

SourceCode-Selenium-TestNg-Project

AddToCartPageTest.java

```
package com.simplilearn.Selenium;

import static org.testng.Assert.assertEquals;
import static org.testng.Assert.assertNotEquals;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

public class AddProductToCartPageTest {

    WebDriver driver=null;

    String siteurl="http://localhost:9010/";

    String driverpath="drivers/windows/chromedriver.exe";

    @BeforeTest

    public void setup() {

        System.setProperty("webdriver.chrome.driver", driverpath);

        driver=new ChromeDriver();

        driver.get(siteurl);

    }
```

```

@AfterTest

public void cleanup() {
    if(driver!=null)
        driver.quit();
}

@Test(description="Test add to cart title match",priority=1)
public void addCartPageTest() throws InterruptedException {

    Thread.sleep(2000);
    driver.findElement(By.id("email")).sendKeys("thavik@gmail.com");
    driver.findElement(By.id("password")).sendKeys("Thavik@000");
    Thread.sleep(2000);
    driver.findElement(By.xpath("/html/body/div[2]/form/button")).submit();
    Thread.sleep(2000);
    driver.findElement(By.xpath("//*[@id=\"mynavbar\"]/ul/li[1]/a")).click();

}

@Test(description="Test add cart page source url",priority=2)
public void addCartPageSourceUrlTest() {
    String url=driver.getCurrentUrl();
    assertEquals("http://localhost:9010/cart",url);
}

@Test(description="Test sport shoes registartion page title ",priority=3)
public void homePageTitleTest() {

    String actualTitle=" ";
    assertEquals(actualTitle,driver.getTitle());
}

```

```
        System.out.println(driver.getTitle());
    }
}
```

LoginPageTest.java

```
package com.simplilearn.Selenium;

import static org.testng.Assert.assertEquals;
import static org.testng.Assert.assertNotEquals;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

public class LoginPageTest {

    WebDriver driver=null;
    String siteurl="http://localhost:9010/";
    String driverpath="drivers/windows/chromedriver.exe";

    @BeforeTest
    public void setup() {

        System.setProperty("webdriver.chrome.driver", driverpath);
```

```

driver=new ChromeDriver();

        driver.get(siteurl);
    }

    @AfterTest
    public void cleanup() {
        driver.quit();
    }

    @Test(description="Test Medicare login page title match",priority=1)
    public void loginPageTitleTest() throws InterruptedException {
        Thread.sleep(2000);
        driver.findElement(By.id("email")).sendKeys("thavik@gmail.com");
        driver.findElement(By.id("password")).sendKeys("Thavik@000");
        Thread.sleep(2000);
        driver.findElement(By.xpath("/html/body/div[2]/form/button")).submit();

        String expectedTitle = "";
        String actualTitle = driver.getTitle();
        assertEquals(actualTitle, expectedTitle);
    }

    @Test(description="Test Medicare login page source url",priority=2)
    public void loginPageSourceUrlTest() {
        String url=driver.getCurrentUrl();
        assertEquals("http://localhost:9010/login",url);
    }

    @Test(description="Test Medicare login page title invalid match",priority=3)
    public void loginPageTitleTest2() {
        String actualTitle=" ";
        assertEquals(actualTitle,driver.getTitle());
    }

```

```
}
```

```
}
```

PlaceOrderTest.java

```
package com.simplilearn.Selenium;

import static org.testng.Assert.assertEquals;
import static org.testng.Assert.assertNotEquals;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

public class PlaceOrderTest {

    WebDriver driver=null;
    String siteurl="http://localhost:9010/";
    String driverpath="drivers/windows/chromedriver.exe";

    @BeforeTest
    public void setup() {
```

```

        System.setProperty("webdriver.chrome.driver", driverpath);
driver=new ChromeDriver();

        driver.get(siteurl);

    }

    @AfterTest
    public void cleanup() {
        if(driver!=null)
            driver.quit();
    }

    @Test(description="Test add to cart title match",priority=1)
    public void addCartPageTest() throws InterruptedException {

        Thread.sleep(2000);

        driver.findElement(By.id("email")).sendKeys("thavik@gmail.com");
        driver.findElement(By.id("password")).sendKeys("Thavik@000");
        Thread.sleep(2000);

        driver.findElement(By.xpath("/html/body/div[2]/form/button")).submit();
        Thread.sleep(2000);
        driver.findElement(By.xpath("//*[@id=\"mynavbar\"]/ul/li[2]/a")).click();

    }

    @Test(description="Test add cart page source url",priority=2)
    public void addCartPageSourceUrlTest() {
        String url=driver.getCurrentUrl();
        assertEquals("http://localhost:9010/orders",url);
    }

    @Test(description="Test sport shoes registartion page title ",priority=3)

```

```
public void homePageTitleTest() {  
  
    String actualTitle=" ";  
    assertEquals(actualTitle,driver.getTitle());  
    System.out.println(driver.getTitle());  
}  
}
```

RegistrationPageTest.java

```
package com.simplilearn.Selenium;  
  
import static org.testng.Assert.assertEquals;  
import static org.testng.Assert.assertNotEquals;  
  
import org.openqa.selenium.By;  
import org.openqa.selenium.WebDriver;  
import org.openqa.selenium.chrome.ChromeDriver;  
import org.testng.annotations.AfterTest;  
import org.testng.annotations.BeforeTest;  
import org.testng.annotations.Test;  
  
public class RegistrationPageTest {  
  
    WebDriver driver=null;  
    String siteurl="http://localhost:9010/register";
```

```
String driverpath="drivers/windows/chromedriver.exe";
```

```
@BeforeTest
```

```
public void setup() {
```

```
    System.setProperty("webdriver.chrome.driver", driverpath);
```

```
    driver=new ChromeDriver();
```

```
    driver.get(siteurl);
```

```
}
```

```
@AfterTest
```

```
public void cleanup() {
```

```
    if(driver!=null)
```

```
        driver.quit();
```

```
}
```

```
@Test(description="Test Medicare register page title match",priority=1)
```

```
public void registrationPageTest() throws InterruptedException {
```

```
    Thread.sleep(2000);
```

```
    driver.findElement(By.id("name")).sendKeys("Thavik");
```

```
    driver.findElement(By.id("email")).sendKeys("thavik@gmail.com");
```

```
    Thread.sleep(2000);
```

```
    driver.findElement(By.id("password")).sendKeys("Thavik@000");
```

```
    driver.findElement(By.xpath("/html/body/div[2]/form/button")).submit();
```

```
}
```

```
@Test(description="Test Medicare registration page source url",priority=2)
```

```
public void homePageSourceUrlTest() {
```



```

        String url=driver.getCurrentUrl();

        assertEquals("http://localhost:9010/register-user",url);
    }

    @Test(description="Test Medicare registartion page title ",priority=3)
    public void homePageTitleTest() {

        String actualTitle=" ";

        assertEquals(actualTitle,driver.getTitle());

        System.out.println(driver.getTitle());
    }
}

```

SearchProductPageTest.java

```

package com.simplilearn.Selenium;

import static org.testng.Assert.assertEquals;
import static org.testng.Assert.assertNotEquals;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

public class SearchProductPageTest {

```

```

WebDriver driver=null;

String siteurl="http://localhost:9010/";

String driverpath="drivers/windows/chromedriver.exe";


@BeforeTest

public void setup() {

    System.setProperty("webdriver.chrome.driver", driverpath);

    driver=new ChromeDriver();

    driver.get(siteurl);

}

@AfterTest

public void cleanup() {

    if(driver!=null)

        driver.quit();

}

@Test(description="Test search product title match",priority=1)

public void addCartPageTest() throws InterruptedException {

    Thread.sleep(2000);

    driver.findElement(By.id("email")).sendKeys("thavik@gmail.com");

    driver.findElement(By.id("password")).sendKeys("Thavik@000");

    Thread.sleep(2000);

    driver.findElement(By.xpath("/html/body/div[2]/form/button")).submit();

    Thread.sleep(2000);

    driver.findElement(By.xpath("//*[@id=\"mynavbar\"]/ul/li[2]/a")).click();

```

```

    }

    @Test(description="Test search product page source url",priority=2)
    public void addCartPageSourceUrlTest() throws InterruptedException {

        Thread.sleep(2000);
//        driver.findElement(By.id("search-product")).sendKeys("fever");
//        driver.findElement(By.id("search-product-button")).click();

        String url=driver.getCurrentUrl();

        assertEquals("http://localhost:9010/orders",url);
    }

    @Test(description="Test search product page title ",priority=3)
    public void homePageTitleTest() {

        String actualTitle=" ";

        assertEquals(actualTitle,driver.getTitle());

        System.out.println(driver.getTitle());
    }

}

```

FillterButtonTest.java

```

package com.simplilearn.Selenium;

import static org.testng.Assert.assertEquals;
import static org.testng.Assert.assertNotEquals;

import org.openqa.selenium.By;

```

```
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.AfterTest;
import org.testng.annotations.BeforeTest;
import org.testng.annotations.Test;

public class FillterButtonTest {

    WebDriver driver=null;

    String siteurl="http://localhost:9010/";

    String driverpath="drivers/windows/chromedriver.exe";

    @BeforeTest
    public void setup() {

        System.setProperty("webdriver.chrome.driver", driverpath);

        driver=new ChromeDriver();

        driver.get(siteurl);

    }

    @AfterTest
    public void cleanup() {
        if(driver!=null)
            driver.quit();
    }

    @Test(description="Test search product title match",priority=1)
    public void addCartPageTest() throws InterruptedException {
```

```

        Thread.sleep(2000);

        driver.findElement(By.id("email")).sendKeys("thavik@gmail.com");

        driver.findElement(By.id("password")).sendKeys("Thavik@000");

        Thread.sleep(2000);

        driver.findElement(By.xpath("/html/body/div[2]/form/button")).submit();

Thread.sleep(2000);
driver.findElement(By.xpath("//*[@id=\"mynavbar\"]/ul/li[2]/a")).click();

    }

    @Test(description="Test search product page source url",priority=2)
    public void addCartPageSourceUrlTest() throws InterruptedException {

        Thread.sleep(2000);
//        driver.findElement(By.id("fillter-button")).click();
//        driver.findElement(By.id("lth")).click();

        String url=driver.getCurrentUrl();

        assertEquals("http://localhost:9010/orders",url);

    }

    @Test(description="Test search product page title ",priority=3)
    public void homePageTitleTest() {

        String actualTitle=" ";

        assertEquals(actualTitle,driver.getTitle());

        System.out.println(driver.getTitle());

    }

}

```

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>com.simplilearn.Selenium</groupId>
  <artifactId>ATE-Capstone_Selenium_TestNG_Project</artifactId>
  <version>0.0.1-SNAPSHOT</version>

  <name>ATE-Capstone_Selenium_TestNG_Project</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>org.seleniumhq.selenium</groupId>
      <artifactId>selenium-java</artifactId>
      <version>4.15.0</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.testng/testng -->
  <dependency>
    <groupId>org.testng</groupId>
    <artifactId>testng</artifactId>
    <version>7.8.0</version>
    <scope>test</scope>
  </dependency>

    <!-- reporting -->
    <dependency>
      <groupId>org.uncommons</groupId>
      <artifactId>reportng</artifactId>
      <version>1.1.4</version>
      <scope>test</scope>
      <exclusions>
        <exclusion>
          <groupId>org.testng</groupId>
          <artifactId>testng</artifactId>
        </exclusion>
      </exclusions>
    </dependency>

    <!-- com.google.inject/guice -->
    <dependency>
      <groupId>com.google.inject</groupId>
```

```

        <artifactId>guice</artifactId>
        <version>3.0</version>
    </dependency>

    <!-- com.google.inject/guice -->
    <dependency>
        <groupId>velocity</groupId>
        <artifactId>velocity-dep</artifactId>
        <version>1.4</version>
    </dependency>
</dependencies>

<build>
    <plugins>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-surefire-plugin</artifactId>
            <version>2.18.1</version>
            <configuration>
                <properties>
                    <property>
                        <name>usedefaultlisteners</name>
                        <value>>false</value>
                    </property>
                    <property>
                        <name>listener</name>
                        <value>org.uncommons.reporting.HTMLReporter,
org.uncommons.reporting.JUnitXMLReporter</value>
                    </property>
                </properties>
                <workingDirectory>target</workingDirectory>
            </configuration>
        </plugin>
    </plugins>
</build>
</project>

```
