register_link();
    }
}

```



```

    }

    @Test(priority='2')
    public void test_getTitle_page()
    {
        String expected = "http://localhost:9010/register";
        String Actual = hp.getURL_page();
        Assert.assertEquals(Actual, expected);
    }

    @Test(priority='3')
    public void Test_register_user()
    {
        rp.register_user();
    }

    @Test(priority='4')
    public void Test_validate_registration_URL()
    {
        String expected = "http://localhost:9010/register-user";
        String Actual = rp.validate_registration_URL();
        assertEquals(Actual, expected);
    }

    @Test(priority='5')

    public void Test_validate_registration_Text()
    {
        String expected = "Hello Ashu !";
        String actualText = rp.validate_registration_Text();
        Assert.assertEquals(actualText, expected);
    }

}

```

#AddCartPage

```
package com.SportyShoe.SeleniumCucumberScripts;
```

```
import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
```

```
public class AddtoCartPage {
```

```
    @FindBy(xpath="//a[@id='shoe101']")
    WebElement viewShoeBTN;
```

```
    @FindBy(xpath = "//a[@id='cart101']")
    WebElement addtocartBTN;
```

```

@FindBy(xpath="//div[@class='alert alert-success']/descendant::p[1]")
WebElement successText;

JavascriptExecutor executor;

public AddtoCartPage(WebDriver driver) {
    PageFactory.initElements(driver, this);
    executor = (JavascriptExecutor) driver;
}

public void add_product_to_cart() throws InterruptedException
{

    executor.executeScript("arguments[0].click();", addtocartBTN);

}

public String validate_success_message()
{

    String successtext = successText.getText();
    return successtext;

}
}

```

#Add Cart Page Test

```

package com.SportyShoe.Tests;

import org.testng.Assert;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

import com.SportyShoe.SeleniumCucumberScripts.AddtoCartPage;
import com.SportyShoe.SeleniumCucumberScripts.HomePage;
import com.SportyShoe.SeleniumCucumberScripts.LoginPage;
import com.SportyShoe.SeleniumCucumberScripts.RegisterPage;
import com.SportyShoe.SeleniumCucumberScripts.TestBase;

public class AddtoCartPageTest extends TestBase {

    HomePage hp;
    RegisterPage rp;
    LoginPage lp;
    AddtoCartPage ac;

    @BeforeTest
    public void start_browser()
    {

```

```

        OpenBrowser("Chrome");
        hp = new HomePage(driver);
        rp = new RegisterPage(driver);
        lp = new LoginPage(driver);
        ac = new AddtoCartPage(driver);
    }

    @Test(priority='1')
    public void test_login()
    {
        lp.user_login();
    }

    @Test(priority='2')
    public void test_add_product_to_cart() throws InterruptedException
    {
        ac.add_product_to_cart();
    }

    @Test(priority='3')
    public void test_validate_success_message()
    {
        String expected = "Message:Shoe BlueWave Running Shoes Added
Successfully to Cart";
        String actualText=    ac.validate_success_message();
        Assert.assertEquals(actualText, expected);
    }
}

```

#OrderPage

```
package com.SportyShoe.SeleniumCucumberScripts;
```

```
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
```

```
public class OrderPage {

    @FindBy(linkText="Orders")
    WebElement orderlink;

    public OrderPage(WebDriver driver) {
        PageFactory.initElements(driver, this);
    }

    public void click_orderPage()
    {
        orderlink.click();
    }

}

```

#OrderPageTest

```
package com.SportyShoe.Tests;

import org.testng.Assert;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;
import com.SportyShoe.SeleniumCucumberScripts.HomePage;
import com.SportyShoe.SeleniumCucumberScripts.LoginPage;
import com.SportyShoe.SeleniumCucumberScripts.OrderPage;
import com.SportyShoe.SeleniumCucumberScripts.RegisterPage;
import com.SportyShoe.SeleniumCucumberScripts.TestBase;

public class OrderPageTest extends TestBase {

    HomePage hp;
    RegisterPage rp;
    LoginPage lp;
    OrderPage op;

    @BeforeTest
    public void start_browser()
    {
        OpenBrowser("Chrome");
        hp = new HomePage(driver);
        rp = new RegisterPage(driver);
        lp = new LoginPage(driver);
        op = new OrderPage(driver);
    }

    @Test(priority='1')
    public void test_login()
    {
        lp.user_login();
    }

    @Test(priority='2')

    public void test_click_orders()
    {
        op.click_orderPage();
    }

    @Test(priority='3')
    public void test_getTitle_page()
    {
        String expected = "http://localhost:9010/orders";
        String Actual = hp.getURL_page();
        Assert.assertEquals(Actual, expected);
    }
}
```

#pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<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>ATE-CapstoneProject</groupId>
  <artifactId>ATE-CapstoneProject</artifactId>
  <version>0.0.1-SNAPSHOT</version>

  <name>ATE-CapstoneProject</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>

  <dependencies>
    <dependency>
      <groupId>io.rest-assured</groupId>
      <artifactId>rest-assured</artifactId>
      <version>5.3.0</version>
      <scope>test</scope>
    </dependency>
    <!-- https://mvnrepository.com/artifact/io.rest-assured/rest-assured-common -->
    <dependency>
      <groupId>io.rest-assured</groupId>
      <artifactId>rest-assured-common</artifactId>
      <version>5.3.0</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/io.rest-assured/rest-assured-all -->
    <dependency>
      <groupId>io.rest-assured</groupId>
      <artifactId>rest-assured-all</artifactId>
      <version>5.3.0</version>
      <scope>test</scope>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.json/json -->
    <dependency>
      <groupId>org.json</groupId>
      <artifactId>json</artifactId>
      <version>20180813</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.hamcrest/hamcrest -->
    <dependency>
      <groupId>org.hamcrest</groupId>
      <artifactId>hamcrest</artifactId>
      <version>2.2</version>
```

```

    <scope>test</scope>
</dependency>
<!-- https://mvnrepository.com/artifact/io.rest-assured/json-path -->
<dependency>
    <groupId>io.rest-assured</groupId>
    <artifactId>json-path</artifactId>
    <version>5.3.0</version>
    <scope>test</scope>
</dependency>
<dependency>
    <groupId>com.fasterxml.jackson.core</groupId>
    <artifactId>jackson-databind</artifactId>
    <version>2.15.2</version>
</dependency>

    <!-- https://mvnrepository.com/artifact/org.testng/testng -->
<dependency>
    <groupId>org.testng</groupId>
    <artifactId>testng</artifactId>
    <version>7.7.1</version>
    <scope>test</scope>
</dependency>
<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->
<dependency>
    <groupId>io.cucumber</groupId>
    <artifactId>cucumber-java</artifactId>
    <version>7.10.1</version>
</dependency>

    <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->
<dependency>
    <groupId>io.cucumber</groupId>
    <artifactId>cucumber-junit</artifactId>
    <version>7.10.1</version>
    <scope>compile</scope>
</dependency>

    <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->
<dependency>
    <groupId>io.cucumber</groupId>
    <artifactId>cucumber-core</artifactId>
    <version>7.10.1</version>
</dependency>

<dependency>
    <groupId>org.apache.logging.log4j</groupId>
    <artifactId>log4j-api</artifactId>
    <version>2.14.1</version>
</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.logging.log4j/log4j-core -->
<dependency>
    <groupId>org.apache.logging.log4j</groupId>

```

```

    <artifactId>log4j-core</artifactId>
    <version>2.14.1</version>
</dependency>
<dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>4.11.0</version>
</dependency>

<!-- https://mvnrepository.com/artifact/io.github.bonigarcia/webdrivermanager -->
<dependency>
    <groupId>io.github.bonigarcia</groupId>
    <artifactId>webdrivermanager</artifactId>
    <version>4.4.3</version>
</dependency>
</dependencies>

<build>
    <pluginManagement><!-- lock down plugins versions to avoid using Maven
defaults (may be moved to parent pom) -->
    <plugins>
        <!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-
core/lifecycles.html#clean_Lifecycle -->
        <plugin>
            <artifactId>maven-clean-plugin</artifactId>
            <version>3.1.0</version>
        </plugin>
        <!-- default lifecycle, jar packaging: see
https://maven.apache.org/ref/current/maven-core/default-
bindings.html#Plugin_bindings_for_jar_packaging -->
        <plugin>
            <artifactId>maven-resources-plugin</artifactId>
            <version>3.0.2</version>
        </plugin>
        <plugin>
            <artifactId>maven-compiler-plugin</artifactId>
            <version>3.8.0</version>
        </plugin>
        <plugin>
            <artifactId>maven-surefire-plugin</artifactId>
            <version>2.22.1</version>
        </plugin>
        <plugin>
            <artifactId>maven-jar-plugin</artifactId>
            <version>3.0.2</version>
        </plugin>
        <plugin>
            <artifactId>maven-install-plugin</artifactId>
            <version>2.5.2</version>
        </plugin>
        <plugin>
            <artifactId>maven-deploy-plugin</artifactId>
            <version>2.8.2</version>
        </plugin>
    </plugins>

```

```
<!-- site lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#site\_Lifecycle -->
<plugin>
  <artifactId>maven-site-plugin</artifactId>
  <version>3.7.1</version>
</plugin>
<plugin>
  <artifactId>maven-project-info-reports-plugin</artifactId>
  <version>3.0.0</version>
</plugin>
</plugins>
</pluginManagement>
</build>
</project>
```